
Facility smart grid information model

Modèle d'informations des réseaux électriques intelligents des installations

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



COPYRIGHT PROTECTED DOCUMENT

© ISO 2017, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. International Standards are drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 205, *Building environment design*.

ASHRAE Standard Project Committee 201
Cognizant TC 1.4 Control Theory and Application
TC 7.5 Smart Building Systems
SPLS Liaisons: Steven Bruning, Hoy R. Bohanon, John Clark

Steven T Bushby, <i>Chair*</i>	Mike Gibson	David R. Pospisil*
Stephen D. Kennedy, <i>Co-Vice Chair*</i>	Matt Gillmore	Bin Qui
Sharon Dinges, <i>Co-Vice Chair*</i>	Krishnan Gowri	Devin Rauss
Robert Hick, <i>Co-Vice Chair*</i>	David W. Guelfo	Steven Ray*
Robert J. Alvord	Jason M. Hanna	Tobin Richardson
Chandrashekhar Appanna*	Richard Harwell	Jeremy Roberts
Peter A. Baselici	Howard Holms	David Robin*
Timothy O. Beight	Joseph W. Hughes	Steven Rosenstock
Joel Bender*	Christopher Johnson	John I Ruiz*
Roger L. Boydston	Allen Jones*	Chantipal Sourignavong
James Butler	David Kaufman	Don Sturek
Matthew Bye	Christopher Kotting*	John Teeter*
Michael Coop	Chuck McParland	Natarajan Venkatakrishnan
Abigail Daken*	Richard Morgan	Kenneth Wacks*
Gregory M. Dobbs	John Nunneley	Eric Winkler*
Jonathan D. Douglas	Robert L. Old	Jacob Yackenovitch*
Amr E. Gado*	Mary Ann Piette	Scott Ziegenfus*

* Denotes voting members at time of approval for publication.

Consultants to the Project Committee

Martin Burns	Jerald Martocci	Evan Wallace
Francis Cleveland		

ASHRAE STANDARDS COMMITTEE 2015-2016

Douglass T. Reindl, <i>Chair</i>	Steven J. Emmerich	Cyrus H. Nasser
Rita M. Harrold, <i>Vice-Chair</i>	Julie M. Ferguson	Heather L. Platt
James Dale Aswegan	Walter T. Grondzik	David Robin
Niels Bidstrup	Roger L. Hedrick	Peter Simmonds
Donald M. Brundage	Srinivas Katipamula	Dennis A. Stanke
John A. Clark	Rick A. Larson	Wayne H. Stoppelmoor, Jr.
Waller S. Clements	Lawrence C. Markel	Jack H. Zarour
John F. Dunlap	Arsen K. Melikov	Julia A. Keen, <i>BOD ExO</i>
James W. Earley, Jr.	Mark P. Modera	James K. Vallort, <i>CO</i>
Keith I. Emerson		

Stephanie C. Reiniche, *Senior Manager of Standards*

SPECIAL NOTE

This American National Standard (ANS) is a national voluntary consensus standard developed under the auspices of the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE). Consensus is defined by the American National Standards Institute (ANSI), of which ASHRAE is a member and which has approved this standard as an ANS, as "substantial agreement reached by directly and materially affected interest categories. This signifies the concurrence of more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that an effort be made toward their resolution." Compliance with this standard is voluntary until and unless a legal jurisdiction makes compliance mandatory through legislation.

ASHRAE obtains consensus through participation of its national and international members, associated societies, and public review.

ASHRAE Standards are prepared by a Project Committee appointed specifically for the purpose of writing the Standard. The Project Committee Chair and Vice-Chair must be members of ASHRAE; while other members may or may not be members of ASHRAE, all must be technically qualified in the subject area of the standard. Every effort is made to balance the concerned interests on all Project Committees.

The Manager of Standards of ASHRAE should be contacted for:

- a. interpretation of the contents of this Standard,
- b. participation in the next review of the Standard,
- c. offering constructive criticism for improving the Standard,
- d. permission to reprint portions of the Standard.

OASIS DISCLAIMER

Excerpts from OASIS [WS-Calendar v1.0 Committee Specification 01] [and] [OASIS Energy Market Information Exchange (EMIX) v1.0 Committee Specification] are reprinted with permission from OASIS Open, and are Copyright © OASIS Open 2009-2012. All Rights Reserved.

[WS-Calendar][and][EMIX] [is]/[are] subject to the terms of the OASIS IPR Policy, and all capitalized terms in the following text have the meanings assigned to them in that Policy, which may be found at the OASIS website: <http://www.oasis-open.org/who/intellectualproperty.php>.

[WS-Calendar][and][EMIX] and translations thereof may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works.

[WS-Calendar][and][EMIX] and the information contained therein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. "

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

CONTENTS

FOREWORD	iii
INTRODUCTION	viii
1. Purpose	1
2. Scope	1
3. Definitions	2
3.1. Terms Adopted from External Sources	2
3.2. Terms Defined for this Standard	3
3.3. Abbreviations and Acronyms Used in this Standard	5
4. FSGIM Structure and Usage	7
4.1. Overview	7
4.2. How the FSGIM Component Model can be Applied to Real-World Examples	9
5. Device and Model Components	14
5.1. Facility Model Overview Diagram	14
5.2. Device	18
5.3. Meter Component	29
5.4. Load Component	36
5.5. Generator Component	59
5.6. Energy Manager Component	94
5.7. Model Elements from External Sources	241
6. Common Primitive Types, Classes, and Enumerations	509
6.1. Time	509
6.2. Enumerations	510
6.3. Primitive Data Types	515
6.4. Measurements	516
6.5. Other Common Classes	548
6.6. Elements Defined in the Common Primitive Types, Classes, and Enumerations Model	551
7. Conformance Requirements	595
7.1. Introduction	595
7.2. Conformance Requirements	596
7.3. Conformance Blocks	599
8. References	815
9. Annex A – UML Model (Normative)	816

INTRODUCTION

The effort to substantially modernize and transform the national electric grid and create what has become known as a "smart grid" is an enormous undertaking that reflects both the size and importance of the electric grid. Viewed in its entirety, it is an international effort involving hundreds of organizations and companies, and it will impact billions of people. The standards infrastructure that will be needed to support this transformation may include over one hundred standards by the time that it is fully in place. This standard is one part of that infrastructure.

Almost all electricity is consumed in a building of some kind – homes, retail establishments, offices, schools, factories, hospitals – the list goes on. This standard attempts to capture the breadth and diversity of these consumers by using the term "facility." A facility is any kind of building or collection of buildings, and all of the electrical loads or local generation sources contained within them or controlled by the facility owner.

Historically, electricity consumption has been viewed as a collection of dumb loads at the end of a distribution system. There has been almost no interaction between the "loads" and those responsible for electricity generations and distribution. The vision of the smart grid changes this picture radically. In a smart grid world, facilities become full partners in supporting and managing the electric grid. Facilities become generators using local renewable or other generation capacity. Facilities moderate electrical demand in response to fluctuations in the price or availability of electricity. Facilities communicate and negotiate with energy providers, sharing information about the facility's projected electrical demand or ability to respond to the energy provider's needs for maintaining grid stability and reliability.

In some respects all facilities have common characteristics and needs with respect to interactions with a smart grid, regardless of whether the facility is a commercial, institutional, or industrial building, or a private home. The Facility Smart Grid Information Model (FSGIM) standard attempts to capture this commonality and standardize the content of the information that a facility manager needs to have, or, in some cases, needs to exchange with the energy provider, in order to manage the facility. Energy providers benefit from the FSGIM standard because it enables interaction with all different types of facilities in a common way. Facility owners benefit because products can be designed for use in multiple facility types and products designed primarily for one type of facility, a home for example, can more easily be used in another, say a commercial building.

An information model is an abstraction, not an implementation. This abstract representation is a way to account for the reality that the technology used to manage a facility may be quite different depending on the type of facility. It is intended that the FSGIM will be used to develop or enhance other standards that define technology and communication protocol specific implementations of the model for particular markets.

The FSGIM was developed in the context of a much larger framework of smart grid standards. It builds on some of those standards in a way that is intended to maintain consistency and harmony with established and developing standards that impact the information needed to managing the facility, while at the same time capturing all of the key information needed in one place.

If the smart grid is to become a reality there must be smart facilities of all types that interact with it. The considerable time and talent that went into developing the FSGIM was invested in order to lay a solid foundation upon which to fulfill this vision.

1. PURPOSE

The purpose of this standard is to define an abstract, object-oriented information model to enable appliances and control systems in homes, buildings, and industrial facilities to manage electrical loads and generation sources in response to communication with a “smart” electrical grid and to communicate information about those electrical loads to utility and other electrical service providers.

2. SCOPE

This model provides the basis for common information exchange between control systems and end use devices found in single - and multi-family homes, commercial and institutional buildings, and industrial facilities that is independent of the communication protocol in use. It provides a common basis for electrical energy consumers to describe, manage, and communicate about electrical energy consumption and forecasts.

The model defines a comprehensive set of data objects and actions that support a wide range of energy management applications and electrical service provider interactions including:

- a) on-site generation,
- b) demand response,
- c) electrical storage,
- d) peak demand management,
- e) forward power usage estimation,
- f) load shedding capability estimation,
- g) end load monitoring (sub metering),
- h) power quality of service monitoring,
- i) utilization of historical energy consumption data, and
- j) direct load control.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

3. DEFINITIONS

See Clause 8 for external standards that are referenced by this standard.

3.1. Terms Adopted from External Sources

The following terms used in this standard are drawn from other standards or published reference sources. The definitions are repeated here and a reference to the appropriate standard or document is provided. Words or phrases in italics refer to terms defined elsewhere in this clause.

asset

A logical entity with measurable and reportable consumption, e.g., an *asset* may be a physical device with its own meter, or the main meter at the *service delivery point* of a service location.
(NAESB 2010a).

baseline

A method of estimating the electricity that would have been consumed by a customer or *demand resource* in the absence of a *demand response event*. It may be calculated using interval metering and/or statistical sampling techniques.
(NAESB 2010a).

combined heat and power

Combined heat and power (CHP), also known as cogeneration, is the simultaneous production of electricity and heat from a single fuel source, such as: natural gas, biomass, biogas, coal, waste heat, or oil.
(<http://www.epa.gov/chp/basic/index.html>).

electrical coupling point

Point of electrical connection between the DER source of energy (generation or storage) and any electric power system (EPS). Each DER (generation or storage) unit has an ECP connecting it to its local power system; groups of DER units have an ECP where they interconnect to the power system at a specific site or plant; a group of DER units plus local loads have an ECP where they are interconnected to the utility power system.
(IEC 61850 Part 7-420).

energy management system

A system used to monitor and control the energy consuming devices in a building. Within this standard, "energy management system" always refers to a customer/facility energy management system and not a utility energy management system.
(EIS Alliance Use-Cases).

non-spinning reserve

1. That generating reserve not connected to the system but capable of serving demand within a specified time.
2. Interruptible load that can be removed from the system in a specified time.
(NERC Glossary).

peak load

The maximum amount of power delivered (*load*) for a given time period.
(NEMA glossary (<http://www.nema.org/gov/energy/glossary/>)).

power quality

Characteristics of the electricity at a given point on an electrical system, evaluated against a set of reference technical parameters.

NOTE: These parameters might, in some cases, relate to the compatibility between electricity supplied on a network and the loads connected to that network.
(IEC 61000-4-30 (2008)).

pricing node (Pnode)

A single network Node or subset of networked Nodes where a physical injection or withdrawal is modeled and for which a Locational Marginal Price is calculated and used for financial settlements.
(CAISO Tariff).

program administrator

An investor-owned, governmental or cooperative distribution company with the responsibility for developing and operating specific programs.
(NAESB REQ.0.2.179).

spinning reserve

Unloaded generation that is synchronized and ready to serve additional demand.
(NERC Glossary).

3.2. Terms Defined for this Standard**aggregated pricing node**

An *aggregated pricing node* is a location in the electric grid where the locational marginal price is calculated by a weighted average of one or more underlying *pricing nodes*.

capacity restraint

A limitation in the generating capacity or transmission capacity of an energy supplier that constrains the amount of electricity that is available to be bought or sold.

Commercial Buildings Energy Consumption Survey

A national sample survey that collects information on the stock of U.S. commercial buildings.

control system

A device or set of devices used to manage, command, direct, or regulate the behavior of other devices or systems.

critical loads

Individual *loads* deemed to be critical to the operation of the facility or process.

critical peak period

A time period during which a special high price for electricity is applied as a way to reduce demand.

curtailable load

Loads whose power consumption can be increased or decreased for the purpose of increasing or decreasing the total electrical demand on the grid.

demand

The rate at which energy is delivered to or used by a system or part of a system at a given instant in time or averaged over any designated interval of time.

demand forecast

Predictions regarding future electrical demand.

demand interval

The period of time over which an energy supplier calculates demand. Typical values range from five minutes to 60 minutes.

demand resource

A *load*, aggregation of loads, behind-the-meter generator, electrical storage system, or thermal storage system capable of providing measurable and verifiable *demand response*.

demand response

A temporary change in electricity usage by a *demand resource* in response to market or reliability conditions.

demand response event

A period of time defined by the *program administrator*, including notifications, deadlines, and transitions, during which *demand resources* provide *demand response*. All notifications, deadlines, and transitions may not be applicable to all *demand response* products or services.

demand target

Target in *demand* management; *peak demand* is a key factor in contract charges for electricity use.

direct load control

A *demand response* activity by which the *program administrator* remotely shuts down or cycles a customer's electrical equipment (e.g., air conditioner, water heater). Direct load control programs are primarily offered to residential or small commercial retail customers.

dispatchable generator

Generators whose power supply may be increased or decreased for the purpose of decreasing or increasing the total electrical demand on the grid.

distributed energy resource

A small, modular, energy generation or storage device located within the electrical distribution system at or near the end user. Distributed energy resources may be connected to the local electrical power grid (e.g., for voltage support) or isolated from the grid in standalone applications, such as part of a micro grid.

emergency standby generation

The provision of capacity deployed by the balancing authority to meet NERC and regional reliability organization contingency requirements.

end device

A physical end-use device that consumes or supplies electricity.

energy consumption

A measure of the quantity of electricity used in a given time period, measured in kilowatt hours.

energy cost

That portion of the charge for electric service based upon the electric energy (kWh) consumed or billed.

energy emissions

The pollution emissions associated with generating a quantity of electrical energy.

energy information provider

A business that provide energy supply or *demand* information to customers or authorized third parties.

energy services company

A business providing *energy* savings, efficiency, and *generation* solutions.

energy services interface

An abstract representation of a bi-directional communication interface between devices within a facility and an external electric *service provider*. This interface enables exchange of information including pricing, energy consumption, demand response interactions, load forecasts, energy production forecasts, and weather information.

energy supplier

An entity that delivers electricity to end use customers. Examples include investor-owned utilities, municipal utilities, and private companies.

forward pricing forecast

The compilation of prices of actual transactions for forward (future) delivery periods that are executed today.

independent system operator

An independent system operator maintains balance of the grid system by controlling the dispatch of plants and ensuring that loads match system resources. As such, the operator must be neutral and independent.

intelligent end point

A system that internally measures, collects, and analyzes its *energy cost/consumption*.

load

A device, system or process that consumes electrical energy.

local codes

Local laws, regulations, and building codes that determine building operation and capabilities.

measurement and verification

Energy measurements that can be determined to a degree of accuracy and trust that is acceptable to all stakeholders.

net demand

Total demand minus total supply. Net demand is determined at the facility interconnection point, billing meter, or internal location, and may be positive or negative.

peak demand

The highest measured *demand* encountered during a specified period of time (e.g., month, year, or finite set of time intervals).

plug load

A device that is powered by means of an electrical plug and matching socket or receptacle. This excludes devices that are accounted for as part of major building end uses such as HVAC, lighting systems, and water heating.

present demand

Demand occurring during the present *demand interval* or subinterval (e.g., Watts or VA).

present subinterval demand

Average rate of energy used over a small portion of a *demand interval*.

present subinterval net demand

Average net rate of energy used over a small portion of a *demand interval*.

present subinterval supply

Average rate of energy supplied over a small portion of a *demand interval*.

ramp rate

The rate at which a generator changes its output or a *demand resource* changes its demand.

service delivery point

The location where electric service is delivered to the facility.

service level agreement

A part of a service contract where the level of service is formally agreed upon.

smart power distribution unit

A device that measures and distributes electric power to an electrical circuit.

smart power strip

A power strip that measures electrical power being drawn by the connected equipment.

supply

The rate of energy generation or storage output that is used to serve facility load and possibly provided to the grid as a generation resource.

time of use pricing

A rate structure characterized by different prices for electricity use in a 24-hour time frame. Time of use pricing is generally used to encourage electricity use during periods of lower demand and discourage electricity use during periods of high demand.

use case

A story, told in structured and detailed steps, about how participants in an Application collaborate to reach a goal.

variable generation

Variable generation comprises any power generating device in which the source of the energy is not dispatchable and that the resource does not have control over its net power output. Variable generating devices include those powered by wind, solar, run of river hydro, tidal, wave, ocean thermal and other renewable resources.

3.3. Abbreviations and Acronyms Used in this Standard

ASHRAE	American Society of Heating, Refrigerating and Air-Conditioning Engineers
CBECS	<i>Commercial Buildings Energy Consumption Survey</i>
CHP	combined heat and power
DER	distributed energy resource
ECP	electrical coupling point
EIP	Energy Information Provider
EM	Energy Manager
EMS	energy management system
ESCO	energy services company

ISO 17800:2017(E)

ESI	Energy Services Interface
ESI EM	Energy Services Interface Energy Manager
ESP	Energy Service Provider
HMI	human machine interface
HVAC	Heating, ventilating, and air conditioning
IEP	Intelligent End Point
ISO	International Organization for Standardization
kW	kilowatt
M&V	Measurement and Verification
OASIS	(Organization for the Advancement of Structured Information Standards) is a not-for-profit consortium that drives the development, convergence and adoption of open standards for the global information society.
RTO	Regional Transmission Operator
SCADA	supervisory control and data acquisition
SLA	Service Level Agreement
SP	Service Provider
SPDU	Smart Power Distribution Unit
SPS	Smart Power Strip

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

4. FSGIM STRUCTURE AND USAGE

4.1. Overview

The Facility Smart Grid Information Model (FSGIM) defines an abstract representation of the energy consuming, producing, and storage systems found in residential, commercial, and industrial facilities. By modeling the energy characteristics of this equipment, the characteristics of a facility's energy needs can be determined. When this information is included in equipment and communicated with a protocol, a facility owner can understand what factors influence the facility's energy consumption; energy consultants can determine how to effectively reduce the energy profile of a facility; architects and engineers can design facilities that optimize the energy profile; controls manufacturers can create products that monitor and manage the facility energy profile; and energy providers can more accurately forecast energy demand as well as the reactions to energy supply constraints.

A key concept in the FSGIM is a "facility." A facility is any kind of building or collection of buildings, and all of the electrical loads or local generation sources contained within them or controlled by the facility owner. Examples of facilities include homes, retail establishments, offices, schools, factories, hospitals, and college campuses.

In some respects all facilities have common characteristics and needs with respect to interactions with a smart grid. The FSGIM attempts to capture this commonality and standardize the content of the information that a facility manager needs to have, or, in some cases, needs to exchange with the energy provider, in order to manage the facility. The FSGIM encompasses detailed information about systems inside the facility that can be thought of as tightly coupled in the sense that they are directly managed and controlled within the facility. Examples include HVAC, lighting, security, facility management systems, and industrial automation systems. The FSGIM also includes details about external systems that can be thought of as loosely coupled in the sense that the facility interacts with them but does not have any direct control over them. Examples of this include sources of weather information, real-time energy pricing information, demand response signals, and sources of energy usage information.

Energy providers benefit from the FSGIM standard because it enables interaction with all different types of facilities in a common way. Facility owners benefit because products can be designed for use in multiple facility types and products designed primarily for one type of facility can more easily be used in another.

In order to guide the content of the FSGIM a set of use cases was developed and vetted with a broad array of industry experts. These use cases define a wide range of scenarios for interaction between a facility and an energy provider. They define specific actors for the transactions and the information that needs to be exchanged to carry out the scenario. In addition to these interactions between the facility and energy provider, the information model also addresses information that is needed within the facility to make control decisions when managing loads and to develop load forecasts.

An information model is an abstraction, not an implementation. This abstract representation is a way to account for the reality that the technology used to manage a facility may be quite different depending on the type of facility. It is intended that the FSGIM will be used to develop or enhance other standards that define technology and communication protocol specific implementations of the model for particular markets. By supporting the same FSGIM, a level of interoperability is ensured since the semantic meaning of the information is standardized. A protocol translator may still be needed to translate between different message syntaxes and signaling but the information content is understood. In this way, the FSGIM will provide the basis for interoperable extensions to new and existing facility information communications protocols. Figure 4.1 illustrates the types of information standardized in the FSGIM and how that relates to the additional details that will be provided by protocol specific implementations.

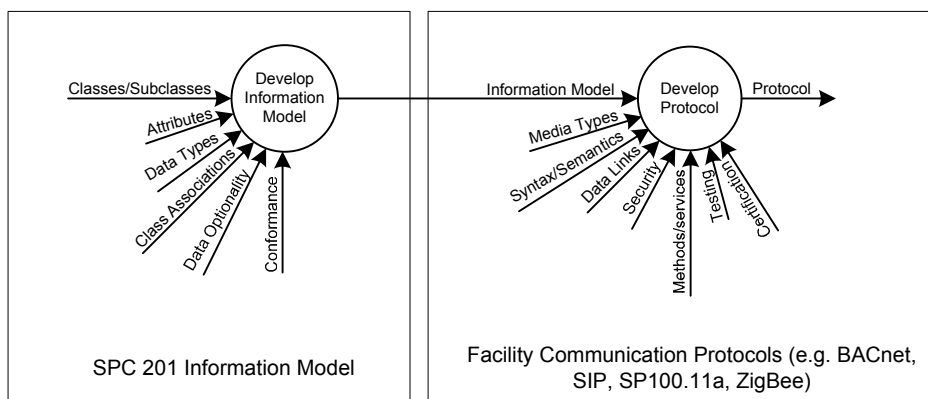


Figure 4.1 - The Facility Smart Grid Information Model will be implemented through modifications to existing facility communications protocols.

In order to fulfill the needs of diverse kinds of facilities described above, it is necessary to structure the FSGIM in a way that enables considerable flexibility and scalability. The approach taken was to define an information model that consists of four major components or building blocks. Those components are a Generator Component, a Meter Component, a Load Component, and an Energy Manager Component. These building blocks can be arranged and combined as needed to represent physical devices, systems, or functions. For example, storage can be represented by combining characteristics of the Load Component (when charging) and the Generator Component (when discharging). A meter might simply be a measuring device or, under some circumstances it might be a "smart meter" that plays a role in controlling loads in addition to its measurement function. Such a smart meter would be represented in the FSGIM as a combination of the Meter Component and the Energy Manager Component. The FSGIM components can also represent aggregations. For example the Load Component can represent characteristics of an individual device or the combined demand of a collection of devices or other loads. With this kind of flexibility it is expected that the information model components can provide all of the needed functionality and flexibility. Note: In a facility, there may be many real-world devices contributing to the computation of an aggregated value. As such, the details of these contributing values and how they are obtained may vary. Thus, judgments must be made in determining how well they can be aggregated. For additional information about aggregation in the FSGIM see Clause 5.6.3.1.

The structure and semantics of the FSGIM are defined using the Unified Modeling Language (UML) [<http://www.omg.org/spec/UML/2.3/>]. An electronic version of the standard suitable for use with UML software tools accompanies the printed standard. The text of most portions of the standard was generated from documentation included in the UML files. UML diagrammatic conventions are used in the figures contained in this standard and define normative relationships and multiplicity details of the FSGIM. A summary of UML conventions is contained in Annex B.

To avoid possible confusion over similar terms used in the model definition a special effort is made to fully qualify terms used to describe portions of the model described in the text of the standard. The four model component are all always referred to as a Generator Component, a Load Component, a Meter Component, and an Energy Manager Component. In each case there is an information model class defined with a similar name. These are the Generator Class, Load Class, Meter Class, and Energy Manager Class. In cases where a physical device represented by the information model is intended it is referred to as the normal English noun without special qualification. For example, a device that measures electrical energy consumption is a meter. The attributes of this meter can be represented using the details of the Meter Class and other related classes defined in the Meter Component of the FSGIM.

The FSGIM was developed in the context of a much larger framework of smart grid standards. It builds on some of those standards in a way that is intended to maintain consistency and harmony with established and developing standards that impact the information needed to manage the facility, while at the same time capturing all of the key information needed in one place. These standards include several IEC standards developed and used by the electricity generation and distribution industry, OASIS WS-Calendar [OASIS WS-Calendar v1.0 (see <http://www.oasis-open.org/committees/ws-calendar/>)], and OASIS Energy Market Information Exchange [(EMIX) v1.0 (see <http://www.oasis-open.org/committees/emix/>)].

The FSGIM defines the datatypes of the attributes used in the model. A convention has been adopted to define the datatype in the broadest sense appropriate to the attribute. For example a measurement would have a datatype of Real, meaning the mathematical concept of a Real number. A protocol specific implementation would map this Real value to something that can actually be represented in a computer, e.g. a Float, that has constraints on the precision and size that can be represented which are not part of the mathematical definition of a Real number. This was done to be more faithful to the meaning of the information being represented while not imposing constraints on the computer hardware and communication protocols that will be used to implement the FSGIM. In cases where

portions of the FSGIM are derived from other standards, datatypes are mapped as needed to be consistent with this convention.

4.2. How the FSGIM Component Model can be Applied to Real-World Examples

Imagine an owner-occupied commercial building in a major metropolitan area that is served by a single retail electric provider. Assume that building is used for commercial office space. In this type of building, the main energy consuming systems are lighting, heating ventilation and air conditioning (HVAC), and information and communications technology (ICT) systems. To protect the company's data center, there is a battery powered uninterruptible power supply for the ICT system. To reduce daytime energy consumption, the building owner has installed solar photovoltaic panels on the building's roof. To ensure business continuity in case of a power failure, the owner has also installed a backup generator system. A drawing of this building is shown in Figure 4.2. This example will illustrate how the FSGIM components can be used to represent these physical devices and their function. Individual subsystems will be modeled and then combined to show the overall results.

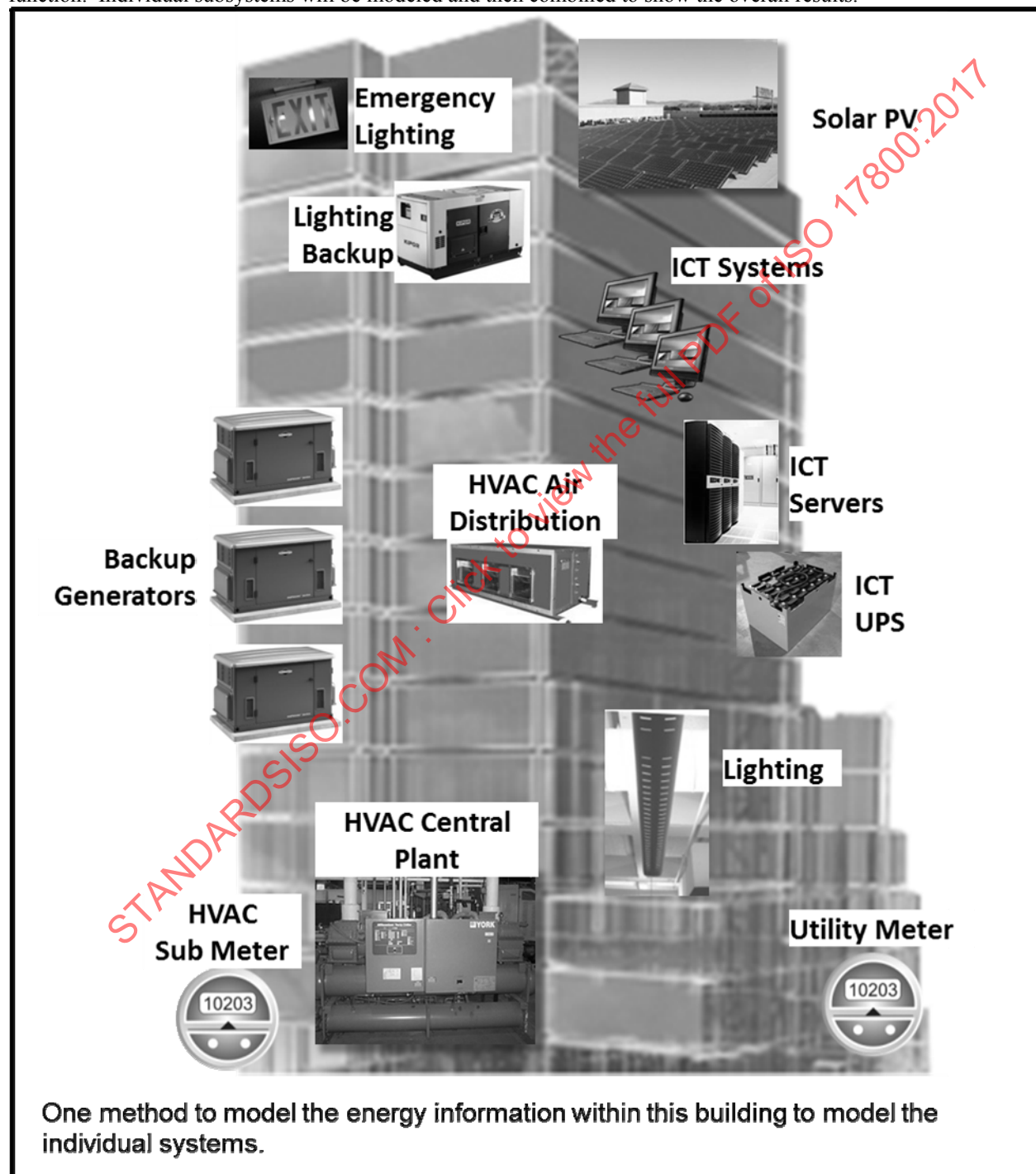


Figure 4.2 - Example Customer Facility Showing Major Energy Components

4.2.1. Modeling the Backup Generators

The facility's backup generator system consists of three separate generators controlled by a generator control system. Since this equipment could be used by the facility to provide services to the grid (e.g., demand response) we have

chosen to model each generator individually instead of as a single composite generator resource. In this case, the generator control system is modeled as an instance of an EM Class while each generator is modeled as an instance of a Generator Class. The FSGIM representation of this sub-system is shown in Figure 4.3.

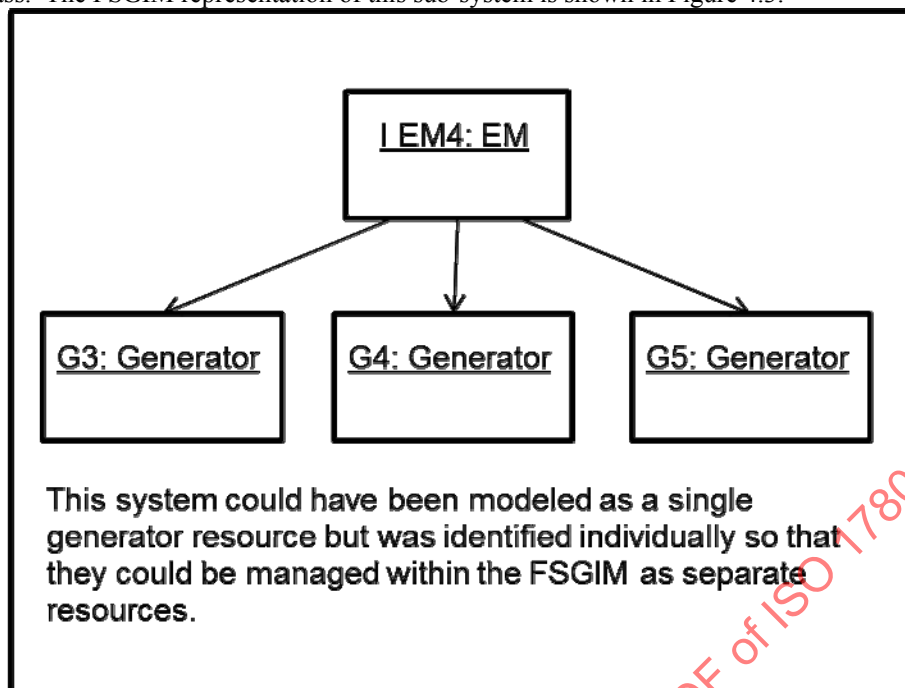


Figure 4.3 - Example of an FSGIM Model for a Generator System of Three Generators

4.2.2. Modeling the ICT System

The ICT system provides the internet, telecommunications, security, personal computers, and servers for the facility. A network management system provides the intelligence for monitoring and managing the energy consumption of this equipment and is therefore modeled as an instance of an EM Class. The equipment (PCs, servers, cameras, phones) managed by the sub-system are modeled as instances of the Load Class and the UPS is modeled as a combination of an instance of a Generator Class (when producing power) and an instance of a Load Class (when being recharged). The FSGIM representation of this sub-system is shown in Figure 4.4.

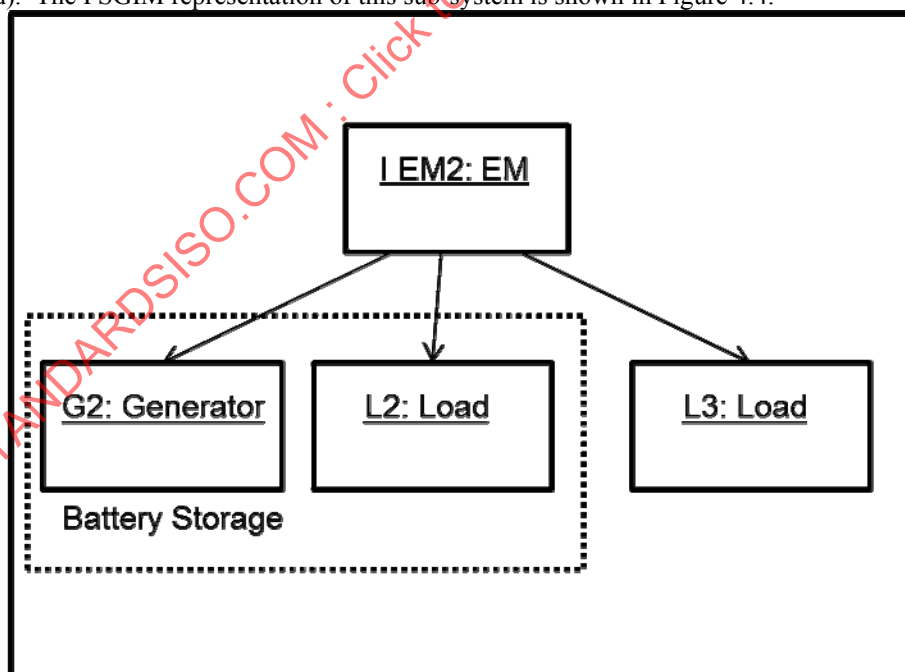


Figure 4.4 - Example of an FSGIM Model for an ICT system that contains a network manager, equipment, and a UPS.

4.2.3. Modeling the Lighting System

The lighting system provides floor, outdoor, and emergency lighting. A lighting control system manages the floor and outdoor lighting while an emergency lighting panel manages the exit lighting. A small generator is used to illuminate the emergency lighting in case of power failure. The lighting control system and emergency lighting control panel are modeled as instances of the EM Class. The indoor/outdoor and emergency lights are modeled as

instances of the Load Class. The FSGIM representation of this sub-system is shown in Figure 4.5. Note that for simplicity the indoor/outdoor lighting is represented as a single Load Class instance, but it could have been represented as several Load Class instances.

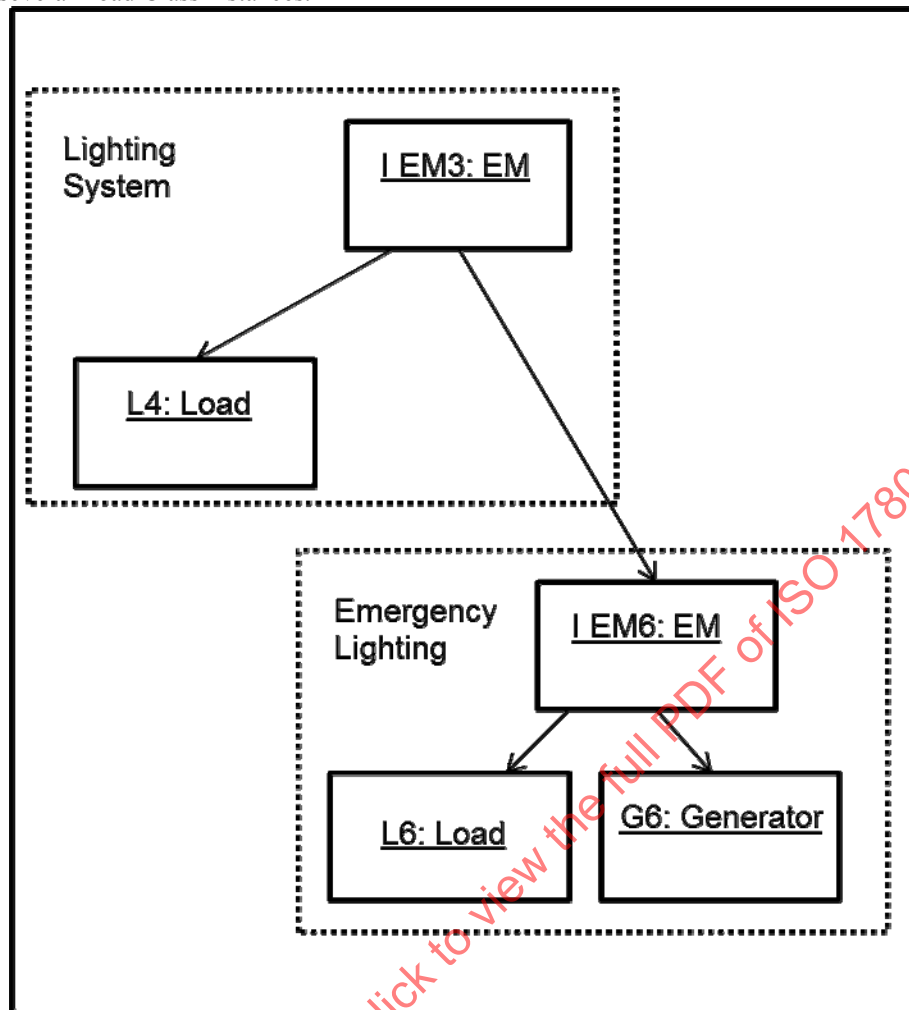


Figure 4.5 - Example of an FSGIM Model for a Lighting System

4.2.4. Modeling the HVAC System

The HVAC system in this example consists of a central plant and an air distribution system. The central plant contains a chiller and pumps, central plant controller, and a sub-meter. For simplicity in this example, the chiller and pump system are modeled as a single instance of the Load Class. The central plant controller is modeled as an instance of an EM Class, and the sub-meter is modeled as an instance of a Meter Class. The air distribution system contains a number of constant volume air handlers. For simplicity in this example, they are modeled as a single instance of the Load Class since there is a single sub-meter that monitors their consumption. The air handler controllers are modeled as an instance of the EM Class. The FSGIM representation of this sub-system is shown in Figure 4.6.

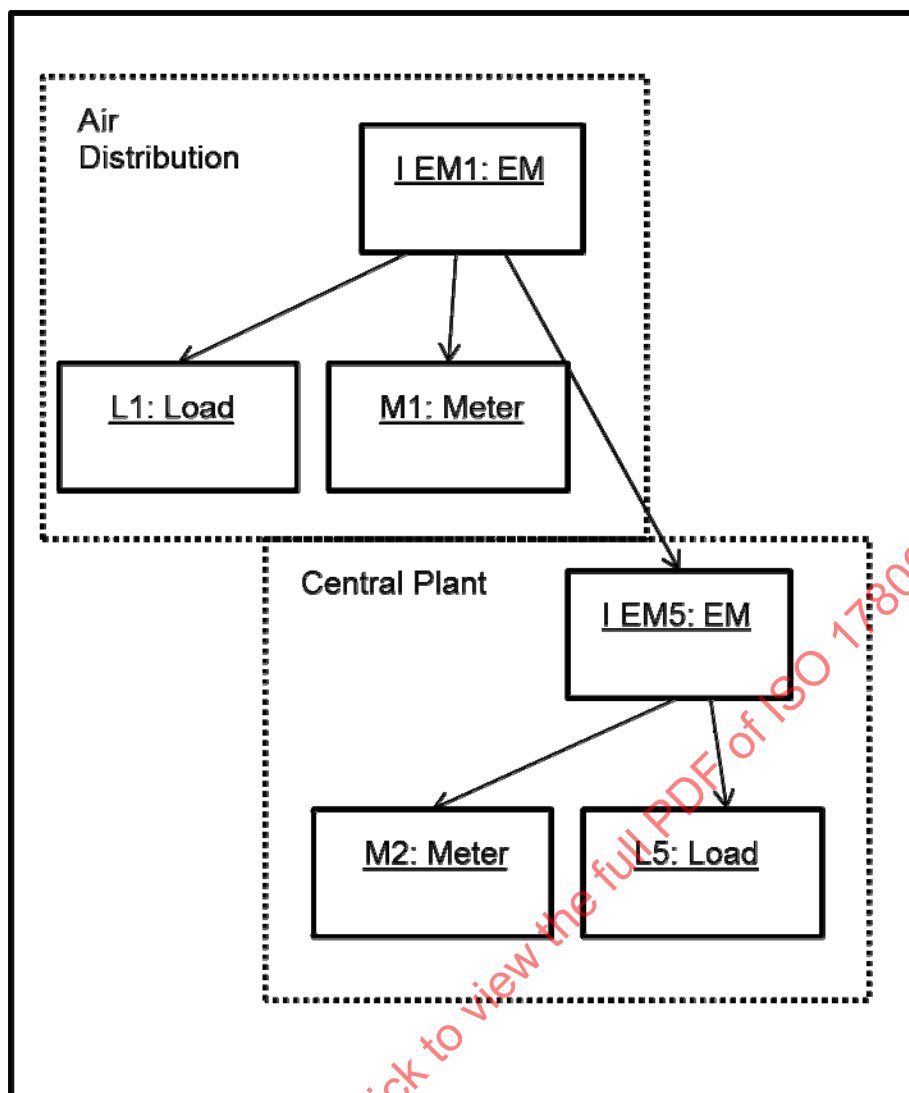


Figure 4.6 - Example of an FSGIM Model for an HVAC System that contains a central plant and air distribution system.

4.2.5. Modeling the Photovoltaic Array and Grid Interface

Electricity enters the facility at a service delivery point through the utility meter. In order to interact with the smart electric grid, energy information is exchanged between the electric delivery system, external energy information providers, and the facility. This interface is called the Energy Services Interface (ESI). This is where consumption information, rate schedules, demand response requests/responses, energy forecasts, and weather information are exchanged. For simplicity in this example, it is assumed that the ESI is connected through the utility meter and is interfaced to the energy management system. In the FSGIM, a specialized form of an EM Class called an ESI EM Class is used to model the facility side of the ESI. In this example, the energy manager represented by an instance of the ESI EM Class also performs an internal energy management function to control the photovoltaic array system. The photovoltaic array system is modeled as an instance of a Generator Class. The FSGIM representation of this sub-system is shown in Figure 4.7.

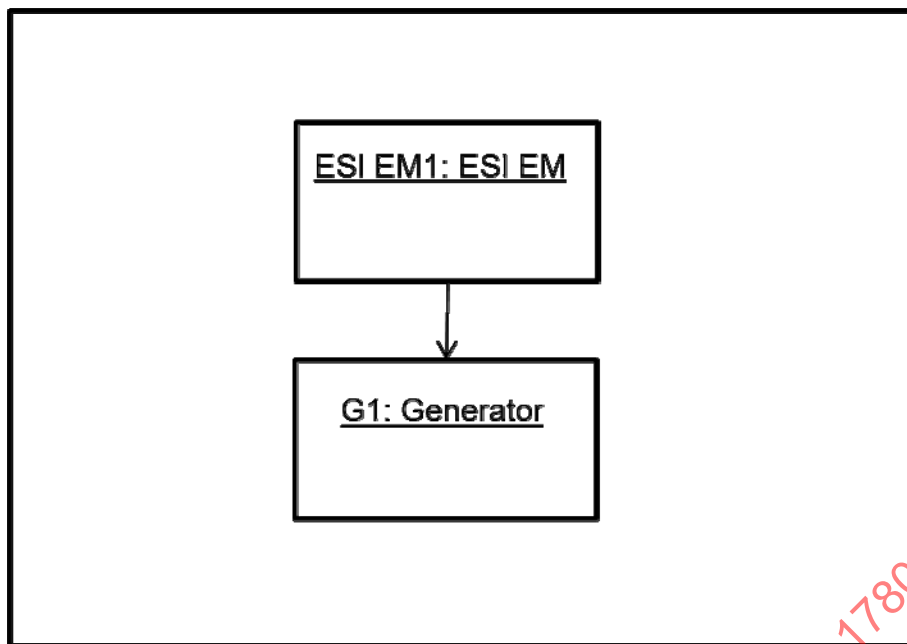


Figure 4.7 - Example of an FSGIM Model of a System that Contains an Energy Manager and a Generator.

4.2.6. Combining All of the Subsystems

To create a model of an entire facility the subsystems are interconnected together as shown in Figure 4.8 in a topology diagram. Once all of the subsystems are connected together, the energy information of the entire system can be collected and disseminated. The FSGIM provides the energy information semantics that are needed to be communicated without limiting the communications methodologies used. The energy managers are responsible for aggregating energy information provided by other energy managers farther down the topology. In this way, a facility can contain many energy managers that cooperate to meet energy goals and provide energy information.

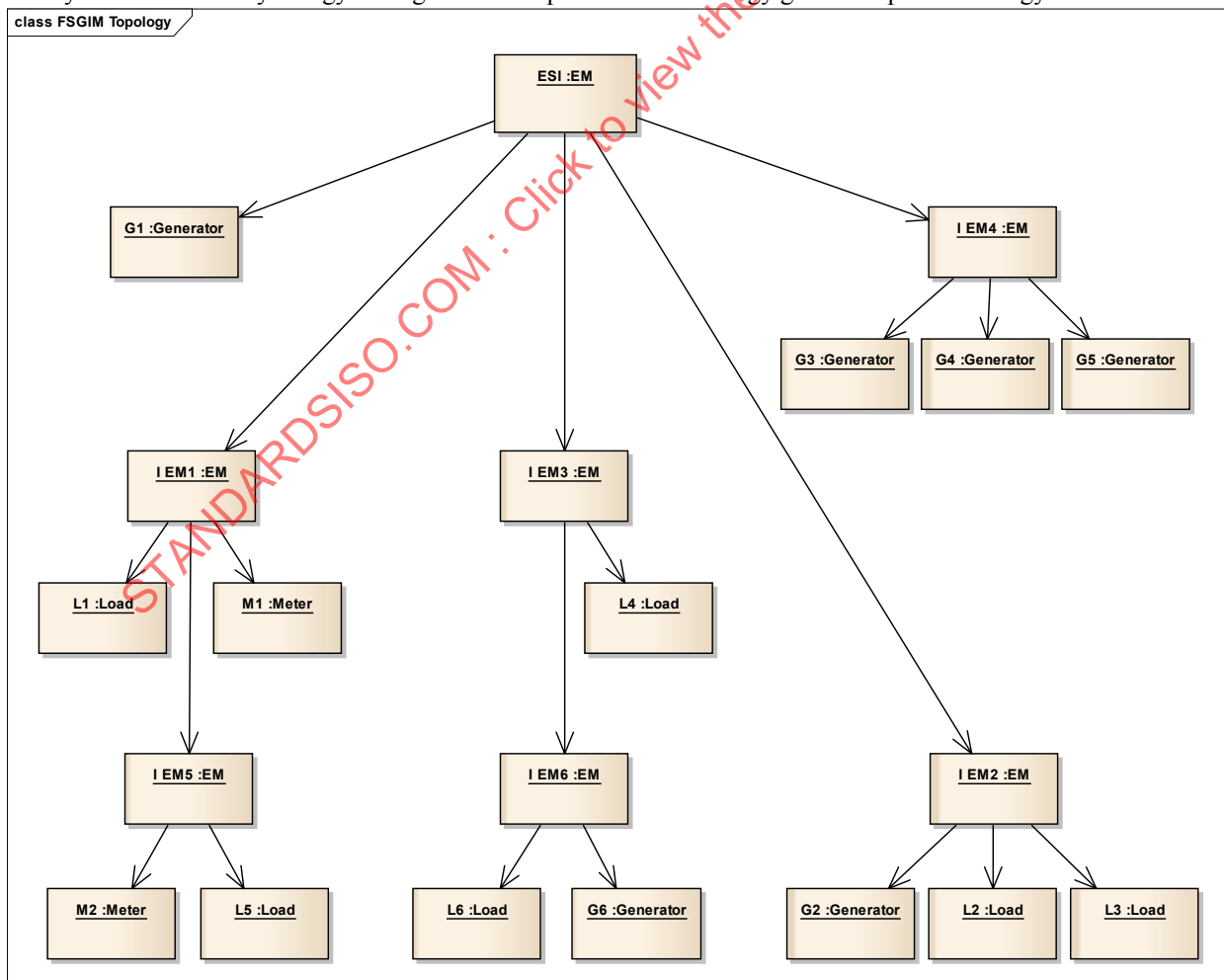


Figure 4.8 - Example Topology Diagram Based on the FSGIM

5. DEVICE AND MODEL COMPONENTS

The FSGIM is comprised of four distinct components. These modeling components can be arranged and combined as needed to represent physical devices, systems, or functions.

The model components are:

- Meter Component -- an abstract representation of any device that measures electricity or emissions;
- Load Component -- an abstract representation of any device that consumes electricity;
- Generator Component -- an abstract representation of any device that produces electricity; and
- Energy Manager Component -- an abstract representation of any device that makes control decisions about energy generation or consumption.

The components are defined and described using UML class diagrams and detailed descriptions of class attributes and enumerations. For each model component the overall structure is displayed in a series of class diagrams. Each class diagram is followed by a summary description of the elements in the diagram, including a cross-reference to the subclause that provides all of the element details.

UML Package Location: *Model\Device and Model Components*

5.1. Facility Model Overview Diagram

This diagram shows the major elements within the Facility Smart Grid Information Model and where they fit within an architecturally layered interface view of a facility.

- Layer 1 represents data primarily coming from sources outside of the facility where these sources are loosely coupled with the systems in the facility.
- Layer 2 shows how the ESI EMs represent the information that is exchanged between the facility and the external world. The package symbols show the external standards that have been used by the FSGIM.
- Layer 3 describes how energy managers represent the information used within the facility for monitoring, control, and analysis.
- Layer 4 represents protocols used within the facility.

As one moves down through the layers tighter coupling among systems is expected. The scope of FSGIM stops short of providing for the representation of local control categories as indicated in the bottom boundary of the diagram.

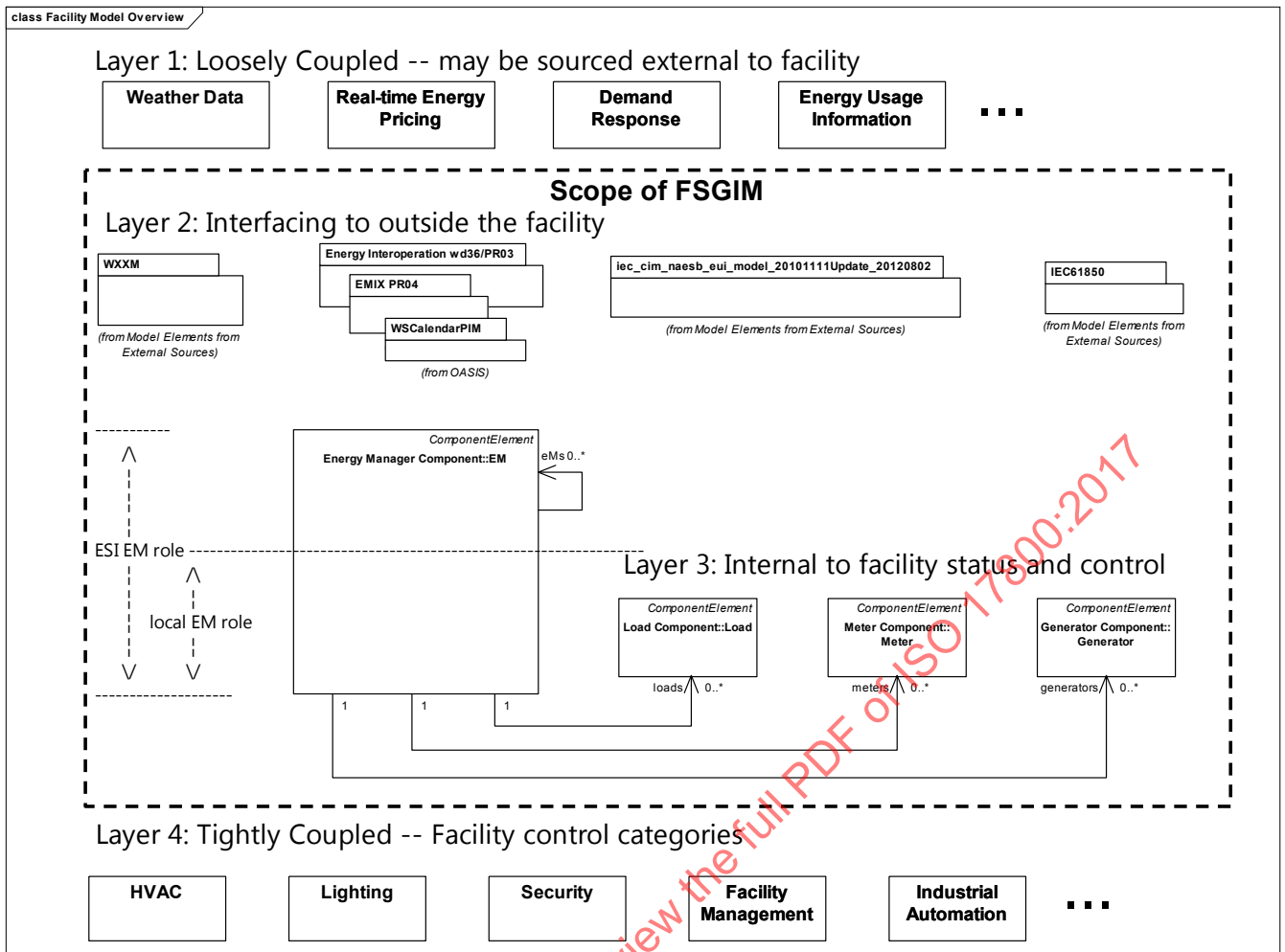


Figure 5.1 - Facility Model Overview

The Facility Model Overview Diagram makes use of the following elements:

EM (Class)

The EM Class is the central class of the Energy-Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

EMIX PR04 (Package)

The Energy Market Information Exchange (EMIX) Component is the portion of the FSGIM that provides an abstract representation of information needed to exchange Price and Product information for power and energy markets. Product definition includes quantity and quality of supply as well as attributes of interest to consumers distinguishing between power and energy sources.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04*. See Clause 5.7.2 for the complete definition of this element.

Energy Interoperation wd36/PR03 (Package)

The Energy Interoperation Component is the portion of the FSGIM that provides an abstract representation of information needed for collaborative and transactive use of energy. It defines information used to coordinate energy supply and use, including power and ancillary services, between two parties such as energy suppliers and facilities.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03*. See Clause 5.7.1 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

iec_cim_naesb_eui_model_20101111Update_20120802 (Package)

This package, derived from NAESB REQ.18/WEQ.19, contains information elements used in the Meter Component, Energy Manager Component, FSGIM measurements, and exchange of energy usage information with external parties.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802*. See Clause 5.7.5 for the complete definition of this element.

IEC61850 (Package)

IEC 61850 is the international standard for communications with electric power and related system intelligent electric devices (IED). There are many parts to this standard.

FSGIM adopts parts of the information model from IEC61850-7-2,3,4 and 420. The latter part deals with the modeling of distributed energy resources (DER). It contains a set of model classes that allow for the nameplate, status, and control of DERs.

These model elements were extracted from the corresponding 61850 parts and were modified to use the primitives of FSGIM where there was a one to one correspondence. LogicalNode class names were expanded to make them more "human readable". No semantic changes were made to the imported model elements.

UML Element Location: *Model\Model Elements from External Sources\IEC61850*. See Clause 5.7.4 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

WSCalendarPIM (Package)

This is a tailored version of the WSCalendar PIM for use in FSGIM.

A PIM is a model of a system designed to be independent of the details needed for particular implementation technologies.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM*. See Clause 5.7.3 for the complete definition of this element.

WXXM (Package)

This package, derived from the Weather Exchange Model (WXXM), contains information elements used in the Energy Manager Component to represent weather observations, hazards, and forecasts.

FSGIM has imported this UML Model, Version 1.1.1., and tailored it as follows:

1. A subset of classes needed for FSGIMWeather were selected.
2. Primitive data types were normalized to FSGIM common primitive types.

3. The inheritance hierarchy was abbreviated while preserving all inherited attributes and relationships. Local versions of deeply nested inheritance were simplified by hand-assembling equivalent classes.
4. Enumerations were coerced to the enumeration style and stereotypes of FSGIM.

These modifications were designed to capture the desired subset of functionality, improve consistency, and preserve all semantics of the base WXXM standard. In each case, the FSGIM class retains the name of the WXXM class and additionally, the XMI identifier of the original to provide traceability back to the original class.

UML Element Location: *Model/Model Elements from External Sources/WXXM*. See Clause 5.7.6 for the complete definition of this element.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

5.2. Device

The Device represents a physical entity that is then modeled using the remaining classes within the FSGIM. This portion of the model also includes classes that are used to name and tag these physical devices and the logical roles that these physical devices assume.

UML Package Location: *Model\Device and Model Components\Device*

5.2.1. Device Diagram

This diagram shows the basic composition of the FSGIM model. It shows how a physical device is decomposed into one or more of its logical components.

Components represent the various roles played by an actual device as viewed by the FSGIM.

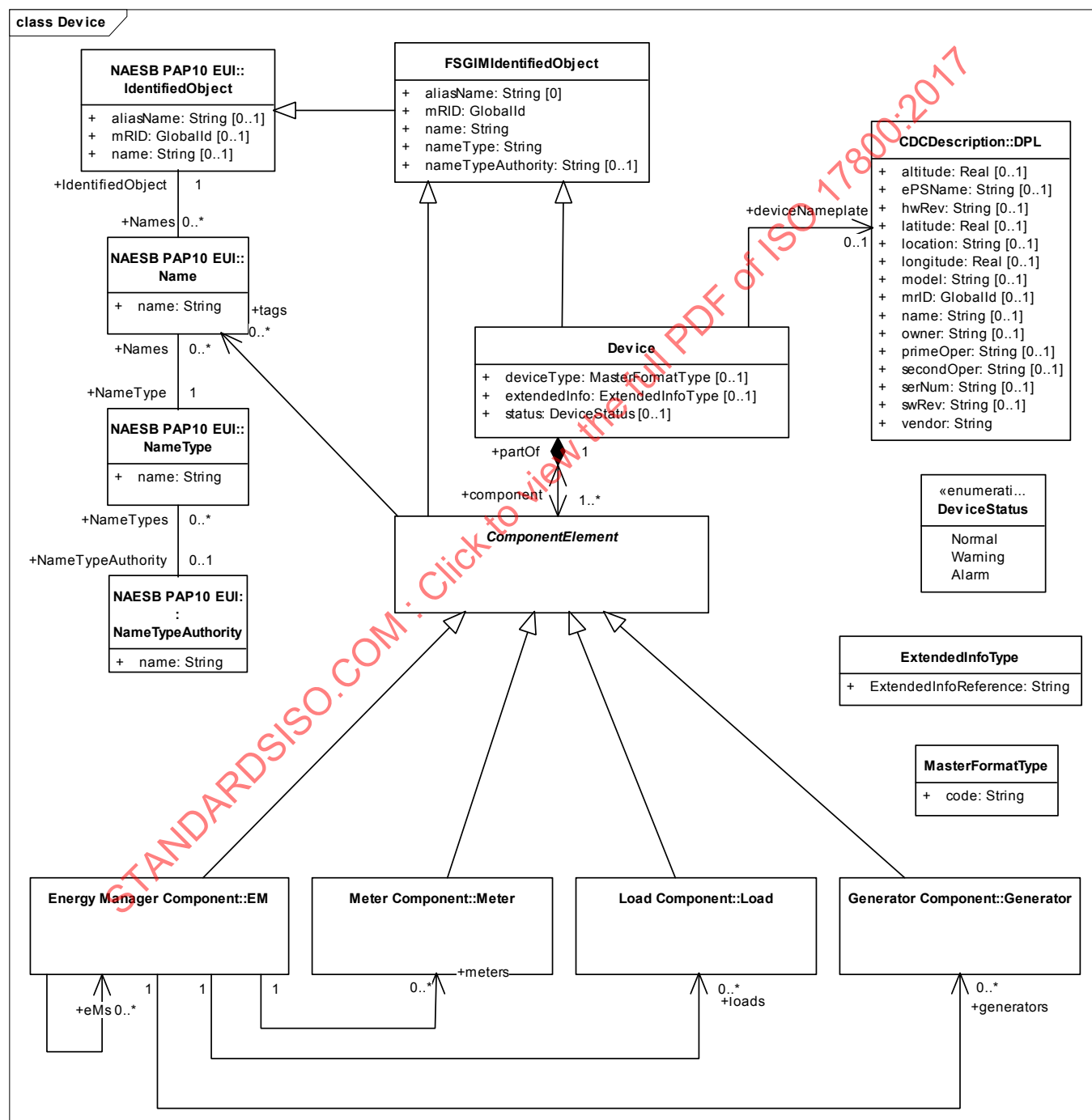


Figure 5.2 - Device

The Device Diagram makes use of the following elements:

ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*. See Clause 5.2.3.1 for the complete definition of this element.

Device (Class)

This is the class that represents a physical device in the facility

UML Element Location: *Model\Device and Model Components\Device\Device*. See Clause 5.2.3.2 for the complete definition of this element.

DeviceStatus (Enumeration)

The general indication of the health of a Device.

UML Element Location: *Model\Device and Model Components\Device\DeviceStatus*. See Clause 5.2.3.6 for the complete definition of this element.

DPL (Class)

This common data class is used to identify entities like primary equipment or physical devices.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCDescription\DPL. See Clause 5.7.4.5.3.1.18 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

ExtendedInfoType (Class)

This class defines an reference external from this model that can be used to gain additional information about the device.

UML Element Location: *Model\Device and Model Components\Device\ExtendedInfoType*. See Clause 5.2.3.3 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

MasterFormatType (Class)

This class defines a string attribute with specific constraints to map into the Master Format 2012 numbering system.

UML Element Location: *Model\Device and Model Components\Device\MasterFormatType*. See Clause 5.2.3.5 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Name*. See Clause 5.7.5.2.1.14 for the complete definition of this element.

NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameType*. See Clause 5.7.5.2.1.15 for the complete definition of this element.

NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameTypeAuthority*. See Clause 5.7.5.2.1.16 for the complete definition of this element.

5.2.2. Object Identification Diagram

This diagram illustrates the correspondence between the NAESB standard use of IdentifiedObject and the FSGIM profile of that class and the associated Name class.

In the FSGIM, the mRID becomes a required globally unique identifier. This makes instances of objects that inherit from IdentifiedObject recognizable regardless of the protocol transformations that may occur as information is exchanged via FSGIM-implementing protocol standards. That is, while the syntactical encoding of the identifier may be different, there is a canonical form to which the identifiers can be transformed such that the same object is recognized as the same.

Additionally, the aliasName has been profiled out of the IdentifiedObject class. Finally, the name attribute has been modified to allow NameType and NameTypeAuthority attribution.

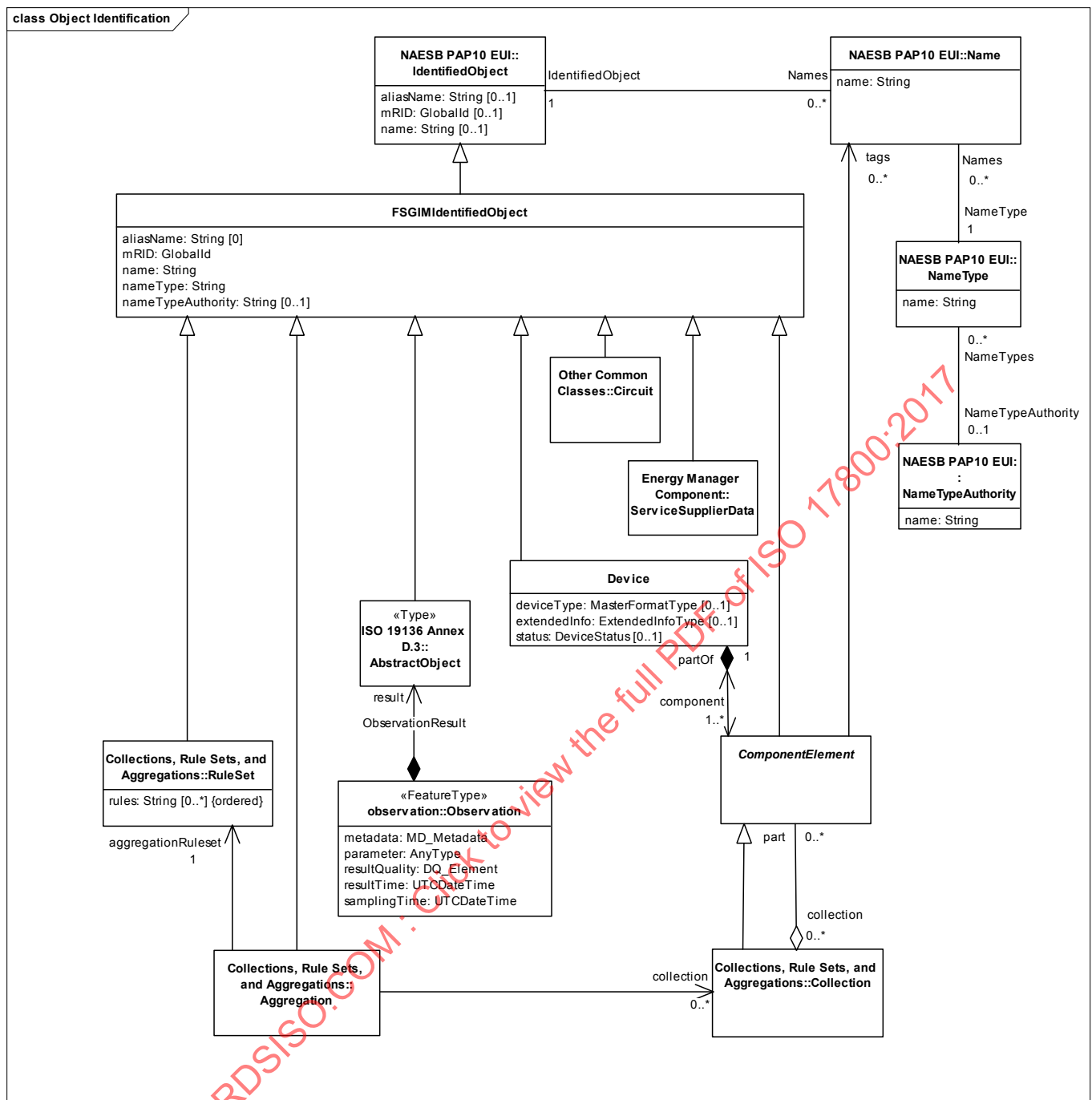


Figure 5.3 - Object Identification

The Object Identification Diagram makes use of the following elements:

AbstractObject (Class)

Abstract object class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractObject*. See Clause 5.7.6.6.3.1.3 for the complete definition of this element.

Aggregation (Class)

An aggregation allows the assembly of statistics of interest arrived at through defined algorithms applied to other parts of the model. The description of an aggregation includes all of the inputs that comprise the algorithms as well as the outputs of the algorithm. An aggregation instance represents one or more aggregateQuantities that are computed by applying aggregation rules for each quantity against a Collection or FilteredCollection of Components.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Aggregation*. See Clause 5.6.7.11.1 for the complete definition of this element.

Circuit (Class)

This class is used to model electrical circuits. All ConnectionPoints that are electrically tied together will point to the same instance of this class.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\Circuit*. See Clause 6.6.5.1 for the complete definition of this element.

Collection (Class)

This class defines a set of entities that is identified as a group because the members share some trait. A Collection can contain other Collections.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Collection*. See Clause 5.6.7.11.3 for the complete definition of this element.

ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*. See Clause 5.2.3.1 for the complete definition of this element.

Device (Class)

This is the class that represents a physical device in the facility

UML Element Location: *Model\Device and Model Components\Device\Device*. See Clause 5.2.3.2 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject*. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Name*. See Clause 5.7.5.2.1.14 for the complete definition of this element.

NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameType*. See Clause 5.7.5.2.1.15 for the complete definition of this element.

NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameTypeAuthority. See Clause 5.7.5.2.1.16 for the complete definition of this element.

Observation (Class)

Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\SWE\OMI_Observation\observation\Observation. See Clause 5.7.6.6.5.1.2 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

ServiceSupplierData (Class)

This class contains information associated with the generated emissions, quality of power, and energy-price units associated with an identified supplier of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\ServiceSupplierData*. See Clause 5.6.7.9 for the complete definition of this element.

5.2.3. Elements Defined in the Device Model

The following elements are defined in this model.

5.2.3.1. ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

Parent Class: FSGIMIdentifiedObject (See Clause 5.2.3.4.)

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*.

This element has the following connections to other elements:

Table 5.1 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Composition (with bi-directional navigation)	partOf	The physical device that houses the functionality of the EM Class, Meter Class, Load Class, and/or Generator Class.	Device See Clause 5.2.3.2.	[1]
Association	tags	This attribute may contain a set of strings to be used at the configuration phase of installation such as "hallway", "3rd Floor", "Critical", ... These strings may be optionally qualified with a NameType and NameTypeAuthority.	Name See Clause 5.7.5.2.1.14.	[0..*]

5.2.3.2. Device (Class)

This is the class that represents a physical device in the facility

Parent Class: FSGIMIdentifiedObject (See Clause 5.2.3.4.)

UML Element Location: *Model\Device and Model Components\Device\Device*.

Table 5.2 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
deviceType	This attribute defines an unique code (as specified in Master Format 2012) that uniquely identifies the device installed.	MasterFormatType See Clause 5.2.3.5.	[0..1]
extendedInfo	This attribute references extended information about the device.	ExtendedInfoType See Clause 5.2.3.3.	[0..1]
status	Operational state of a device within a facility.	DeviceStatus See Clause 5.2.3.6.	[0..1]

This element has the following connections to other elements:

Table 5.3 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Composition	component	The logical building blocks that are combined to represent the physical device.	ComponentElement See Clause 5.2.3.1.	[1..*]
Association	deviceNameplate	The nameplate associated with this device.	DPL See Clause 5.7.4.5.3.1.18.	[0..1]

5.2.3.3. ExtendedInfoType (Class)

This class defines an reference external from this model that can be used to gain additional information about the device.

UML Element Location: *Model\Device and Model Components\Device\ExtendedInfoType*.

Table 5.4 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
ExtendedInfoReference	This attribute defines a string reference to additional information about the device. This reference may be any string reference such as a URL, URL, or other identifying information. This attribute is only the reference to the extended information, it shall not hold the extended information directly.	String See Clause 6.6.3.7.	[1]

5.2.3.4. FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*.

Table 5.5 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
aliasName	This attribute redefines the NAESB IdentifiedObject and sets its multiplicity to 0 to profile it out of the model since aliases are handled through the Name/NameType/NameTypeAuthority construct in the FSGIM. (This attribute redefines the aliasName attribute of the IdentifiedObject Class. See Clause 5.7.5.2.1.8.)	String See Clause 6.6.3.7.	[0]

Attribute Name	Description	Attribute Type	Multiplicity
mRID	Master resource identifier issued by a model authority. The mRID is globally unique within an exchange context. This attribute redefines the mRID attribute of the NAESB IdentifiedObject to force the attribute to be of type GlobalId. (This attribute redefines the mRID attribute of the IdentifiedObject Class. See Clause 5.7.5.2.1.8.)	GlobalId See Clause 6.6.3.4.	[1]
name	Any free text that names the object. This attribute redefines the NAESB IdentifiedObject attribute to force the multiplicity to be 1. (This attribute redefines the name attribute of the IdentifiedObject Class. See Clause 5.7.5.2.1.8.)	String See Clause 6.6.3.7.	[1]
nameType	Name of the name type.	String See Clause 6.6.3.7.	[1]
nameTypeAuthority	Name of the name type authority.	String See Clause 6.6.3.7.	[0..1]

5.2.3.5. MasterFormatType (Class)

This class defines a string attribute with specific constraints to map into the MasterFormat 2012 numbering system.

UML Element Location: *Model\Device and Model Components\Device\MasterFormatType*.

Table 5.6 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
code	This attribute is an eleven character string that defines a coded identifier for the device defined. The identifier is defined in the MasterFormat (2012) Numbers and Titles document. The MasterFormat is a near exhaustive list of building components maintained by the construction industry. Each component has an assigned unique immutable identifier taking the form of NNSNNSNNPXX where: space = ' period = '.' number = '0','1','2','3','4','5','6','7','8','9' N = number S = space P = (period, space) X = (0..9, space) For example, a portable dehumidifier uses code '23 84 13.33'; a room air-conditioner uses code '23 81 16 '. A value of "99 99 99 99" shall be used to model an unknown load.	String See Clause 6.6.3.7.	[1]

5.2.3.6. DeviceStatus (Enumeration)

The general indication of the health of a Device.

UML Element Location: *Model\Device and Model Components\Device\DeviceStatus*.

Table 5.7 - DeviceStatus Enumerated Values

Enumerated Value	Definition
Normal	Device is in a normal operating condition.
Warning	Some type of abnormal condition that does not constitute an alarm has been detected.
Alarm	A serious abnormal condition has been detected.

5.2.3.7. EFacilitySystem (Enumeration)

This enumeration is used to define the subsystem to which the ComponentElement is associated within the facility. ComponentElements that cross subsystems should use the 'general' category.

UML Element Location: *Model\Device and Model Components\Device\EFacilitySystem*.

Table 5.8 - EFacilitySystem Enumerated Values

Enumerated Value	Definition
athletic	The load is associated with athletic equipment including exercise equipment, gymnasium equipment, pool heaters, saunas, steam rooms and scoreboards.
audio Visual	The load is associated with Audio-Visual equipment including projection screens, fixed projection screens, portable projection screens, rear projection screens, projectors, movie projectors, slide projectors, overhead projectors, opaque projectors, video projectors, players, recorders and information centers (kiosks).
banking	The load is associated with a banking subsystem including safe deposit boxes, safes vault ventilators, teller equipment, automatic banking systems, money handling equipment, detention pass-thru doors and depositories.
disposal	The load is associated with a disposal subsystem including packaged incinerators, recycling equipment, facility waste compactors, composting equipment, facility waste shredders, and facility waste balers.
elevators	The load is associated with an elevator or escalator subsystem.
entertainment	The load is associated with an entertainment subsystem including lighting rigging systems, scenery rigging systems, curtain systems, and voice amplifiers.
exterior	The load is associated with exterior equipment including automatic flagpoles, fountains, decorative lighting, pest control devices, deicers and signage.
fire	This load is associated with Fire subsystem including fire panels, smoke detection devices, strobes, evacuation equipment, fire fighter control panels, smoke control and automatic sprinkler systems.
foodservice	The load is associated with Foodservice equipment including refrigerated food storage cases, walk-in coolers, walk-in freezers, food preparation appliances, food preparation surfaces, food delivery carts, food delivery conveyors, commercial ranges, commercial ovens, bar equipment, service line equipment, soda fountain equipment, coffee and espresso equipment, ice machines, disposal equipment and commercial dishwashers.

Enumerated Value	Definition
general	This load is associated to multiple subsystems none of which is primary.
healthcare	The load is associated with healthcare equipment including medical sterilizing equipment, refrigeration units, examination equipment, treatment equipment, patient care equipment, dental equipment, optical equipment, operating room equipment, radiology equipment and mortuary equipment.
hospitality	The load is associated with Hospitality equipment including registration equipment.
HVAC	The load is associated with a heating ventilation and air conditioning (HVAC) subsystem including chillers, ice storage, boilers, electric heaters, heat pumps, air handlers, packaged units, pumps, air distribution systems, air handlers, air compressors, variable air volume controllers, fan coils, humidifiers, dehumidifiers, thermostats, humidistats and facility management systems (FMS).
IT	The load is associated with information technology (IT) equipment including servers, CRAC units, network switches, hubs, routers, firewalls, access points, telephone switches and other MDF/IDF closet equipment.
laboratory	The load is associated with Laboratory equipment including fume hoods, incubators, equipment washers, sterilizers, environment cabinets, emergency safety appliances, service fittings and accessories biological safety cabinets.
laundry	The load is associated with Laundry equipment including dry cleaning equipment; drying and conditioning equipment; finishing equipment; commercial ironing equipment; commercial washers; driers; coin-operated laundry equipment and hanging garment conveyors.
lighting	The load is associated with a lighting subsystem including lighting panels; lighting circuits; floor and task lighting; emergency lighting; lava lamps; and shade control.
maintenance	The load is associated with Maintenance equipment including; floor and wall cleaning equipment; housekeeping carts; vacuum cleaning systems; façade access equipment; window washing systems; welders; drill presses and lathes.
merchandise	The load is associated with Mercantile equipment including cash registers; checking equipment; vending machines; checkroom equipment; weighing and wrapping equipment; barbering and beauty equipment.
office	The load is associated with Office equipment including telephones, computers, printers, copiers, facsimile machines and white noise generators.
pedestrian	The load is associated to a pedestrian equipment, including pedestrian gates, rotary gates turnstiles, money changing machines, fare collection equipment, pedestrian detection equipment, electronic detection and counting systems and pedestrian security equipment.
parking	The load is associated with parking equipment including parking card control units; parking ticket dispensers, parking meters, parking fee collection equipment and parking gates.
production	The load is associated with a production subsystems including assembly line equipment, robots, part picking equipment, stocking equipment, inventory control equipment, assembly, quality testing and production control equipment.
recreational	The load is associated with recreational equipment including game room equipment, video games, and pinball machines.
residential	The load is associated with Residential equipment including lights, bathroom appliances, kitchen appliances, laundry appliances, televisions, stereo equipment and fans.
security	The load is associated with a physical security subsystem including video surveillance, door control, building access, tamper monitoring, terrorist activity monitoring, radio communications, asset tracking, personnel tracking and video analytics.
shipping	The load is associated with Shipping equipment including airport luggage handlers, conveyor belts, packaging, dock levelers, forklifts and postal equipment.
vehicle	The load is associated to a vehicle, including servicing, lubrication, tire changing, battery charging and washing equipment.
other	The load is associated to a subsystem that is not listed in this enumeration.

5.3. Meter Component

The Meter Component is the portion of the FSGIM that provides an abstract representation of the function of measuring power, energy, and emissions.

UML Package Location: *Model\Device and Model Components\Meter Component*

5.3.1. FSGIM Meter Model Diagram

The FSGIM Meter Model incorporates a subset of the full NAESB model for a meter. The subset mirrors the NAESB model from the UsagePoint "down" with two exceptions. (1) It does not include a UsageSummary because the Meter Component in the FSGIM does not retain long-term historical data. (2) It does not include the IntervalBlock because interval blocks are handled by the Energy Manager Component in the FSGIM. In summary, the FSGIM Meter Model translates the NAESB presentation of measurements into the FSGIM presentation of measurements for power, energy, and emissions.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

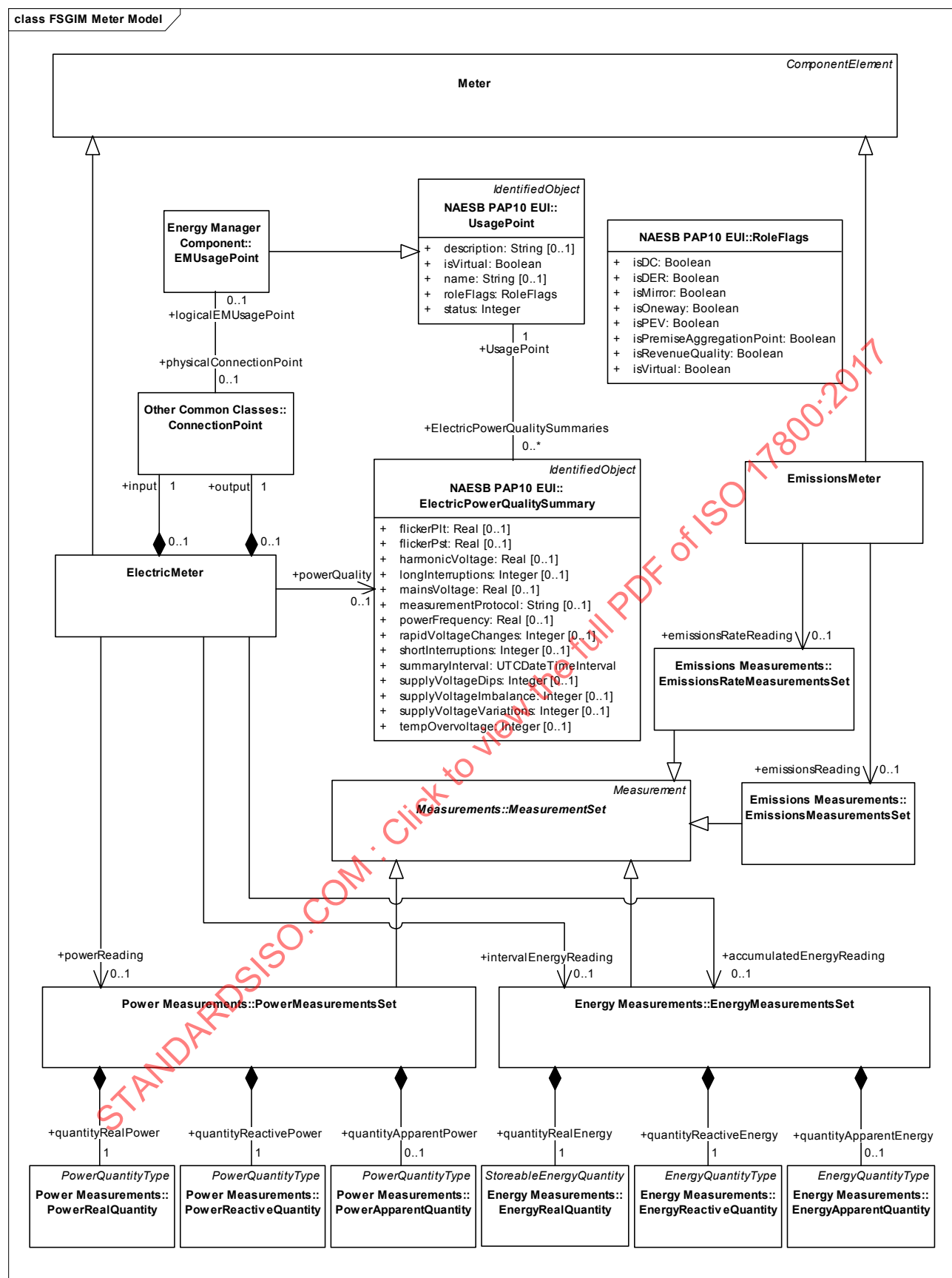


Figure 5.4 - FSGIM Meter Model

The FSGIM Meter Model Diagram makes use of the following elements:

ConnectionPoint (Class)

This class is used to model the physical connections from Generators, Loads, Meters, and Energy Routers to sources or sinks of energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\ConnectionPoint*. See Clause 6.6.5.2 for the complete definition of this element.

ElectricMeter (Class)

The ElectricMeter Class is a subclass of the Meter Class. It is an abstract representation of devices that measure real and reactive power or energy. These devices optionally measure other electrical parameters such as apparent power, apparent energy, present subinterval demand or power quality.

UML Element Location: *Model\Device and Model Components\Meter Component\ElectricMeter*. See Clause 5.3.2.2 for the complete definition of this element.

ElectricPowerQualitySummary (Class)

A summary of power quality events. This information represents a summary of power quality information typically required by energy management systems. It is not intended to satisfy the detailed requirements of power quality monitoring. All values are as defined by measurementProtocol during the period. The standards typically also give ranges of allowed values. See referenced standards for definition, measurement protocol and period.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\ElectricPowerQualitySummary*. See Clause 5.7.5.2.1.5 for the complete definition of this element.

EmissionsMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions quantity measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMeasurementsSet*. See Clause 6.6.4.8.12 for the complete definition of this element.

EmissionsMeter (Class)

The EmissionsMeter Class is a subclass of the Meter Class which is used to represent devices that are capable of measuring pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\EmissionsMeter*. See Clause 5.3.2.3 for the complete definition of this element.

EmissionsRateMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions rate measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsRateMeasurementsSet*. See Clause 6.6.4.8.15 for the complete definition of this element.

EMUsagePoint (Class)

Logical point on a network at which consumption or production is either measured (e.g., metered) or estimated (e.g., unmetered street lights).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMUsagePoint*. See Clause 5.6.7.8 for the complete definition of this element.

EnergyApparentQuantity (Class)

Quantity/metadata pair used to represent apparent energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentQuantity*. See Clause 6.6.4.7.6 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

EnergyReactiveQuantity (Class)

Quantity/metadata pair used to represent reactive energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveQuantity*. See Clause 6.6.4.7.11 for the complete definition of this element.

EnergyRealQuantity (Class)

Quantity/metadata pair used to represent real energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantity*. See Clause 6.6.4.7.13 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

PowerApparentQuantity (Class)

Quantity and pair used to represent apparent power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerApparentQuantity*. See Clause 6.6.4.6.8 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

PowerReactiveQuantity (Class)

Quantity and pair used to represent reactive power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveQuantity*. See Clause 6.6.4.6.13 for the complete definition of this element.

PowerRealQuantity (Class)

Quantity and metadata pair used to represent real power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealQuantity*. See Clause 6.6.4.6.15 for the complete definition of this element.

RoleFlags (Class)

Describes a set of specific communication or physical attributes that the associated UsagePoint has. The term RoleFlags in this specific case pertains to the application-specific nature of the UsagePoint. More than one of these flags can be true for any UsagePoint.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\RoleFlags*. See Clause 5.7.5.2.1.28 for the complete definition of this element.

UsagePoint (Class)

Logical point on a network at which consumption or production is either physically measured (e.g., metered) or estimated (e.g., unmetered street lights).

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\UsagePoint. See Clause 5.7.5.2.1.26 for the complete definition of this element.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

5.3.2. Elements Defined in the Meter Component Model

The following elements are defined in this model.

5.3.2.1. Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

Parent Class: ComponentElement (See Clause 5.2.3.1.)

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*.

5.3.2.2. ElectricMeter (Class)

The ElectricMeter Class is a subclass of the Meter Class. It is an abstract representation of devices that measure real and reactive power or energy. These devices optionally measure other electrical parameters such as apparent power, apparent energy, present subinterval demand or power quality.

Parent Class: Meter (See Clause 5.3.2.1.)

UML Element Location: *Model\Device and Model Components\Meter Component\ElectricMeter*.

This element has the following connections to other elements:

Table 5.9 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	accumulatedEnergyReading	The accumulated energy measurement set for a given location at the timestamp defined by Measurement.timeReference.	EnergyMeasurementsSet See Clause 6.6.4.7.9.	[0..1]
Composition	input	The connection point associated with the input of this ElectricMeter. The input side of a Meter is defined as the connection point nearer to the GridCircuit.	ConnectionPoint See Clause 6.6.5.2.	[1]
Association	intervalEnergyReading	The energy measurement set for a given location over the interval defined by Measurement.timeReference.	EnergyMeasurementsSet See Clause 6.6.4.7.9.	[0..1]
Composition	output	The connection point associated with the output of this ElectricMeter. The output side of a Meter is defined as the connection point farther from the GridCircuit.	ConnectionPoint See Clause 6.6.5.2.	[1]
Association	powerQuality	The power quality statistical data set for a given location.	ElectricPowerQualitySummary See Clause 5.7.5.2.1.5.	[0..1]
Association	powerReading	The power measurement set for a given location.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[0..1]

5.3.2.3. EmissionsMeter (Class)

The EmissionsMeter Class is a subclass of the Meter Class which is used to represent devices that are capable of measuring pollution emissions.
Constraint: An EmissionsMeter must have either an emissionsRateReading or an emissionsRateReading and may have both.

Parent Class: Meter (See Clause 5.3.2.1.)

UML Element Location: *Model\Device and Model Components\Meter Component\EmissionsMeter*.
This element has the following connections to other elements:

Table 5.10 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	emissionsRateReading	The emissions rate measurement set for a given location.	EmissionsRateMeasurementsSet See Clause 6.6.4.8.15.	[0..1]
Association	emissionsReading	The emissions measurement set for a given location.	EmissionsMeasurementsSet See Clause 6.6.4.8.12.	[0..1]

5.4. Load Component

The Load Component is the portion of the FSGIM that provides an abstract representation of all devices that consume electricity. This includes devices powered from the electrical grid, powered from an alternate source, or battery powered devices. Some loads can only be switched on and off. Other loads can be modified to manage peaks or in response to pricing or other grid interactions.

The Load Component contains classes that support the ability to forecast demand, energy consumption, or both. It also contains classes that support some types of protection to prevent damage of physical equipment, however real devices may need additional safety constraints that are not captured in the model.

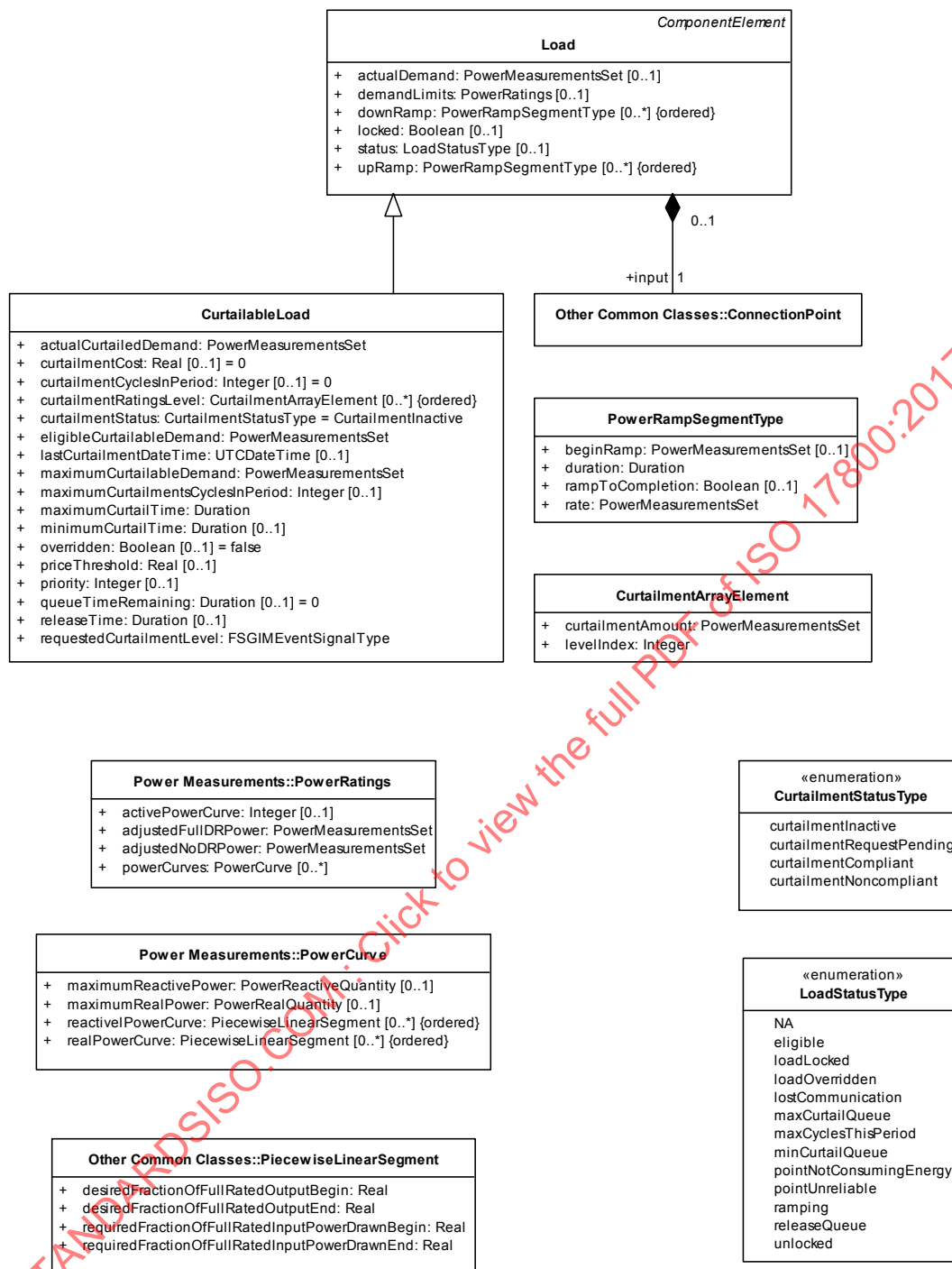
UML Package Location: *Model\Device and Model Components\Load Component*

5.4.1. Load Classes and Elements Diagram

The Load Classes and Elements figure describes the Load and CurtailableLoad Classes. All devices that use electricity will support the Load class. Those that can be curtailed will also support the CurtailableLoad class.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

class Load Classes and Elements



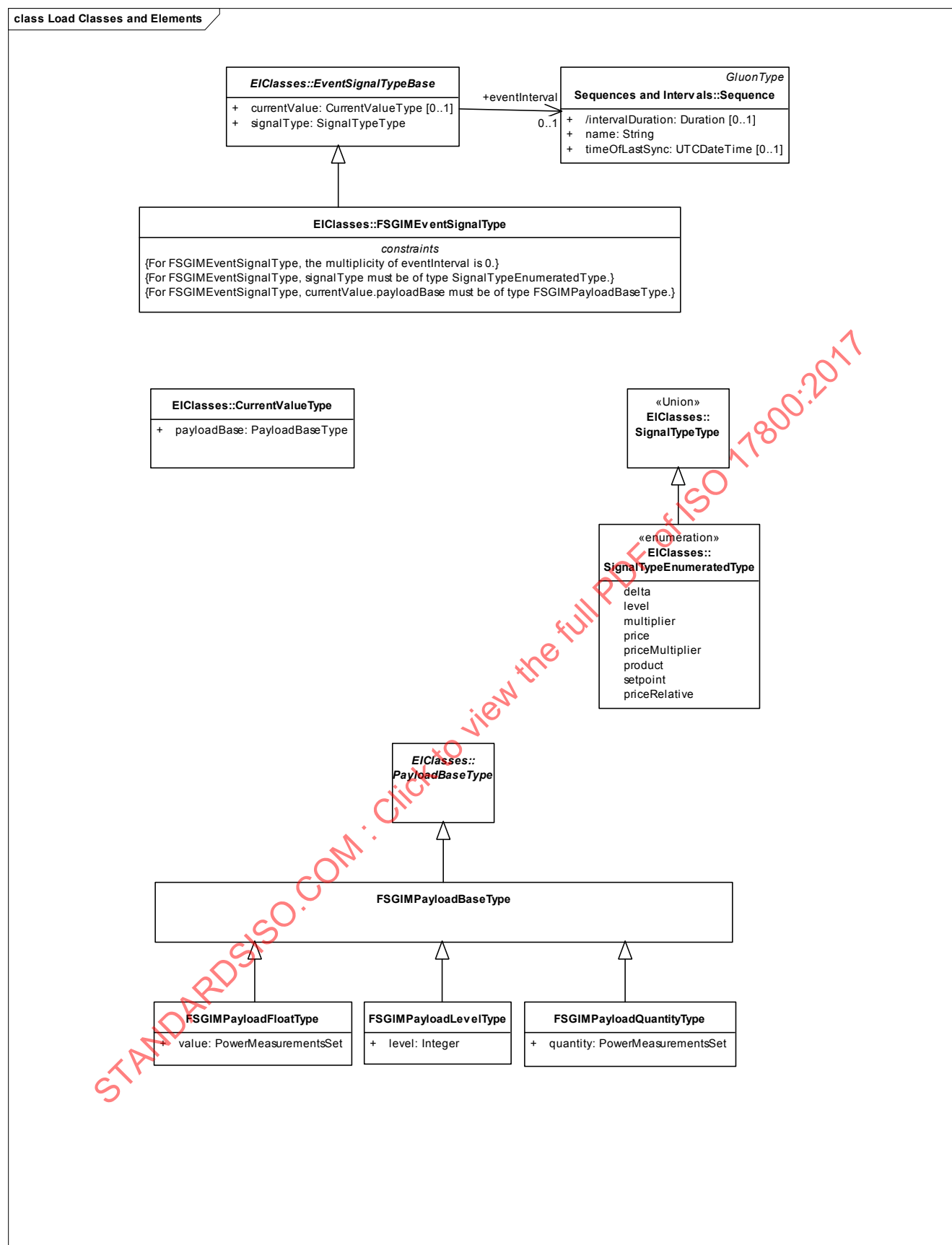


Figure 5.5 - Load Classes and Elements

The Load Classes and Elements Diagram makes use of the following elements:

ConnectionPoint (Class)

This class is used to model the physical connections from Generators, Loads, Meters, and Energy Routers to sources or sinks of energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\ConnectionPoint*. See Clause 6.6.5.2 for the complete definition of this element.

CurrentValueType (Class)

This class contains the common information, at the time of an event, in an Energy Interoperation signal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\CurrentValueType*. See Clause 5.7.1.2.1.5 for the complete definition of this element.

CurtableLoad (Class)

The CurtableLoad class defines the attributes that are unique to devices whose electrical consumption can be curtailed. The attributes define the information needed to represent curtailment policies based on criteria including load priorities, cost constraints, protection of physical equipment, and safety considerations. The CurtableLoad class also provides attributes for restoration of curtailed loads.

UML Element Location: *Model\Device and Model Components\Load Component\CurtableLoad*. See Clause 5.4.2.2 for the complete definition of this element.

CurtablementArrayElement (Class)

This class defines a single array element of the curtablementRatingsLevel attribute. Each array element defines the fixed demand to which the load shall be controlled as a function of the stated level. The array is ordered by the curtablementRatingsLevel.levelIndex attribute.

UML Element Location: *Model\Device and Model Components\Load Component\CurtablementArrayElement*. See Clause 5.4.2.3 for the complete definition of this element.

CurtablementStatusType (Enumeration)

This enumeration indicates if the load is presently curtailed and if curtailed is in compliance with the curtablement request received.

UML Element Location: *Model\Device and Model Components\Load Component\CurtablementStatusType*. See Clause 5.4.2.10 for the complete definition of this element.

EventSignalTypeBase (Abstract Class)

This class conveys the kind of demand response event being signaled and the desired resulting load or generation.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventSignalTypeBase*. See Clause 5.7.1.2.1.13 for the complete definition of this element.

FSGIMEventSignalType (Class)

This class conveys the kind of demand response event being signaled from an energy manager to a curtable load or a dispatchable generator. The signal indicates the desired resulting load or generation.

For FSGIMEventSignalType, the multiplicity of eventInterval is 0.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\FSGIMEventSignalType*. See Clause 5.7.1.2.1.15 for the complete definition of this element.

FSGIMPayloadBaseType (Class)

This class is the union of the different classes used to define the needed curtablement of an individual load.

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadBaseType*. See Clause 5.4.2.9 for the complete definition of this element.

FSGIMPayloadFloatType (Class)

The curtablement request is of the form of either an absolute or relative demand curtablement value from the present baseline demand. This class requires the use of a 'setpoint' or 'delta' signal type. This class is based on the EIClasses::PayloadFloatType Class but includes metadata that the Energy Interoperation model captures elsewhere.

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadFloatType*. See Clause 5.4.2.6 for the complete definition of this element.

FSGIMPayloadLevelType (Class)

The curtailment request is of the form of a predefined level of curtailment. This curtailment level's range is specified by the array size of the curtailmentRatingsLevelIndex attribute. The value of the integer requestedCurtailmentLevel.curtailmentValue.payloadBase.level is the index into the curtailmentRatingsLevel array. This class supports the 'level' signal type. This class is identical to the EIClasses::PayloadLevelType Class.

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadLevelType*. See Clause 5.4.2.7 for the complete definition of this element.

FSGIMPayloadQuantityType (Class)

The curtailment request is in the form of a percentage of the present baseline demand. This class requires the use of a 'multiplier' signal type. This class is identical to the EIClasses::PayloadQuantityType Class.

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadQuantityType*. See Clause 5.4.2.8 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

LoadStatusType (Enumeration)

This enumeration provides the present overall state of the load.

UML Element Location: *Model\Device and Model Components\Load Component\LoadStatusType*. See Clause 5.4.2.11 for the complete definition of this element.

PayloadBaseType (Abstract Class)

base for information in Signal / Baseline / Report Payloads

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadBaseType*. See Clause 5.7.1.2.1.17 for the complete definition of this element.

PiecewiseLinearSegment (Class)

The PiecewiseLinearSegment class defines the attributes needed to specify a single straight line segment for a piecewise linear curve. Each straight line segment is specified by two X-axis coordinates, percentOFFullRatedOutputBegin and percentOFFullRatedOutputEnd; and by two Y-axis coordinates, percentOFFullRatedInputPowerDrawnBegin and percentOFFullRatedInputPowerDrawnEnd.

The entire piecewise linear curve is defined by the runningProfile attribute; where the 'percent of full rated input power' is a function of the 'percent of full rated output'. That is, as the output varies between 0 and 100 percent; the function maps to the percentage of input power (0..100) required to achieve the specified output.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\PiecewiseLinearSegment*. See Clause 6.6.5.5 for the complete definition of this element.

PowerCurve (Class)

This class describes the characteristics of a mathematical function used to estimate the power consuming characteristics of a load or the power generating characteristics of a generator.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerCurve*. See Clause 6.6.4.6.2 for the complete definition of this element.

PowerRampSegmentType (Class)

The PowerRampSegmentType data structure is used to define a single array element of the recoveryRamp and stagingRamp array of the Load class. Each array element defines the beginning demand for the line segment and the rate

of rise or drop. These attributes combined with the duration completely forms a line segment defining a portion of the ramp.

UML Element Location: *Model\Device and Model Components\Load Component\PowerRampSegmentType*. See Clause 5.4.2.4 for the complete definition of this element.

PowerRatings (Class)

This class describes the power characteristics of a Load (or Generator) component. The attributes defined allow specifying the minimum and maximum expected power draw from the load (supply from a generator). It also allows a series of predefined operation power curves to be defined with one designated as presently being operational.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRatings*. See Clause 6.6.4.6.4 for the complete definition of this element.

Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be surmised by monitoring the time references.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*. See Clause 5.6.7.10.4 for the complete definition of this element.

SignalTypeEnumeratedType (Enumeration)

SignalTypeEnumerated lists the pre-defined Types used to specify the Payload Types and conformance in a Stream

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SignalTypeEnumeratedType*. See Clause 5.7.1.2.1.34 for the complete definition of this element.

SignalTypeType (Class)

SignalType is used in EventSignals to specify the Payload Types in a Signal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SignalTypeType*. See Clause 5.7.1.2.1.29 for the complete definition of this element.

5.4.2. Elements Defined in the Load Component Model

The following elements are defined in this model.

5.4.2.1. CurtailmentValueType (Class)

This class contains the signal type and curtailment information requested by an energy manager during an event. It provides a 'union' of the various signaling types and curtailment information into a single attribute.

UML Element Location: *Model\Device and Model Components\Load Component\CurtailmentValueType*.

Table 5.11 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
payloadBase	This attribute provides the means to 'union' the differing curtailmentValues into a single attribute.	PayloadBaseType See Clause 5.7.1.2.1.17.	[1]

5.4.2.2. CurtailableLoad (Class)

The CurtailableLoad class defines the attributes that are unique to devices whose electrical consumption can be curtailed. The attributes define the information needed to represent curtailment policies based on criteria including load priorities, cost constraints, protection of physical equipment, and safety considerations. The CurtailableLoad class also provides attributes for restoration of curtailed loads.

Parent Class: Load (See Clause 5.4.2.5.)

UML Element Location: *Model\Device and Model Components\Load Component\CurtailableLoad*.

Table 5.12 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
actualCurtailedDemand	This attribute defines the amount of demand presently curtailed for this load (as measured by the present subinterval demand) as measured within the PowerMeasurementsSet for this load. Constraint: This attribute's value shall be greater than or equal to 0.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
curtailmentCost	This attribute defines the monetary value of the fixed and variable cost associated with curtailing the load (when positive) or the savings associated by curtailing the load (when negative) on a per unit of power basis.	Real See Clause 6.6.3.6.	[0..1]
curtailmentCyclesInPeriod	This attribute defines the number of times the load has been curtailed since this attribute was last reset. This attribute shall be increased by one each time a two-state load is curtailed, or a multi-state load is curtailed from one curtailment state to another. Constraint: If the maximumCurtailmentsInPeriod attribute is defined, then this attribute shall be defined. Constraint: This attribute's value shall be greater than or equal to 0.	Integer See Clause 6.6.3.5.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
curtailmentRatingsLevel	This attribute defines the array of curtailment amounts when the requestedCurtailmentLevel.signalType equals 'level'. This array transforms the level requested by the Energy Manager as specified by the attribute requestedCurtailmentLevel.level into the fixed demand to which the load shall be controlled. Constraint: This attribute is required if requestedCurtailmentLevel.signalType may equal 'level'	CurtailmentArrayElement See Clause 5.4.2.3.	[0..*]
curtailmentStatus	This attribute indicates whether the load is presently curtailed and, if curtailed, is in compliance with the curtailment request received.	CurtailmentStatusType See Clause 5.4.2.10.	[1]
eligibleCurtailableDemand	This attribute defines the amount of load still eligible to be curtailed for two-state and multi-state curtailable loads. For two-state loads this value will either be the value of maximumLoad as defined in the Load class or zero depending on whether the load is presently curtailed. For multi-state curtailable loads this value is the total remaining load that can be curtailed as defined in the curtailmentRatings attribute. Constraint: The value of this attribute shall be greater or equal to 0. Constraint: The value of this attribute shall not be greater than the value of the Load.maximumLoad attribute.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
lastCurtailmentDateTime	This attribute defines the date and time on which the load was last curtailed. For multistage curtailable loads, each time a multistage curtailable load is curtailed, this attribute shall be updated.	UTCDateTime See Clause 6.6.1.11.	[0..1]
maximumCurtailableDemand	This attribute defines the maximum amount of demand curtailable for this load as measured in PowerMeasurementsSet. Constraint: This attribute's value shall be greater than or equal to 0.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
maximumCurtailmentsCyclesInPeriod	This attribute defines the maximum number of times this load may be curtailed within the specified period. When the curtailmentCyclesInPeriod reaches this attribute's value, the loadStatus attribute will be set to maxCyclesThisPeriod rendering the load ineligible for curtailment until the period information is reset. The duration of the period and the resetting of this attribute is a local matter. Constraint: If the actualCurtailmentsInPeriod attribute is defined, then this attribute shall be defined. Constraint: This attribute's value shall be greater than or equal to 0.	Integer See Clause 6.6.3.5.	[0..1]
maximumCurtailTime	This attribute specifies the maximum number of seconds that a load may be curtailed before it shall be released from its curtailment state. This duration is often used to maintain comfort levels, or meet safety code requirements. It is used to set the queueTimeRemaining attribute when placing the load in the maxCurtailQueue state. Constraint: This attribute's value shall be greater than 0. Constraint: This attribute's value shall be greater than or equal to the minimumCurtailTime attribute's value.	Duration See Clause 6.6.1.9.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
minimumCurtailTime	This attribute specifies the minimum number of seconds that a load shall be curtailed before it may be released from its curtailment state. This duration is used to protect the load's hardware from excess cycling. However, it may also be used to maintain comfort levels, or meet safety code requirements. It is used to set the QueueTimeRemaining attribute when placing the load in the minCurtailQueue state. Constraint: This attribute's value shall be greater than 0. Constraint: This attribute's value shall be less than or equal to the maximumCurtailTime attribute's value.	Duration See Clause 6.6.1.9.	[0..1]
overridden	This attribute is typically controlled by an external entity to affect eligibility of the load. For example, an Energy Manager may override this load if the present price of energy is falls within a defined economic threshold. If the present price of energy exceeds the threshold, the Energy Manager might toggle this attribute to allow the load to be curtailed. This attribute differs from the loadLocked attribute in that this attribute is meant to be connected to an automation system and only persists while the override is in effect. The loadLocked attribute persists indefinitely and meant to be set by manual intervention.	Boolean See Clause 6.6.3.3.	[0..1]
priceThreshold	This attribute defines the marginal operational cost per unit of power basis for this load. The priority is not created within the load; this is a placeholder for the Energy Manager. Constraint: This attribute's value shall be greater than or equal to 0.	Real See Clause 6.6.3.6.	[0..1]
priority	This attribute defines the curtailment priority which defines the order upon which curtailment occurs across a set of loads. A priority value of 1 is considered a critical load that may not be shed except during a critical event. Constraint: How eligible loads of the same priority curtail is a local matter. Constraint: Priorities range from 1 to 1000 Constraint: Priority 1 loads curtail last Constraint: Priority 1000 loads curtail earliest	Integer See Clause 6.6.3.5.	[0..1]
queueTimeRemaining	This attribute specifies the duration that a controlled load has been on a queue. As the load transitions from the minCurtailQueue to the maxCurtailQueue and the releaseQueue this attribute is set to its maximum extent upon entry in the queue and counts down to zero on the queue when it transitions off the queue. This attribute should be updated at least once each 60 seconds. Constraint: actualQueueTimeRemaining >= 0	Duration See Clause 6.6.1.9.	[0..1]
releaseTime	This attribute specifies the number of seconds that a load shall be released from curtailment before it is again eligible for curtailment. This duration is used to protect the load's hardware from excess cycling. However, it may also be used to restore comfort levels, or meet safety code requirements. It is used to set the queueTimeRemaining attribute when placing the load in the releaseQueue state. If this attribute is not present, the releaseTime shall be considered to be zero (0) and	Duration See Clause 6.6.1.9.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
requestedCurtailmentLevel	immediately become available for curtailment. Constraint: This attribute's value shall be greater than 0. This attribute defines the demand level the Energy Manager is requesting the load to operate. The request by the Energy Manager is always a curtailment amount; however, the load will compare this curtailment requested amount to the present demand of the load and may then a) curtail it more, b) restore it more or c) leave the load at its present demand state. The request is made in one of four ways defined by the value requestedCurtailmentLevel.signalType : 1) When the requestedCurtailmentLevel.signalType attribute is set to 'setpoint', the requestedCurtailmentLevel.currentValue describes the fixed amount of demand to which the load should operate. currentValue must be a positive number or zero. The currentValue is compared against the actualCurtailableAmount and maximumCurtailmentAmount to determine the curtailment or restoration needed to meet the request. a) If currentValue equals the actualCurtailedAmount, no further curtailment or restoration will occur. The curtailmentStatus will be set to curtailmentCompliant. b) If currentValue is less than the actualCurtailedAmount, then if the load has surpassed its minCurtailTime but has not yet reached the maxCurtailTime, the load will be considered for early release. The load may be released in its entirety for a two-state load or partially for a multistate load until the actualCurtailedAmount reaches currentValue's value. If the load cannot be released to exactly match the request, the load will be released as close as possible as it can be released without exceeding the request. The curtailmentStatus will remain as curtailmentCompliant. c) If currentValue is greater than the actualCurtailedAmount, the load may be further curtailed as possible upwards to the maximumCurtailmentAmount. The curtailmentStatus attribute will then be adjusted to reflect whether the curtailment is curtailmentCompliant or curtailmentNoncompliant. d) If currentValue is greater than the maximumCurtailableAmount, the load will be completely curtailed and the curtailmentStatus will be set to curtailmentNoncompliant. Stated mathematically, assure $0 \leq \text{currentValue} \leq \text{maximumCurtailmentAmount}$ And if $\text{currentValue} \geq \text{actualCurtailableAmount}$; amount to curtail = $\text{currentValue} - \text{actualCurtailableAmount}$ else if	FSGIMEventSignalType e See Clause 5.7.1.2.1.15.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
	currentValue < actualCurtailableAmount; amount to restore = actualCurtailableAmount - currentValue. 2) When the requestedCurtailmentLevel.signalType attribute is set to 'delta', the requestedCurtailmentLevel.currentValue attribute describes the relative amount of demand to which the load should be adjusted. currentValue may be a positive or negative number. The sum of currentValue and actualCurtailableAmount, is compared against zero and maximumCurtailmentAmount to determine if curtailment or restoration is needed to meet the request. a) If the value of currentValue is zero no further curtailment or restoration will occur. The curtailmentStatus will be set to curtailmentCompliant. b) If currentValue is positive, the load will be further curtailed as possible upwards to the maximumCurtailmentAmount. If the sum of currentValue and actualCurtailableAmount is greater than maximumCurtailmentAmount, the load will be completely curtailed and the curtailmentStatus will be set to curtailmentNoncompliant otherwise it will be set to curtailment Compliant. c) If currentValue is negative, then if the load has surpassed its minCurtailTime but has not yet reached the maxCurtailTime, the load will be considered for early release. The load may be released in its entirety for a two-state load or partially for a multistate load until the currentValue value is reached or actualCurtailableAmount is zero. If the sum of currentValue and actualCurtailableAmount is less than zero, the load will be completely released and the curtailmentStatus will be set to curtailmentNoncompliant otherwise it will be set to curtailment Compliant. Stated mathematically, assure 0 <= currentValue+ actualCurtailableAmount <= maximumCurtailmentAmount And if currentValue >= 0, amount to curtail = currentValue else if currentValue < 0, amount to restore = currentValue. 3) When the requestedCurtailmentLevel.signalType attribute is set to 'multiplier', the requestedCurtailmentLevel.currentValue attribute divided by 100 describes the percentage of the maximumCurtailableAmount value to which the load should be adjusted. currentValue must be a positive number in the range of 0 to 100. The calculated value of product of currentValue/100 and maximumCurtailmentAmount is compared to the actualCurtailableAmount to determine the curtailment or restoration needed to meet the		

Attribute Name	Description	Attribute Type	Multiplicity
	request. a) If this calculated value is equal to actualCurtailedAmount, then no further curtailment or restoration will occur. The curtailmentStatus will be set to curtailmentCompliant. b) If this calculated value is less than the actualCurtailedAmount, then if the load has surpassed its minCurtailTime but has not yet reached the maxCurtailTime the load will be considered for early release. The load may be released in its entirety for a two-state load or partially for a multistate load until the actualCurtailedAmount reaches the calculated value. The curtailmentStatus will remain as curtailmentCompliant. c) If the calculated value is greater than the actualCurtailedAmount, the load will be further curtailed as possible upwards to the maximumCurtailmentAmount. The curtailmentStatus attribute will then be adjusted to reflect whether the curtailment is curtailmentCompliant or curtailmentNoncompliant. Stated mathematically, If $(\text{currentValue}/100) * \text{maximumCurtailmentAmount} \geq \text{actualCurtailableAmount; amount to curtail} = ((\text{currentValue}/100) * \text{maximumCurtailmentAmount}) - \text{actualCurtailableAmount}$ else if $(\text{currentValue}/100) * \text{maximumCurtailmentAmount} < \text{actualCurtailableAmount; amount to release} = \text{actualCurtailableAmount} - ((\text{currentValue}/100) * \text{maximumCurtailmentAmount})$ 4) When the requestedCurtailmentLevel.signalType attribute is set to 'level', requestedCurtailmentLevel.currentValue attribute describes the predefined level to which the load will be adjusted. currentValue must be a positive number within the range of 1 to n, where n is defined as the number of entries in the curtailmentRatingsLevel array. When the request is received, the load will adjust its curtailment to the level defined by curtailmentRatingsLevel(currentValue).curtailmentAmount a) If curtailmentAmount equals the actualCurtailedAmount, no further curtailment or restoration will occur. The curtailmentStatus will be set to curtailmentCompliant. b) If curtailmentAmount is less than the actualCurtailedAmount, then if the load has surpassed its minCurtailTime but has not yet reached the maxCurtailTime, the load will be considered for early release. The load may be released in its entirety for a two-state load or partially for a multistate load until the actualCurtailedAmount reaches curtailmentAmount's value. The curtailmentStatus will remain as curtailmentCompliant. c) If curtailmentAmount is greater than the actualCurtailedAmount, the load may be further curtailed as possible upwards to the maximumCurtailmentAmount. The curtailmentStatus attribute will then be adjusted to reflect whether the curtailment is curtailmentCompliant or curtailmentNoncompliant.		

Attribute Name	Description	Attribute Type	Multiplicity
	Stated mathematically, if curtailmentRatingsLevel(currentValue).curtailmentAmount >= actualCurtailableAmount; amount to curtail = curtailmentRatingsLevel(currentValue). - actualCurtailableAmount else if curtailmentRatingsLevel(currentValue).curtailmentAmount < actualCurtailableAmount; amount to release = actualCurtailableAmount -curtailmentRatingsLevel(currentValue). NOTE: A load that may be curtailed multiple times can be curtailed only when in the eligible, minCurtailQueue or maxCurtailQueue states. Upon additional curtailments, the curtailment timers may or may not be reset as a local matter.		

5.4.2.2.1. Curtailment State Machine Diagram

The Load Curtailment State Machine defines the states a load transcends as it is curtailed and restored. The load must first pass all eligibility requirements before being a curtailment candidate. Once curtailed, it must transition through both the curtailment and release time queues.

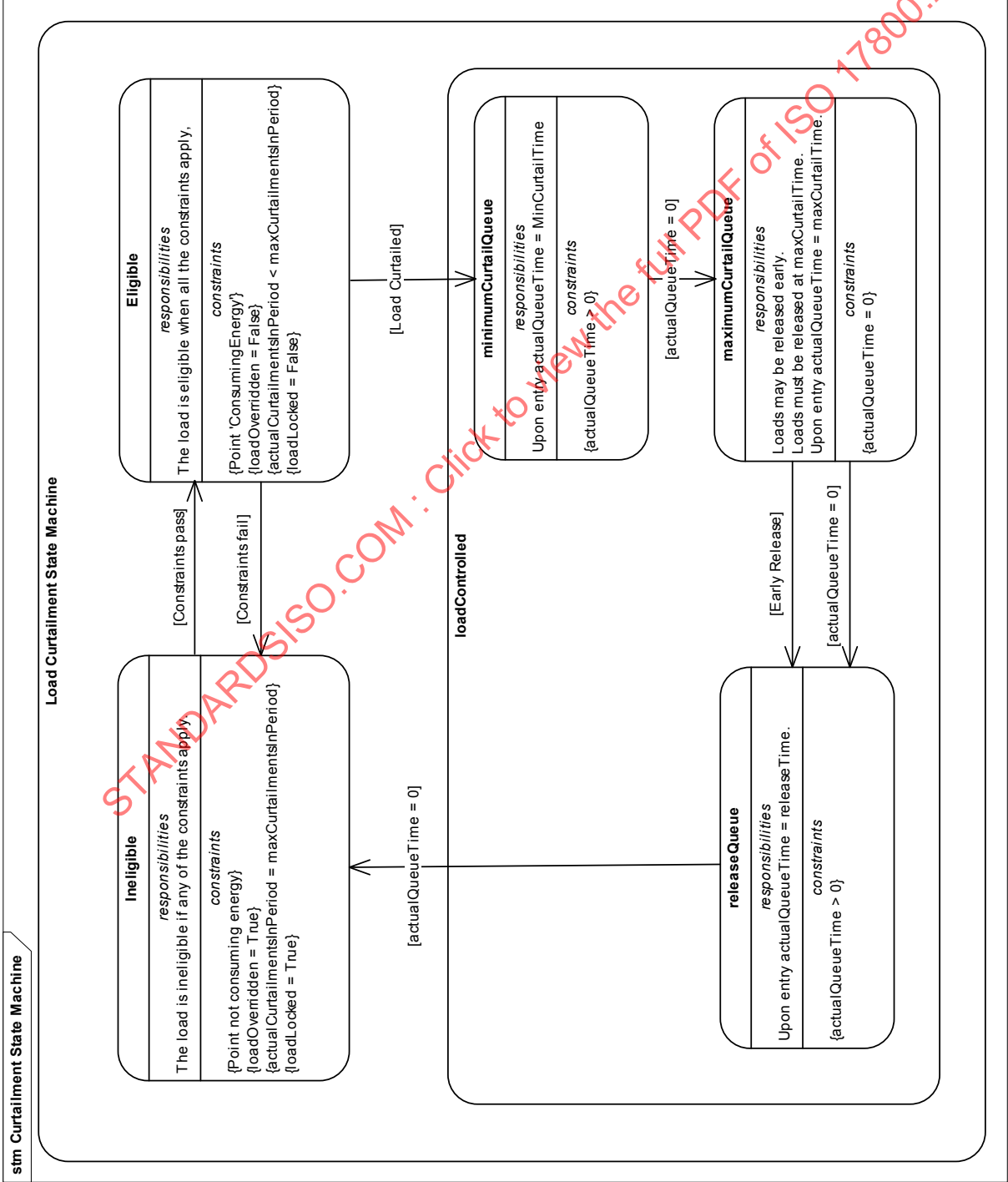


Figure 5.6 – Curtailment State Machine

The Curtailment State Machine Diagram makes use of the following elements:

Eligible (State)

Loads in the eligible state have passed all curtailment criteria and can be curtailed by the energy manager if needed. A periodic test of the pertinent control attributes verifies that the real-time status of the attributes allows the load to be curtailed. If the eligibility test fails, the state transitions to the 'Ineligible' state.

A load is eligible for curtailment when all of the following conditions occur.

- 1) The point controlled by the load is presently consuming energy. That is, the device being controlled will save energy if controlled to a reduced state.
- 2) The load is not overridden (CurtableLoad.loadOverridden = False), meaning that there are no higher priority applications presently controlling this device.
- 3) The load is unlocked (CurtableLoad.loadLocked = False), meaning that the user has not manually deactivated this load.
- 4) The maximum curtailments allotted for this load in this period has not been achieved (CurtableLoad.actualCurtailmentsInPeriod < CurtableLoad.maximumCurtailmentsInPeriod).
- 5) The point controlled by the load indicates that its present value is reliable. ~~Reliability~~ is a meta-state of a point that indicates the point's present value is valid or suspect (due to either hardware or software problems).
- 6) The multi-shed load has previously been curtailed and is on the 'Shed Queue'. It is a local matter whether the load may immediately be re-shed or if it must timeout prior to being re-shed.

When all the aforementioned conditions are met the Load.loadStatus state shall be set to 'eligible'.

A load becomes ineligible for curtailment and will return to the 'Ineligible' state whenever any of the following conditions occur.

- 1) The point controlled by the load is no longer consuming energy. The load status attribute shall be set to indicate this state (CurtableLoad.loadStatus = PointAlreadyShutoff).
- 2) The load becomes locked, (CurtableLoad.loadLocked = TRUE). The load status attribute shall be set to indicate this state (CurtableLoad.loadStatus = Locked).
- 3) If this optional attribute is defined and the load becomes overridden, (CurtableLoad.loadOverridden = TRUE). The load status attribute shall be set to indicate this state (CurtableLoad.loadStatus = OverrideInEffect).

A load may also be ineligible for curtailment whenever any of the following apply:

- 4) The point controlled by the load indicates that its present value is not reliable. The load status attribute shall be set to indicate this state (CurtableLoad.loadStatus = PointUnreliable).
- 5) The point controlled by the load indicates that it is presently under control by an application that is at a higher priority; hence the requested control action may not take effect.

- 6) The multi-shed load has previously been curtailed and is on the 'minimumCurtailedQueue' or 'maximumCurtailedQueue'. It is a local matter whether the load may immediately be re-shed or if it must timeout prior to being re-shed.

Ineligible (State)

Loads in this state cannot be curtailed at the present time because of a failure in passing one or more of the load curtailment eligibility tests. A periodic test of the pertinent control attributes determines if the load is eligible for curtailment. If the eligibility tests succeed, the state transitions to the 'Eligible' state. If not, the tests are repeated periodically to determine if the load has changed its state.

A load is ineligible for curtailment whenever any of the following apply:

- 1) The point controlled by the load is presently not consuming energy. A load cannot save energy if it is not already consuming energy. The load status attribute shall be set to indicate this state (CurtailedLoad.loadStatus = PointNotConsumingEnergy).
- 2) The load is locked, (CurtailedLoad.loadLocked = TRUE). This attribute may be set by the user for any reason. The load status attribute shall be set to indicate this state (CurtailedLoad.loadStatus = Locked).
- 3) If this optional attribute is defined and the load is presently overridden, (CurtailedLoad.loadOverridden = TRUE); the load status attribute shall be set to indicate this state (CurtailedLoad.loadStatus = OverrideInEffect).
- 4) The load has already been curtailed and is on the 'Shed Queue', (CurtailedLoad.curtailmentState = MinCurtailedTime OR CurtailedLoad.curtailmentState = MaxCurtailedTime). The load status attribute shall be set to indicate this state (CurtailedLoad.loadStatus = MinCurtailedQueue) or (CurtailedLoad.loadStatus = MaxCurtailedQueue) respectively.
- 5) The load has already been curtailed and is on the 'Release Queue', (CurtailedLoad.curtailmentState = MinReleaseTime). The load status attribute shall be set to indicate this state (CurtailedLoad.loadStatus = ReleaseQueue).
- 6) If this optional attribute is defined and the load has been curtailed the maximum number of times this period, (CurtailedLoad.actualCurtailmentsInPeriod = CurtailedLoad.maxCurtailmentsInPeriod). The load status attribute shall be set to indicate this state (CurtailedLoad.loadStatus = MaxCurtailmentsReached).

A load may also be ineligible for curtailment whenever any of the following apply:

- 7) The point controlled by the load indicates that its present value is not reliable. The load status attribute shall be set to indicate this state (CurtailedLoad.loadStatus = PointUnreliable).
- 8) The point controlled by the load indicates that it is presently under control by an application that is at a higher priority; hence the requested control action may not take effect.
- 9) The multi-shed load has previously been curtailed and is on the 'Shed Queue'. It is a local matter whether the load may immediately be re-shed or if it must timeout prior to being re-shed.

loadControlled (State)

Loads may be in one of three mutually exclusive states a) eligible, b) ineligible or c) load controlled. When a load is eligible, it has passed all its tests and can be shed at any time. When a load is ineligible, it has failed at least one test and cannot be shed until the failed test is rectified.

Loads that are presently controlled are on a queue and temporarily cannot be shed until they time off of their queues.

maximumCurtailQueue (State)

Once a curtailed load has remained on the minCurtailQueue for the minimum required time, it may be released or may continue to be curtailed. the algorithm used to determine the early restoration of curtailed loads is a local matter.

At the time the load reaches its maximum shed time, the load must be released and placed in the release queue. It is not a candidate to be shed again until it completes its time on the release queue.

minimumCurtailQueue (State)

When an eligible load is selected for curtailment by the EM, the load will be curtailed. When curtailed, the load must reside on the shed queue (CurtailableLoad.curtailmentState = minCurtailTime) for the minimum shed time (CurtailableLoad.minimumCurtailTime). Once reached, the load may remain on the shed queue additional time as conditions dictate, but no longer than the 'maximum shed time' (CurtailableLoad.maximumCurtailTime).

At the time the load reaches its minimum or maximum shed time, the load will be released and placed in the release queue. It is not a candidate to be shed again until it completes its time on the release queue.

releaseQueue (State)

The load has been released and must stay in this state (CurtailableLoad.curtailmentState = minReleaseTime) for its specified time before being considered to be eligible for shed again.

5.4.2.3. CurtailmentArrayElement (Class)

This class defines a single array element of the curtailmentRatingsLevel attribute. Each array element defines the fixed demand to which the load shall be controlled as a function of the stated level. The array is ordered by the curtailmentRatingsLevel.levelIndex attribute.

UML Element Location: *Model\Device and Model Components\Load Component\CurtailmentArrayElement*.

Table 5.13 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
curtailmentAmount	This attribute defines the absolute demand to be achieved upon receipt of a request from an energy manager as specified in the requestedCurtailmentLevel.currentValue attribute. When deployed in the curtailmentRatingsLevel array, this attribute defines the absolute demand the load should target for each of the defined levels. When deployed in the curtailmentRatingsPercent array, this attribute defines the absolute demand the load should target for each of the defined percents.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
levelIndex	This attribute defines the level or percentage index when used in the curtailmentRatingsLevel or curtailmentRatingsMultiplier attributes respectively. When	Integer See Clause 6.6.3.5.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
	used in the curtailmentRatingsLevel array , this attribute is matched to the requestedCurtailmentLevel.currentValue as supplied by the Energy Manager to transform the level request to the absolute demand requested. When used in the curtailmentRatingsMultiplier array, this attribute is matched to the requestedCurtailmentLevel.currentValue as supplied by the Energy Manager to transform the multiplier (percentage) request to the absolute demand requested.		

5.4.2.4. PowerRampSegmentType (Class)

The PowerRampSegmentType data structure is used to define a single array element of the recoveryRamp and stagingRamp array of the Load class. Each array element defines the beginning demand for the line segment and the rate of rise or drop. These attributes combined with the duration completely forms a line segment defining a portion of the ramp.

UML Element Location: *Model\Device and Model Components\Load Component\PowerRampSegmentType*.

Table 5.14 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
beginRamp	The attribute defines the quantity of power at the start of the ramp segment. If this attribute is not defined in the segment, the start of the ramp is assumed to be the end of the ramp of the previous segment.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[0..1]
duration	The attribute defines the time horizon in seconds upon which the associated rise or drop is valid. Constraint: This attribute shall be greater than 0.	Duration See Clause 6.6.1.9.	[1]
rampToCompletion	The attribute defines whether the ramping up or down of this load may be halted in midstream (false) or once started must complete through all segments of the ramp (true). As an example, a multistate fan may only use a portion of the ramp, as it sequences from low to medium to high speed levels (false); whereas, a production line, once started, may need to run through its complete set of ramp segments (true). If the attribute is not defined it is assumed to be false.	Boolean See Clause 6.6.3.3.	[0..1]
rate	The attribute defines rate of rise (positive value) in demand or the rate of drop (negative value) in demand when a load either powers up or shuts down respectively. Its sister attribute, duration, defines the time frame upon which the rate is defined.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]

5.4.2.5. Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

Parent Class: ComponentElement (See Clause 5.2.3.1.)

UML Element Location: *Model\Device and Model Components\Load Component\Load*.

Table 5.15 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
actualDemand	This attribute defines the power being consumed by the device (as measured by the present subinterval demand) at the present time in units specified by the PowerMeasurementsSet. Constraint: The "quantity" that makes up the PowerRealQuantity of the PowerMeasurementsSet shall be greater than or equal to 0.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[0..1]
demandLimits	The operational demand characteristics and their associated end points for the load.	PowerRatings See Clause 6.6.4.6.4.	[0..1]
downRamp	This attribute defines the reduction in power over time when a load being partially or fully de-energized has a complex load reduction profile. For each element of the Load.downRamp array, the downRamp[n].rate defines the amount of power decrease and the downRamp[n].duration defines the length of time in seconds upon which the decrease is in effect. If the downRamp[n].beginRamp attribute is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the downRamp attribute name implies that the rise is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be decreasing. The downRamp function shall measure the time from the load being fully energized until the power is completely depleted. If downRamp is not present, the power decrease to 0 shall be instantaneous. When a curtailable load is partially curtailed (less than the maximumCurtailableDemand) and curtailment is increased, the power will decrease starting at n-th downRamp of the sum from 0 to n of the downRamp[n].rate that is closest to the actualCurtailedDemand.	PowerRampSegmentType See Clause 5.4.2.4.	[0..*]
upRamp	This attribute defines the increase in power over time when a load being partially or fully energized has a complex demand restoration profile. For each element of the Load.upRamp array, the upRamp[n].rate defines the amount of power increase and the upRamp[n].duration defines the length of time in seconds upon which the increase is in effect. If the upRamp[n].beginRamp attribute is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the upRamp attribute name implies that the rise is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing. The upRamp function shall measure the time from the load being fully de-energized until the power is completely restored as defined by Load.maximumDemand. If this attribute	PowerRampSegmentType See Clause 5.4.2.4.	[0..*]

Attribute Name	Description	Attribute Type	Multiplicity
locked	is not present, the power increase upon restoration shall be instantaneous. When a curtailable load is partially curtailed (less than the maximumCurtailableDemand) and curtailment is reduced, the power will increase starting at n-th recoveryRamp of the sum from 0 to n of the upRamp[n].rate that is closest to the actualCurtailedDemand. This attribute defines whether the load is locked and therefore ineligible for curtailment; or unlocked and available for curtailment. Load locking behavior changes depending on the load's curtailmentStatus attribute value at the time the load was locked. If the Load.locked attribute is set to TRUE and the: 1) load is not curtailable, the load is immediately locked. The behavior of this operation is a local matter; 2) curtailmentStatus is curtailmentInactive, the load will immediately be locked out from curtailment eligibility; 3) curtailmentStatus is curtailmentNoncompliant, the load will cycle to its curtailmentInactive state then immediately be locked out of curtailment eligibility; 4) If the CurtailableLoad supports multi-stage curtailment, the load will cycle to its curtailmentInactive state for the present curtailment stage and then be locked out of any further curtailment eligibility. 5) curtailmentStatus is curtailmentNoncompliant, the load will cycle to its curtailmentInactive state then immediately be locked out of curtailment eligibility; 6) If the CurtailableLoad supports multi-stage curtailment, the load will cycle to its curtailmentInactive state for the present curtailment stage and then be locked out of any further curtailment eligibility. Loads that are locked will remain in the locked state indefinitely until the Load.locked attribute is reset to FALSE. The mechanism used to unlock the load is a local matter. This attribute defines the current status of the load. For non-curtailable loads, it provides the present communication status and reliability of the data. For curtailable loads, it also defines if the load is eligible for curtailment or why it is ineligible for curtailment.	Boolean See Clause 6.6.3.3.	[0..1]
status		LoadStatusType See Clause 5.4.2.11.	[0..1]

This element has the following connections to other elements:

Table 5.16 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Composition	input	The connection point associated with the input of this Load.	ConnectionPoint See Clause 6.6.5.2.	[1]

5.4.2.6. FSGIMPayloadFloatType (Class)

The curtailment request is of the form of either an absolute or relative demand curtailment value from the present baseline demand. This class requires the use of a 'setpoint' or 'delta' signal type. This class is based on the EIClasses::PayloadFloatType Class but includes metadata that the Energy Interoperation model captures elsewhere.

Parent Class: FSGIMPayloadBaseType (See Clause 5.4.2.9.)

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadFloatType*.

Table 5.17 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
value	This attribute defines an absolute amount of demand to curtail or release in units defined by the PowerMeasurementsSet. For values greater than zero, the load will attempt to curtail this specified amount. For values less than zero, the load will attempt to release control of the specified amount. The latter functionality facilitates the ancillary services market.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]

5.4.2.7. FSGIMPayloadLevelType (Class)

The curtailment request is of the form of a predefined level of curtailment. This curtailment level's range is specified by the array size of the curtailmentRatingsLevelIndex attribute. The value of the integer requestedCurtailmentLevel.curtailmentValue.patloadBase.level is the index into the curtailmentRatingsLevel array. This class supports the 'level' signal type. This class is identical to the EIClasses::PayloadLevelType Class.

Parent Class: FSGIMPayloadBaseType (See Clause 5.4.2.9.)

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadLevelType*.

Table 5.18 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
level	This attribute defines an index into the CurtailableLoad.curtailmentRatingsLevel array. When directed to be curtailed by 'level', the load will curtail the demand as specified by curtailmentRatingsLevel[[level]]. Constraint: This attribute's value shall be positive. Constraint: This attribute's value shall not exceed the number of array elements defined in the curtailmentRatings attribute.	Integer See Clause 6.6.3.5.	[1]

5.4.2.8. FSGIMPayloadQuantityType (Class)

The curtailment request is in the form of a percentage of the present baseline demand. This class requires the use of a 'multiplier' signal type. This class is identical to the EIClasses::PayloadQuantityType Class.

Constraint: 0 < Percentage <= 100

Parent Class: FSGIMPayloadBaseType (See Clause 5.4.2.9.)

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadQuantityType*.

Table 5.19 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
quantity	This attribute defines a relative amount of demand to curtail the load in the units defined in PowerMeasurementsSet. The relative amount is a percentage of the present demand baseline. Constraint: percentage <= 100 Constraint: percentage >= 0	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]

5.4.2.9. FSGIMPayloadBaseType (Class)

This class is the union of the different classes used to define the needed curtailment of an individual load.

Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Device and Model Components\Load Component\FSGIMPayloadBaseType*.

5.4.2.10. CurtailmentStatusType (Enumeration)

This enumeration indicates if the load is presently curtailed and if curtailed is in compliance with the curtailment request received.

UML Element Location: *Model\Device and Model Components\Load Component\CurtailmentStatusType*.

Table 5.20 – CurtailmentStatusType Enumerated Values

Enumerated Value	Definition
curtailmentInactive	This load is not presently curtailed
curtailmentRequestPending	The curtailment request has been processed and is pending.
curtailmentCompliant	The curtailment request has been processed. The load is compliant with the request.
curtailmentNoncompliant	The curtailment request has been processed. The load is not compliant with the request.

5.4.2.11. LoadStatusType (Enumeration)

This enumeration provides the present overall state of the load.

UML Element Location: *Model\Device and Model Components\Load Component\LoadStatusType*.

Table 5.21 – LoadStatusType Enumerated Values

Enumerated Value	Definition
NA	Not applicable
eligible	The load is presently communicating properly and its data values are correct. In addition, for curtailable loads this load is presently eligible to be curtailed.
loadLocked	The load is ineligible for curtailment since it has been locked.
loadOverridden	An external process has set this override attribute prohibiting curtailment.
lostCommunication	The load presently cannot be accessed.
maxCurtailQueue	The load is in curtailment and presently being timed for the maximum curtailment time.
minCurtailQueue	The load has been cycled the maximum number of times this period.
pointNotConsumingEnergy	The load is in curtailment and presently being timed for the minimum curtailment time.
pointUnreliable	The load is ineligible for curtailment since the point associated with the load is already shut off and not consuming any energy. The present value of a point is questioned due to some hardware or software failure. When a point is unreliable, it still may present a value (e.g., Space Temp Present Value = 67 DegF) but carries along a second attribute that indicates this value is suspect. When a point is in the unreliable state, it shall not be curtailed.
ramping	The load is ramping. That is, it is a transitional state and is either starting up or shutting down. While in this temporary state, it is not eligible for curtailment.
releaseQueue	The load is ineligible for curtailment. The load has completed its curtailment and is presently timing down the restore time before it is again eligible for curtailment.
unlocked	The load has recently been unlocked. It will analyze all conditions and set its present eligibility state after analysis completes.

5.5. Generator Component

The Generator Component is the portion of the FSGIM that provides an abstract representation of all devices that produce electricity or store it. This includes fossil fuel powered generators, photovoltaic arrays, wind generators, and any other generation/storage that is operated and managed by the facility.

UML Package Location: *Model\Device and Model Components\Generator Component*

5.5.1. Generator Classes and Elements Diagram

This is a high level presentation of the Generator component.

The classes which make up the Generator model are derived from the IEC 61850 standard and the Control Interface is fully specified in that standard. The term Logical Node (LN) comes from this standard and represents an aggregation of function that can be used along with others to assemble an application instance.

The Generator Component and its composite classes are envisioned to physically represent the "electrical coupling point (ECP)" where the generator or storage device connects to the electric power system.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

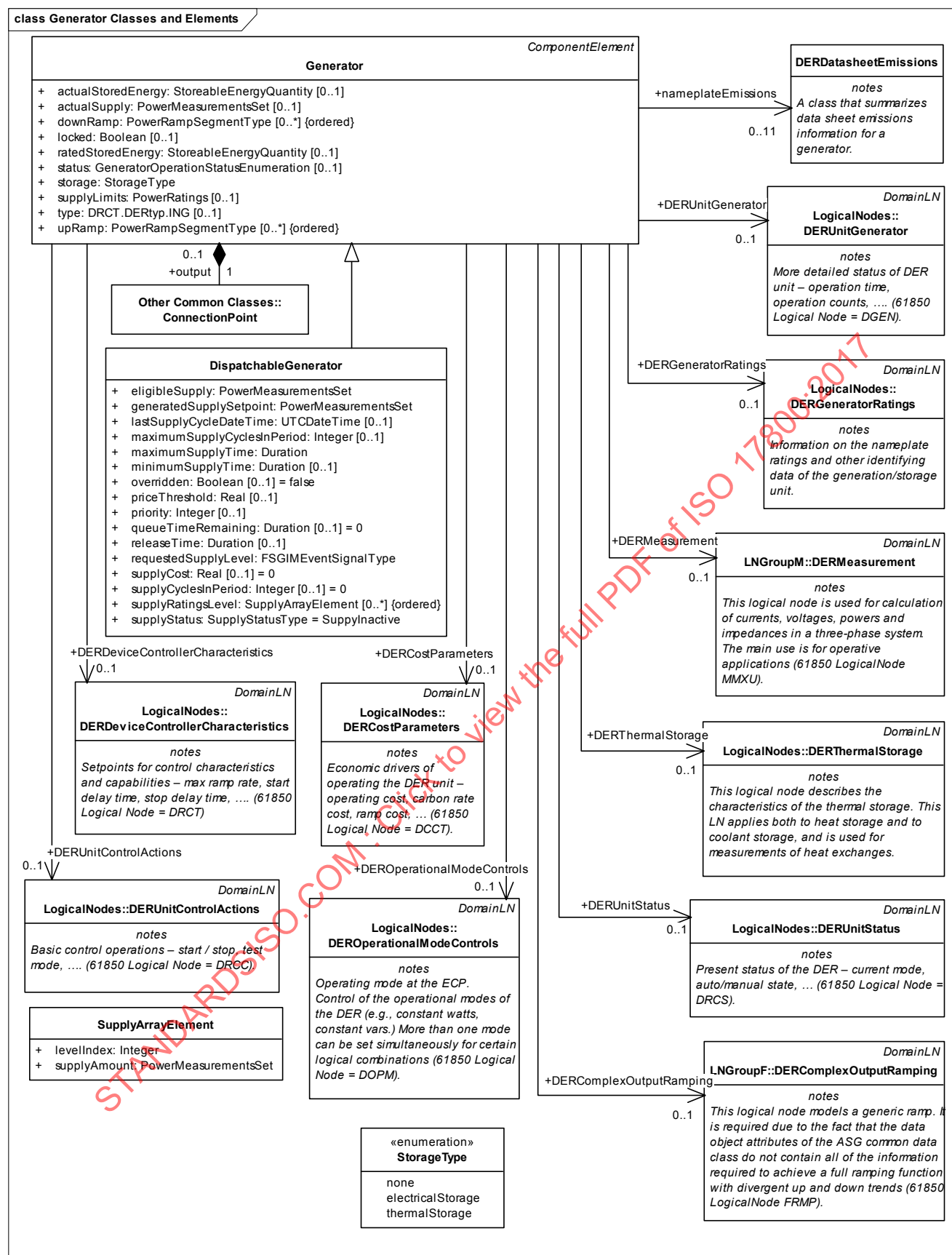


Figure 5.7 - Generator Classes and Elements

The Generator Classes and Elements Diagram makes use of the following elements:

ConnectionPoint (Class)

This class is used to model the physical connections from Generators, Loads, Meters, and Energy Routers to sources or sinks of energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\ConnectionPoint*. See Clause 6.6.5.2 for the complete definition of this element.

DERComplexOutputRamping (Class)

This logical node models a generic ramp. It is required due to the fact that the data object attributes of the ASG common data class do not contain all of the information required to achieve a full ramping function with divergent up and down trends (61850 LogicalNode FRMP).

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupF\DERComplexOutputRamping*. See Clause 5.7.4.5.4.3.2 for the complete definition of this element.

DERCostParameters (Class)

Economic drivers of operating the DER unit – operating cost, carbon rate cost, ramp cost, ... (61850 Logical Node = DCCT).

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERCostParameters*. See Clause 5.7.4.5.1.3.1 for the complete definition of this element.

DERDatasheetEmissions (Class)

A class that summarizes data sheet emissions information for a generator.

UML Element Location: *Model\Device and Model Components\Generator Component\DERDatasheetEmissions*. See Clause 5.5.6.1 for the complete definition of this element.

DERDeviceControllerCharacteristics (Class)

Setpoints for control characteristics and capabilities – max ramp rate, start delay time, stop delay time, ... (61850 Logical Node = DRCT)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERDeviceControllerCharacteristics*. See Clause 5.7.4.5.1.3.2 for the complete definition of this element.

DERGeneratorRatings (Class)

Information on the nameplate ratings and other identifying data of the generation/storage unit.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERGeneratorRatings*. See Clause 5.7.4.5.1.3.3 for the complete definition of this element.

DERMeasurement (Class)

This logical node is used for calculation of currents, voltages, powers and impedances in a three-phase system. The main use is for operative applications (61850 LogicalNode MMXU).

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupM\DERMeasurement*. See Clause 5.7.4.5.4.3.8 for the complete definition of this element.

DEROperationalModeControls (Class)

Operating mode at the ECP.

Control of the operational modes of the DER (e.g., constant watts, constant vars.) More than one mode can be set simultaneously for certain logical combinations (61850 Logical Node = DOPM).

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DEROperationalModeControls*. See Clause 5.7.4.5.1.3.4 for the complete definition of this element.

DERThermalStorage (Class)

This logical node describes the characteristics of the thermal storage. This LN applies both to heat storage and to coolant storage, and is used for measurements of heat exchanges.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERThermalStorage. See Clause 5.7.4.5.1.3.5 for the complete definition of this element.

DERUnitControlActions (Class)

Basic control operations – start / stop, test mode, ... (61850 Logical Node = DRCC).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitControlActions. See Clause 5.7.4.5.1.3.6 for the complete definition of this element.

DERUnitGenerator (Class)

More detailed status of DER unit – operation time, operation counts, ... (61850 Logical Node = DGEN).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitGenerator. See Clause 5.7.4.5.1.3.7 for the complete definition of this element.

DERUnitStatus (Class)

Present status of the DER – current mode, auto/manual state, ... (61850 Logical Node = DRCS).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitStatus. See Clause 5.7.4.5.1.3.8 for the complete definition of this element.

DispatchableGenerator (Class)

The DispatchableGenerator class defines the attributes that are unique to power generating devices that can supply power when needed. The attributes define the information needed to represent supply policies based on criteria including priorities, cost constraints, protection of physical equipment, and safety considerations. The DispatchableGenerator class also provides attributes for orderly reduction of generated supply.

UML Element Location: *Model\Device and Model Components\Generator Component\DispatchableGenerator*. See Clause 5.5.6.3 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

StorageType (Enumeration)

An enumeration that defines the energy storage characteristics of an instance of the Generator Class.

UML Element Location: *Model\Device and Model Components\Generator Component\StorageType*. See Clause 5.5.6.5 for the complete definition of this element.

SupplyArrayElement (Class)

This class defines a single array element of the supplyRatingsLevel attribute. Each array element defines the fixed supply to which the generator shall be controlled as a function of the stated level. The array is ordered by the supplyRatingsLevel.levelIndex attribute.

UML Element Location: *Model\Device and Model Components\Generator Component\SupplyArrayElement*. See Clause 5.5.6.4 for the complete definition of this element.

5.5.2. Generator Model Guide Diagram

This generator model guide provides a key to understanding the composition of the generator component classes. It shows the simple hierarchy of logical nodes, common data classes, and simple primitive attributes used to compose the parts of the generator component.

class Generator Model Guide

Generator Model Guide

The Generator Component is composed of classes called "Logical Nodes (LN)s". Each logical node provides a model of distinct dimension of the generator component. Shown here are the operational mode controls of the generator -- "DEROperationalModeControls"

Logical Nodes are composed of "Common Data Classes (CDC)s". Each CDC in turn has attributes that are typically, but not always, primitives such as an integer or string.

LN's have CDCs as Attributes:
e.g., OperationalModeConstantW

Domain LN	
LogicalNodes::DEROperationalModeControls	
+ RampTimeSeconds: ING [0..1]	
«Controls»	
+ OperationalModeConstantW: SPC [0..1]	
+ OperationModeConstantPowerFactor: SPC [0..1]	
+ OperationModeConstantV: SPG [0..1]	
+ OperationModeConstantVAR: SPC [0..1]	
+ OperationModeExportImport: SPC [0..1]	
+ OperationModeIslanded: SPC [0..1]	
+ OperationModeMaximumVAR: SPC [0..1]	
+ OperationModePeak: SPC [0..1]	
+ OperationModePM: SPC [0..1]	
+ OperationModeVOverride: SPC [0..1]	
+ RevertTimeSeconds: ING [0..1]	
+ WindowTimeSeconds: ING [0..1]	

Most CDCs have primitives as Attributes
e.g., Single Point Control (SPC) ctlVal is Boolean, t is AbsoluteDateTime, etc..

Some are structured, such as q which is a Quality structure.

CDCControl::SPC	
+ ctlNum: Integer [0..1]	
+ ctlVal: Boolean	
+ origin: Originator [0..1]	
+ pulseConfig: PulseConfig [0..1]	
+ q: Quality [0..1]	
+ stSeld: Boolean [0..1]	
+ stVal: Boolean [0..1]	
+ subVal: Boolean [0..1]	
+ t: UTCDateTime [0..1]	

Some primitives are enumerations
e.g.,

Quality.validity:ValidityKind

«Enumeration,... ConstructedDAs:: ValidityKind	
good = 0	
invalid = 1	
questionable = 3	

ConstructedDAs::Quality	
+ detailQual: DetailQual	
+ operatorBlocked: Boolean = false	
+ source: SourceKind = process	
+ test: Boolean = false	
+ validity: ValidityKind	

Example Control Operation shows how you would use this model (assuming an appropriate service was defined and service definitions are themselves outside the scope of FSGIM):

Set

DEROperationalModeControls.OperationModeConstantW.ctlVal
to TRUE

to establish constant Watts mode of operation

Figure 5.8 - Generator Model Guide

The Generator Model Guide Diagram makes use of the following elements:

DEROperationalModeControls (Class)

Operating mode at the ECP.

Control of the operational modes of the DER (e.g., constant watts, constant vars.) More than one mode can be set simultaneously for certain logical combinations (61850 Logical Node = DOPM).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DEROperationalModeControls. See Clause 5.7.4.5.1.3.4 for the complete definition of this element.

Quality (Class)

Quality contains data that describe the quality of the data from the server. Quality of the data is also related to the mode of a logical node. Further details can be found in IEC 61850-7-4. The different quality attributes are *not* independent. The default value shall be applied if the functionality of the related attribute is not supported. The mapping may specify to exclude the attribute from the message if it is not supported or if the default value applies.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\ConstructedDAs\Quality. See Clause 5.7.4.5.3.2.6 for the complete definition of this element.

SPC (Class)

This common data class is used to represent single point controls.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControl\SPC. See Clause 5.7.4.5.3.1.16 for the complete definition of this element.

ValidityKind (Enumeration)

This type is used as validity of a value. See statechart diagram 'Quality.validity'.

UML Element Location: *Model\Model Elements from External*

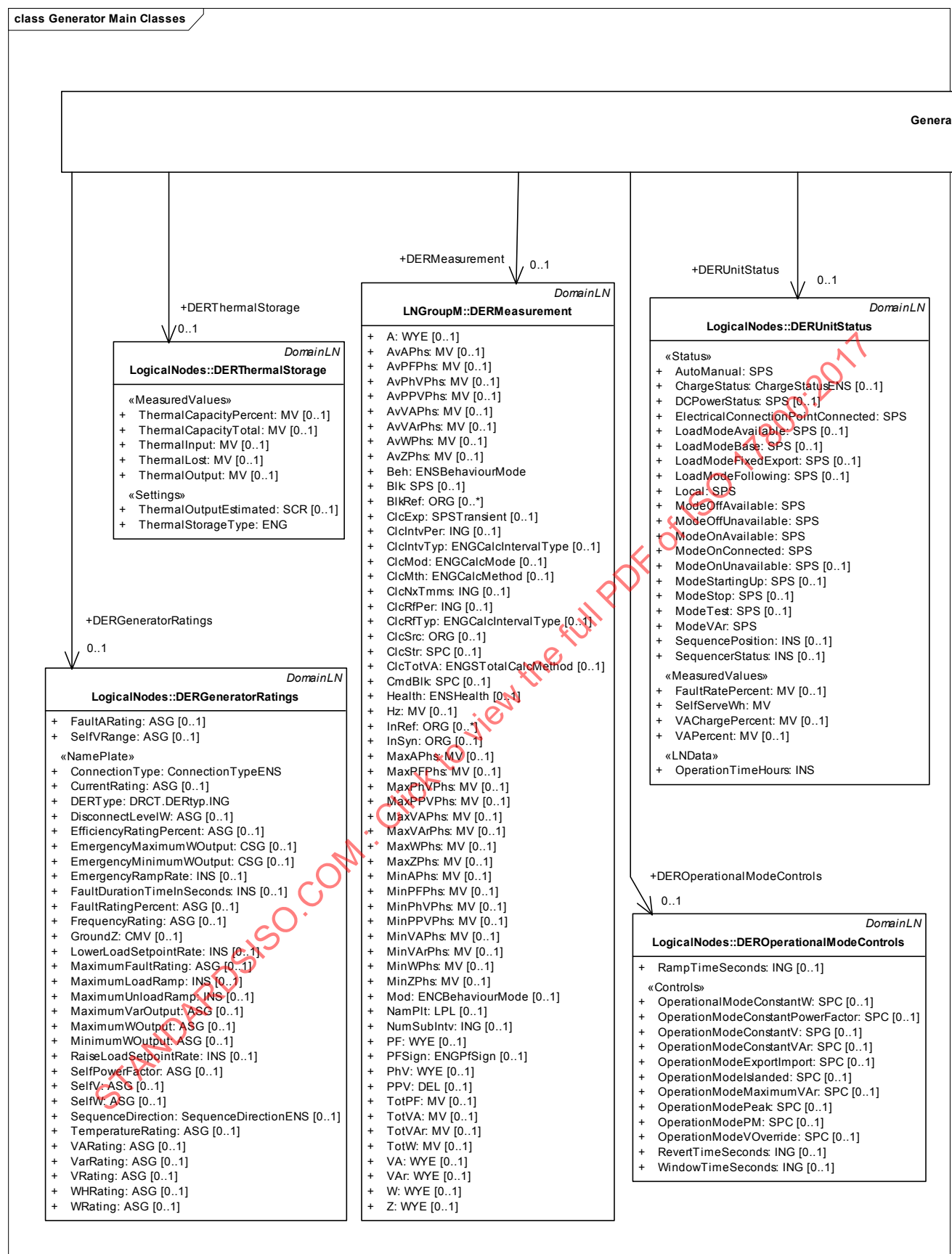
Sources\IEC61850\IEC61850_7_3\ConstructedDAs\ValidityKind. See Clause 5.7.4.5.3.2.12 for the complete definition of this element.

5.5.3. Generator Main Classes Diagram

This diagram presents the top level classes of the Generator Component. Shown are classes that contain details describing the nameplate, cost parameters, control characteristics, status, controllable actions, and operational modes of generators and storage devices. Additionally, the DERStatusHistory class is introduced which is a container for status information that can be aggregated by the Energy Manager Component.

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



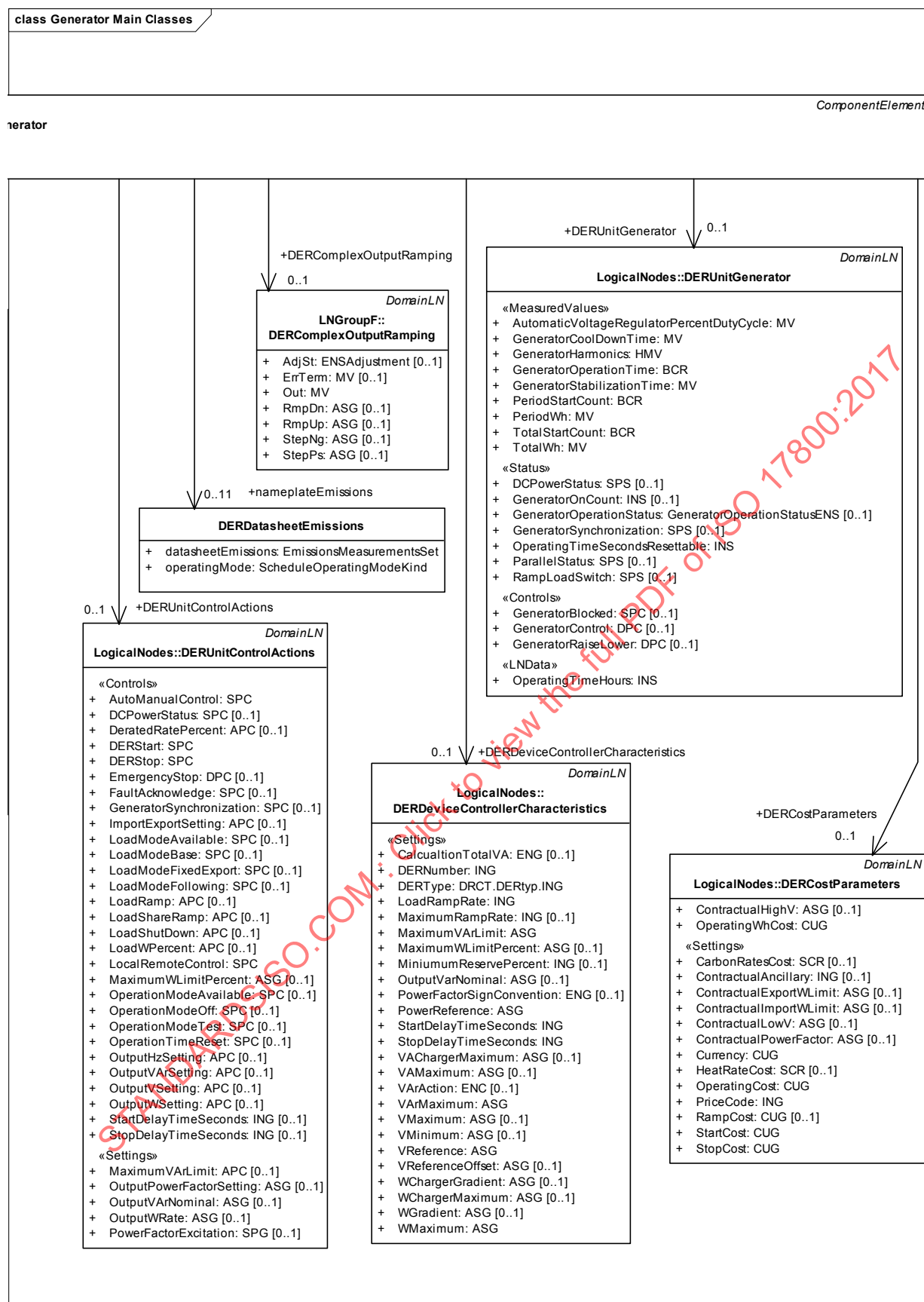


Figure 5.9 - Generator Main Classes

The Generator Main Classes Diagram makes use of the following elements:

DERComplexOutputRamping (Class)

This logical node models a generic ramp. It is required due to the fact that the data object attributes of the ASG common data class do not contain all of the information required to achieve a full ramping function with divergent up and down trends (61850 LogicalNode FRMP).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupF\DERComplexOutputRamping. See Clause 5.7.4.5.4.3.2 for the complete definition of this element.

DERCostParameters (Class)

Economic drivers of operating the DER unit – operating cost, carbon rate cost, ramp cost, ... (61850 Logical Node = DCCT).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERCostParameters. See Clause 5.7.4.5.1.3.1 for the complete definition of this element.

DERDatasheetEmissions (Class)

A class that summarizes data sheet emissions information for a generator.

UML Element Location: *Model\Device and Model Components\Generator Component\DERDatasheetEmissions*. See Clause 5.5.6.1 for the complete definition of this element.

DERDeviceControllerCharacteristics (Class)

Setpoints for control characteristics and capabilities – max ramp rate, start delay time, stop delay time, ... (61850 Logical Node = DRCT)

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERDeviceControllerCharacteristics. See Clause 5.7.4.5.1.3.2 for the complete definition of this element.

DERGeneratorRatings (Class)

Information on the nameplate ratings and other identifying data of the generation/storage unit.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERGeneratorRatings. See Clause 5.7.4.5.1.3.3 for the complete definition of this element.

DERMeasurement (Class)

This logical node is used for calculation of currents, voltages, powers and impedances in a three-phase system. The main use is for operative applications (61850 LogicalNode MMXU).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupM\DERMeasurement. See Clause 5.7.4.5.4.3.8 for the complete definition of this element.

DEROperationalModeControls (Class)

Operating mode at the ECP.

Control of the operational modes of the DER (e.g., constant watts, constant vars.) More than one mode can be set simultaneously for certain logical combinations (61850 Logical Node = DOPM).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DEROperationalModeControls. See Clause 5.7.4.5.1.3.4 for the complete definition of this element.

DERThermalStorage (Class)

This logical node describes the characteristics of the thermal storage. This LN applies both to heat storage and to coolant storage, and is used for measurements of heat exchanges.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERThermalStorage. See Clause 5.7.4.5.1.3.5 for the complete definition of this element.

DERUnitControlActions (Class)

Basic control operations – start / stop, test mode, ... (61850 Logical Node = DRCC).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitControlActions. See Clause 5.7.4.5.1.3.6 for the complete definition of this element.

DERUnitGenerator (Class)

More detailed status of DER unit – operation time, operation counts, ... (61850 Logical Node = DGEN).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitGenerator. See Clause 5.7.4.5.1.3.7 for the complete definition of this element.

DERUnitStatus (Class)

Present status of the DER – current mode, auto/manual state, ... (61850 Logical Node = DRCS).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitStatus. See Clause 5.7.4.5.1.3.8 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

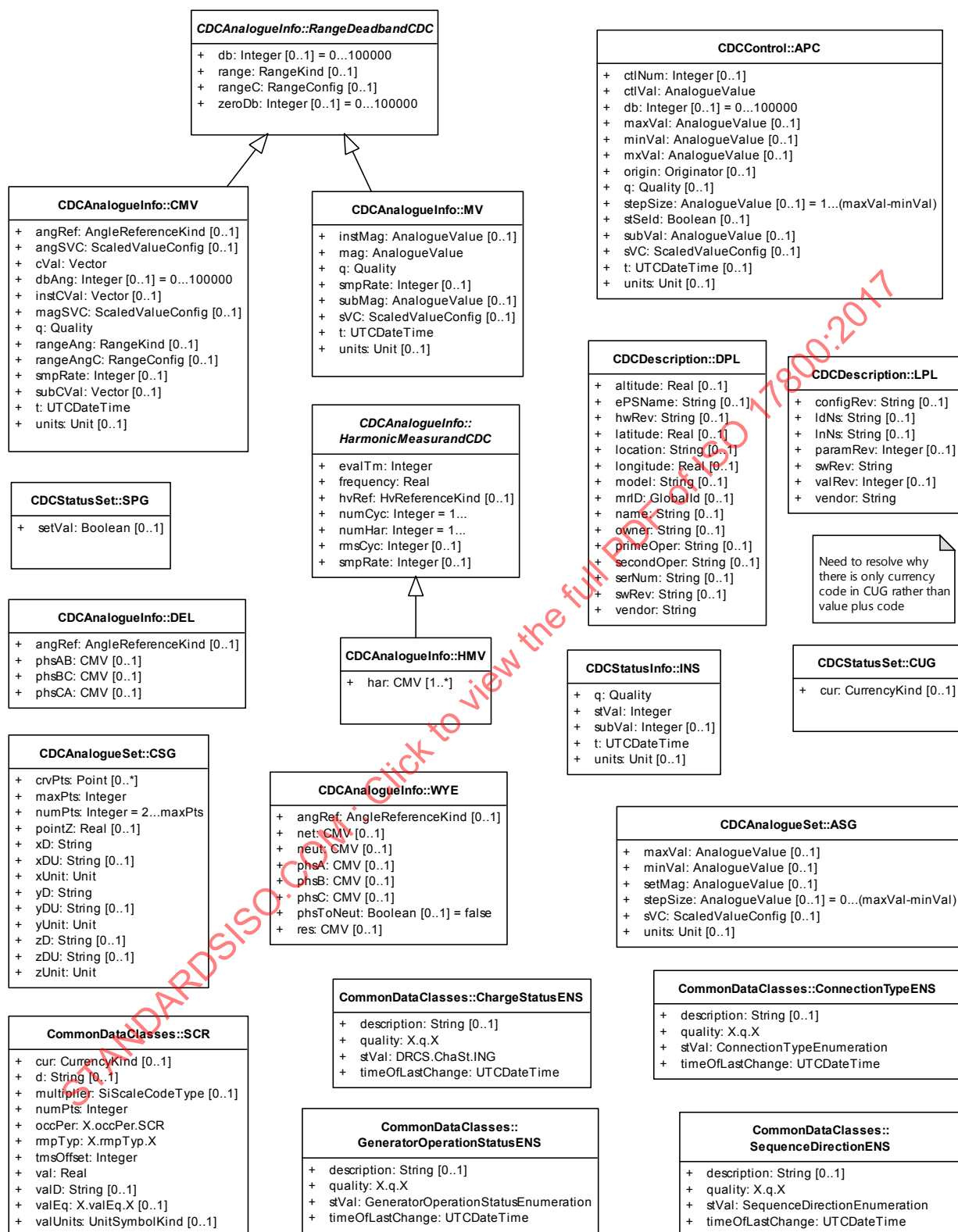
5.5.4. Common Data Classes (CDC)s Used in the Component Diagram

Most of the attributes of the major classes in the Generator Component are themselves based on common data classes that contain the details of each attribute. There are four types of CDCs used in this model:

1. Status: Information representing the status of a process or function.
2. Settings: Information needed for the operation of a process or function.
3. Measured Values: Data measured at the process or function.
4. Controls: Process or function data that can be changed remotely.

For example the major class DEROperationalModeControls has an attribute OperationModeConstantV which is of type SPG. SPG represents a "single point setting"

Common Data Classes used in Generator Component



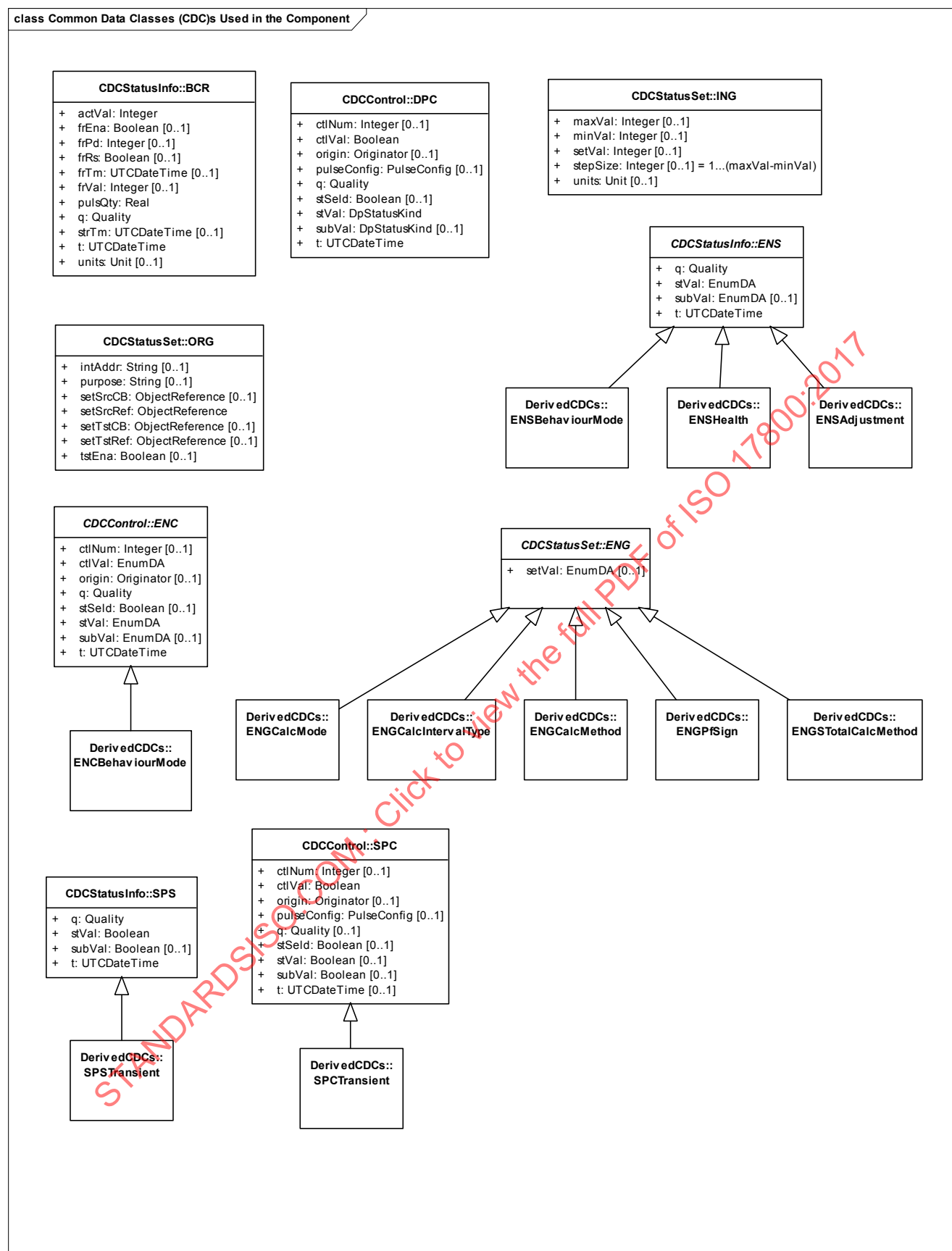


Figure 5.10 - Common Data Classes (CDC)s Used in the Component

The Common Data Classes (CDC)s Used in the Component Diagram makes use of the following elements:

APC (Class)

This common data class is used to represent analogue controls.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControl\APC*. See Clause 5.7.4.5.3.1.13 for the complete definition of this element.

ASG (Class)

This common data class is used to represent analogue settings.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueSet\ASG*. See Clause 5.7.4.5.3.1.10 for the complete definition of this element.

BCR (Class)

This common data class is used to represent binary counter reading.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\BCR*. See Clause 5.7.4.5.3.1.21 for the complete definition of this element.

ChargeStatusENS (Class)

The ChargeStatusENS class is an FSGIM class derived from the 61850 ENS class. The ChargeStatusENS class attributes are exactly the same as those in the ENS class except the status value attribute references the DRCS.ChaSt.ING enumeration class rather than the ENS class.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\CommonDataClasses\ChargeStatusENS*. See Clause 5.7.4.5.1.1.1 for the complete definition of this element.

CMV (Class)

This common data class is used to represent complex (phasor) measured values.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\CMV*. See Clause 5.7.4.5.3.1.2 for the complete definition of this element.

ConnectionTypeENS (Class)

Type of electrical connection between the generator and the facility's electrical distribution system. This class is an FSGIM class derived from the 61850 ENS class. The ConnectTypeENS class attributes are exactly the same as those in the ENS class except the status value attribute references another derived FSGIM enumeration named ConnectionTypeEnum. The ConnectionTypeEnum class is an explicitly defined enumeration in FSGIM. In 61850, this enumeration is represented by a single value.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\CommonDataClasses\ConnectionTypeENS*. See Clause 5.7.4.5.1.1.3 for the complete definition of this element.

CSG (Class)

This common data class is used to hold curve points and describe curve settings.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueSet\CSG*. See Clause 5.7.4.5.3.1.11 for the complete definition of this element.

CUG (Class)

This common data class is used to represent currency settings.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\CUG*. See Clause 5.7.4.5.3.1.26 for the complete definition of this element.

DEL (Class)

This class is a collection of measurements of values in a three phase system that represent phase to phase values ("delta"). Values for 'phsAB', 'phsBC' and 'phsCA' have been simultaneously acquired or determined, and their respective time stamps hold the same value 't.secondSinceEpoch'. It shall be assumed that any jitter between the acquisition times dedicated for these values is neglectable. The jitter for simultaneity shall be as indicated in any of the respective 't.timeQuality' attributes of the above values.

Attribute 'angRef' is used in place of individual 'angRef' attributes of 'phsAB', 'phsBC' and 'phsCA'.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\DEL. See Clause 5.7.4.5.3.1.3 for the complete definition of this element.

DPC (Class)

This common data class is used to represent double point controls.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControl\DPC. See Clause 5.7.4.5.3.1.14 for the complete definition of this element.

DPL (Class)

This common data class is used to identify entities like primary equipment or physical devices.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCDescription\DPL. See Clause 5.7.4.5.3.1.18 for the complete definition of this element.

ENC (Abstract Class)

This common data class is used to represent integer controls with the value restricted to those in an enumeration.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControl\ENC. See Clause 5.7.4.5.3.1.15 for the complete definition of this element.

ENCBehaviourMode (Class)

Possible values for 'stVal', 'ctlVal', 'subVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: Mod.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENCBehaviourMode. See Clause 5.7.4.5.4.1.1 for the complete definition of this element.

ENG (Abstract Class)

This common data class is used to represent integer settings with the value restricted to those in an enumeration.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\ENG. See Clause 5.7.4.5.3.1.27 for the complete definition of this element.

ENGCalcIntervalType (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcIntvTyp, ClcRfTyp.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGCalcIntervalType. See Clause 5.7.4.5.4.1.2 for the complete definition of this element.

ENGCalcMethod (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcMth.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGCalcMethod. See Clause 5.7.4.5.4.1.3 for the complete definition of this element.

ENGCalcMode (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcMod.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGCalcMode. See Clause 5.7.4.5.4.1.4 for the complete definition of this element.

ENGPfSign (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: PFSign.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGPfSign. See Clause 5.7.4.5.4.1.5 for the complete definition of this element.

ENGSTotalCalcMethod (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcTotVA.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGSTotalCalcMethod. See Clause 5.7.4.5.4.1.6 for the complete definition of this element.

ENS (Abstract Class)

This common data class is used to represent integer status values with the value restricted to those in an enumeration.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\ENS. See Clause 5.7.4.5.3.1.22 for the complete definition of this element.

ENSAdjustment (Class)

Possible values for 'stVal', 'subVal' are restricted by enumeration. Used for: AdjSt.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENSAdjustment. See Clause 5.7.4.5.4.1.7 for the complete definition of this element.

ENSBehaviourMode (Class)

Possible values for 'stVal', 'subVal' are restricted by enumeration. Used for: Beh, Mod.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENSBehaviourMode. See Clause 5.7.4.5.4.1.8 for the complete definition of this element.

ENSHealth (Class)

Possible values for 'stVal', 'subVal' are restricted by enumeration. Used for: EEHealth, Health, PhyHealth, MechHealth.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENSHealth. See Clause 5.7.4.5.4.1.9 for the complete definition of this element.

GeneratorOperationStatusENS (Class)

The GeneratorOperationStatusENS class is an FSGIM class derived from the 61850 ENS class. The GeneratorOperationStatusENS class attributes are exactly the same as those in the ENS class except the status value attribute references another derived FSGIM enumeration named GeneratorOperationStatusEnumeration. The GeneratorOperationStatusEnumeration class is an explicitly defined enumeration in FSGIM. In 61850, this enumeration is represented by a single value.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\CommonDataClasses\GeneratorOperationStatusENS. See Clause 5.7.4.5.1.1.2 for the complete definition of this element.

HarmonicMeasurandCDC (Abstract Class)

This abstract supertype groups the configuration attributes for common data classes that provide harmonic and interharmonic measurand information. Attributes are used to configure arrays of harmonic values in concrete CDCs ('har[]' in HMT, {'phsAHar', 'phsBHar', 'phsCHar', 'neutHar', 'netHar', 'resHar'}[] in HWYE, and {'phsABHar', 'phsBCHar', 'phsCAHar'}[] in HDEL).

An array (e.g., 'har[]') contains the harmonic and subharmonic ('evalTm' equals 'frequency' period), or the interharmonic values ('evalTm' differs from 'frequency' period). The first array element 'har[0]' contains the dc component, the further array elements 'har[1]'...'har[numHar]' contain the values for the harmonics 1..'numHar', respectively.

For the case of harmonic and subharmonic values ('evalTm' equals 'frequency' period), if 'numCyc' > 1, 'har[]' contains both harmonics, and subharmonics with their multiples. Array elements with the index that is a multiple of $2^{\text{numCyc}-1}$ are harmonics; all the others are subharmonics or multiple of subharmonics.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\HarmonicMeasurandCDC. See Clause 5.7.4.5.3.1.4 for the complete definition of this element.

HMT (Class)

This class is a collection of values that represent the harmonic or interharmonic content of a process value.

NOTE Harmonics for a single circuit may have phase angles, but need no reference for the angle ('angRef'), since by convention the reference is always the fundamental frequency (index 1).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\HMT. See Clause 5.7.4.5.3.1.5 for the complete definition of this element.

ING (Class)

This common data class is used to represent integer settings.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\ING. See Clause 5.7.4.5.3.1.28 for the complete definition of this element.

INS (Class)

This common data class is used to represent integer status values.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\INS. See Clause 5.7.4.5.3.1.23 for the complete definition of this element.

LPL (Class)

This common data class is used for nameplate information of logical nodes.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCDescription\LPL. See Clause 5.7.4.5.3.1.19 for the complete definition of this element.

MV (Class)

This common data class is used to represent scalar measured values.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\MV. See Clause 5.7.4.5.3.1.6 for the complete definition of this element.

ORG (Class)

This common data class is used to represent object reference settings, such as object reference to the logical node of which the statistical data have been calculated.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\ORG. See Clause 5.7.4.5.3.1.29 for the complete definition of this element.

RangeDeadbandCDC (Abstract Class)

This abstract supertype groups the attributes common to those common data classes that provide deadbanded values (in addition to instantaneous ones).

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\RangeDeadbandCDC. See Clause 5.7.4.5.3.1.7 for the complete definition of this element.

SCR (Class)

The SCR class defines the common data class definition for a "Relative Time Schedule Setting". This class provides a means for defining a relative time array of setting values, such as schedules, where the time interval between points may be variable. It has an array of value elements that is characterized by a set of parameters including the number of elements in each array, an array of times between each value and an indication how to ramp between consecutive values, units, multiplier, and a description.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\CommonDataClasses\SCR. See Clause 5.7.4.5.1.1.5 for the complete definition of this element.

SequenceDirectionENS (Class)

The SequenceDirectionENS class is an FSGIM class derived from the 61850 ENS class. The SequenceDirectionENS class attributes are exactly the same as those in the ENS class except the status value attribute references another derived FSGIM enumeration named SequenceDirectionEnumeration. The SequenceDirectionEnumeration class is an explicitly defined enumeration in FSGIM. In 61850, this enumeration is represented by a single value.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\CommonDataClasses\SequenceDirectionENS. See Clause 5.7.4.5.1.1.4 for the complete definition of this element.

SPC (Class)

This common data class is used to represent single point controls.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControl\SPC. See Clause 5.7.4.5.3.1.16 for the complete definition of this element.

SPCTransient (Class)

An SPC whose 'stVal' is transient.

Not explicitly defined in the standard.

Transient Data - the status of data with this designation is momentary and must be logged or reported to provide evidence of their momentary state. The transient property of data only applies to boolean process data attributes (FC=ST) of that data. Transient data is identical to normal data, except that for the process state change from TRUE to FALSE no event is generated for reporting and for logging.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\SPCTransient. See Clause 5.7.4.5.4.1.10 for the complete definition of this element.

SPG (Class)

This common data class is used to represent single point settings.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\SPG. See Clause 5.7.4.5.3.1.30 for the complete definition of this element.

SPS (Class)

This common data class is used to represent single point status values.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\SPS. See Clause 5.7.4.5.3.1.24 for the complete definition of this element.

SPSTransient (Class)

An SPS whose 'stVal' is transient.
Not explicitly defined in the standard.

Transient Data - the status of data with this designation is momentary and must be logged or reported to provide evidence of their momentary state. The transient property of data only applies to boolean process data attributes (FC=ST) of that data. Transient data is identical to normal data, except that for the process state change from TRUE to FALSE no event may be generated for reporting and for logging.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\DerivedCDCs\SPSTransient. See Clause 5.7.4.5.4.1.11 for the complete definition of this element.

WYE (Class)

This class is a collection of simultaneous measurements of values in a three phase system that represent phase to ground values ("wye").

Values for 'phsA', 'phsB', 'phsC', 'neut', 'net' and 'res' have been simultaneously acquired or determined, and their respective time stamps hold the same value 't.secondSinceEpoch'. It shall be assumed that any jitter between the acquisition times dedicated for these values is neglectable. The jitter for simultaneity shall be as indicated in any of the respective 't.timeQuality' attributes of the above values.

Attribute 'angRef' is used in place of individual 'angRef' attributes of 'phsA', 'phsB', 'phsC', 'neut', 'net' and 'res'.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\WYE. See Clause 5.7.4.5.3.1.8 for the complete definition of this element.

5.5.5. Generator Enumerations Diagram

This diagram contains enumerations used by the Generator Component.

class Generator Enumerations

«Enumeration» LogicalNodeEnums:: AdjustmentKind Completed = 1 Cancelled = 2 New adjustments = 3 Under way = 4	«Enumeration» DAEnums:: AngleReferenceKind Va = 0 Vb = 1 Vc = 2 Aa = 3 Ab = 4 Ac = 5 Vab = 6 Vbc = 7 Vca = 8 Vother = 9 Aother = 10 Synchrophasor = 11	«Enumeration» LogicalNodeEnums:: BehaviourModeKind on = 1 on-blocked = 2 test = 3 test/blocked = 4 off = 5	«Enumeration» LogicalNodeEnums:: CalcIntervalTypeKind MS = 1 PER_CYCLE = 2 CYCLE = 3 DAY = 4 WEEK = 5 MONTH = 6 YEAR = 7 EXTERNAL = 8	«Enumeration» LogicalNodeEnums:: CalcMethodKind UNSPECIFIED = 1 TRUE_RMS = 2 PEAK_FUNDAMENTAL = 3 RMS_FUNDAMENTAL = 4 MIN = 5 MAX = 6 AVG = 7 SDV = 8 PREDICTION = 9 RATE = 10
«Enumeration» LogicalNodeEnums:: CalcModeKind TOTAL = 1 PERIOD = 2 SLIDING = 3	«Enumeration» DAEnums:: CtlModelKind status-only = 0 direct-with-normal-security = 1 sbo-with-normal-security = 2 direct-with-enhanced-security = 3 sbo-with-enhanced-security = 4	«enumeration» DAEnums:: ConnectionTypeEnumeration NA = 0 SinglePhase = 1 SplitPhase = 2 TwoPhase = 3 ThreePhaseDelta = 4 ThreePhaseWye = 5 ThreePhaseWyeGrounded = 6 ThreePhase_ThreeWire_InverterType = 7 ThreePhase_FourWire_InverterType = 8 Other = 99	«Enumeration» DAEnums:: ControlOutputKind pulse = 0 persistent = 1	
«enumeration» Enumerations:: CurrencyKind <i>notes</i> <i>Currency is represented in FSGIM as an enumeration. The use of this is optional so it can be omitted when an actual currency is not known. The enumeration is based on ISO 4217 three character codes and the FSIM model contains a snapshot of this list. Any implementation can extend this list as needed but must unambiguously identify how extensions are separate from the ISO standard set.</i>	«enumeration» DAEnums:: DRCT.DERTyp.ING Unknown = 0 virtual = 1 reciprocatingEngine = 2 fuelCell = 3 photoVoltaic = 4 combinedHeatAndPower = 5 compressedAir = 6 pumpedHydroelectric = 7 flywheel = 8 thermalEnergyStorage = 9 electroChemicalBattery = 10 flowBattery = 11 capacitor = 12 superconductingMagneticEnergy = 13 other = 99	«enumeration» DAEnums:: GeneratorOperationStatusEnumeration NA = 0 NotOperating = 1 Operating = 2 StartingUp = 3 ShuttingDown = 4 AtDisconnectLevel = 5 Ramping_Power = 6 Ramping_ReactivePower = 7 Other = 99	«Enumeration» LogicalNodeEnums:: HealthKind Ok = 1 Warning = 2 Alarm = 3	
«enumeration» DAEnums:: DRCS.ChaSt.ING off = 0 fullyDischarged = 1 discharging = 2 charging = 3 fullyCharged = 4 holding = 5 testing = 6	«enumeration» DAEnums:: X.ctlModel.X status-only direct-with-normal-security sbo-with-normal-security direct-with-enhanced-security sbo-with-enhanced-security	«enumeration» DAEnums:: ScheduleOperatingModeKind NA = 0 energy = 1 continuousReserve = 2 continuousReserveSupplemental = 3 energyReserve = 4 emissionReserve = 5 energyBalancing = 6 reactivePower = 7 blackStart = 8 islanding = 9 other = 99	«Enumeration» DAEnums:: HvReferenceKind fundamental = 0 rms = 1 absolute = 2	
	«enumerati... DAEnums:: X.occPer.SCR Hour Day Week Month Year	«enumerati... DAEnums:: X.q.X Good Invalid Questionable	«enumerati... DAEnums:: X.rmpTyp.X Fixed = 1 Ramp = 2 Average = 3	«enumerati... DAEnums:: X.valEq.X 1 2 3

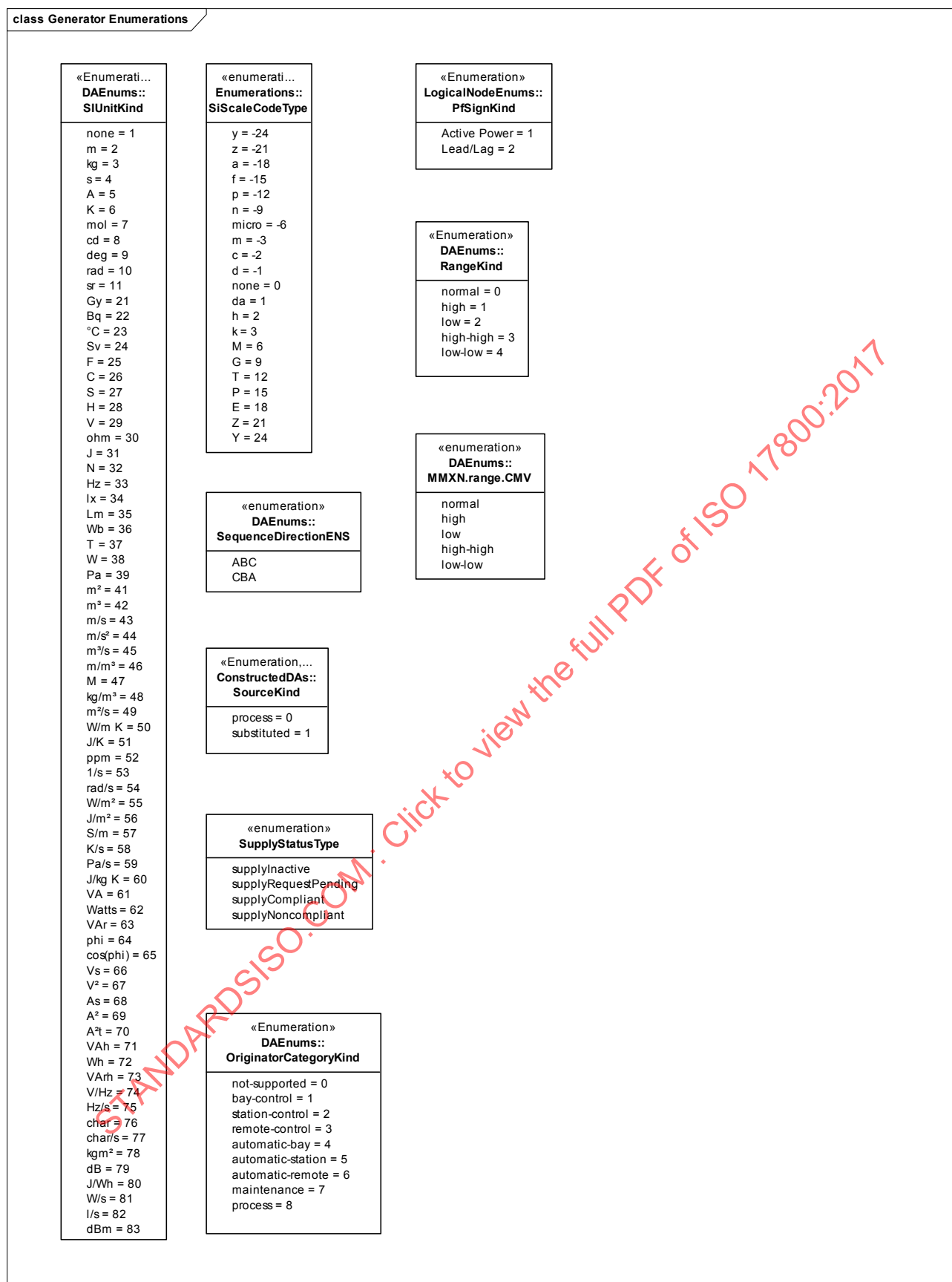


Figure 5.11 - Generator Enumerations

The Generator Enumerations Diagram makes use of the following elements:

AdjustmentKind (Enumeration)

An enumeration of an adjustment status.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\AdjustmentKind. See Clause 5.7.4.5.4.2.1 for the complete definition of this element.

AngleReferenceKind (Enumeration)

An enumeration defining the quantity that is used as reference for the phase angles.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\DAEnums\AngleReferenceKind. See Clause 5.7.4.5.3.3.1 for the complete definition of this element.

BehaviourModeKind (Enumeration)

Switching between the modes (Mod.stVal) should only happen as a result of an operator command to the data object Mod. Mod and Beh are always accessible by the services. The communication services for the data object Mod do not care about the status of the Beh of the LN.

Logical nodes should process receiving data according to their quality information:

- processed as valid means that the application should react according to the quality and the behaviour of the LN.
- processes as invalid means the application should react as if the quality of the data had been invalid.
- processed as blocked means that the application should decide how to react, besides no process-related action based on the value is performed.
- statements "processed" and "not processed" don't belong to communication services and therefore no quality bit can be evaluated.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\BehaviourModeKind. See Clause 5.7.4.5.4.2.2 for the complete definition of this element.

CalcIntervalTypeKind (Enumeration)

An enumeration of a calculation interval frequencies.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\CalcIntervalTypeKind. See Clause 5.7.4.5.4.2.3 for the complete definition of this element.

CalcMethodKind (Enumeration)

An enumeration of calculation methods.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\CalcMethodKind. See Clause 5.7.4.5.4.2.4 for the complete definition of this element.

CalcModeKind (Enumeration)

An enumeration of calculation modes.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\CalcModeKind. See Clause 5.7.4.5.4.2.5 for the complete definition of this element.

ConnectionTypeEnumeration (Enumeration)

Enumerated set of the type of electrical connection between the generator and the electrical distribution system.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\DAEnums\ConnectionTypeEnumeration. See Clause 5.7.4.5.1.2.3 for the complete definition of this element.

ControlOutputKind (Enumeration)

An enumeration defining the kind of control output provided.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\DAEnums\ControlOutputKind. See Clause 5.7.4.5.3.3.2 for the complete definition of this element.

CtlModelKind (Enumeration)

An enumeration defining the kind of control mechanism provided.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\DAEnums\CtlModelKind. See Clause 5.7.4.5.3.3.3 for the complete definition of this element.

CurrencyKind (Enumeration)

Currency is represented in FSGIM as an enumeration. The use of this is optional so it can be omitted when an actual currency is not known. The enumeration is based on ISO 4217 three character codes and the FSIM model contains a snapshot of this list. Any implementation can extend this list as needed but must unambiguously identify how extensions are separate from the ISO standard set.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\CurrencyKind*. See Clause 6.6.2.1 for the complete definition of this element.

DRCS.ChaSt.ING (Enumeration)

This enumeration indicates the state of charge of a storage unit.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\DAEnums\DRCS.ChaSt.ING. See Clause 5.7.4.5.1.2.6 for the complete definition of this element.

DRCT.DERtyp.ING (Enumeration)

Distributed energy resource type.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\DAEnums\DRCT.DERtyp.ING. See Clause 5.7.4.5.1.2.7 for the complete definition of this element.

GeneratorOperationStatusEnumeration (Enumeration)

The GeneratorOperationStatusEnumeration class is an FSGIM enumeration defining the operation state of the generator.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\DAEnums\GeneratorOperationStatusEnumeration. See Clause 5.7.4.5.1.2.4 for the complete definition of this element.

HealthKind (Enumeration)

An enumeration of health status.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\HealthKind. See Clause 5.7.4.5.4.2.6 for the complete definition of this element.

HvReferenceKind (Enumeration)

Reference for harmonic values.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\DAEnums\HvReferenceKind. See Clause 5.7.4.5.3.3.4 for the complete definition of this element.

MMXN.range.CMV (Enumeration)

Measurement minimum and maximum range thresholds.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\DAEnums\MMXN.range.CMV. See Clause 5.7.4.5.1.2.8 for the complete definition of this element.

OriginatorCategoryKind (Enumeration)

An enumeration specifying the category of the originator that caused a change of a value.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\DAEnums\OriginatorCategoryKind. See Clause 5.7.4.5.3.3.5 for the complete definition of this element.

PfSignKind (Enumeration)

An enumeration of power factor sign kind.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\PfSignKind. See Clause 5.7.4.5.4.2.7 for the complete definition of this element.

RangeKind (Enumeration)

An enumeration that indicates the relationship of a measured value against configured settings.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\DAEnums\RangeKind. See Clause 5.7.4.5.3.3.6 for the complete definition of this element.

ScheduleOperatingModeKind (Enumeration)

Ed. Note: The ScheduleOperatingModeKind is a newly defined enumeration in 61850. It has not as yet (Sep 2013) been approved by the committee. Hence, this enumeration was cloned from the 61850 working documents and is subject to modification.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\DAEnums\ScheduleOperatingModeKind. See Clause 5.7.4.5.1.2.1 for the complete definition of this element.

SequenceDirectionENS (Enumeration)

Enumeration that defines the phase sequence connection of the generator.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_420\DAEnums\SequenceDirectionENS. See Clause 5.7.4.5.1.2.2 for the complete definition of this element.

SiScaleCodeType (Enumeration)

Unit multiplier, where the initial value of literal equals the exponent of the multiplier value in base 10. Only the value "none" is normative to the FSGIM.

Note: A value that is representing a percentage can use the UnitSymbolKind of 1 (none) and an SiScaleCodeType of -2 (centi).

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\SiScaleCodeType*. See Clause 6.6.2.3 for the complete definition of this element.

SIUnitKind (Enumeration)

An enumeration of SI units, derived from ISO 1000.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\DAEnums\SIUnitKind. See Clause 5.7.4.5.3.3.7 for the complete definition of this element.

SourceKind (Enumeration)

Source of a value.

UML Element Location: *Model\Model Elements from External*

Sources\IEC61850\IEC61850_7_3\ConstructedDAs\SourceKind. See Clause 5.7.4.5.3.2.11 for the complete definition of this element.

SupplyStatusType (Enumeration)

This enumeration indicates if the load is presently curtailed and if curtailed is in compliance with the curtailment request received.

UML Element Location: *Model\Device and Model Components\Generator Component\SupplyStatusType*. See Clause 5.5.6.6 for the complete definition of this element.

ValidityKind (Enumeration)

This type is used as validity of a value. See statechart diagram 'Quality.validity'.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\ValidityKind*. See Clause 5.7.4.5.3.2.12 for the complete definition of this element.

X.ctlModel.X (Enumeration)

ENUMERATED (status-only | direct-with-normal-security | sbo-with-normal-security | directwith-enhanced-security | sbo-with-enhanced-security).

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.ctlModel.X*. See Clause 5.7.4.5.1.2.9 for the complete definition of this element.

X.occPer.SCR (Enumeration)

Repeat period.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.occPer.SCR*. See Clause 5.7.4.5.1.2.10 for the complete definition of this element.

X.q.X (Enumeration)

Validity component of the complex "quality" attribute with only the following values: good, Invalid, Questionable.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.q.X*. See Clause 5.7.4.5.1.2.11 for the complete definition of this element.

X.rmpTyp.X (Enumeration)

1=Fixed, 2=Ramp, 3=Average.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.rmpTyp.X*. See Clause 5.7.4.5.1.2.12 for the complete definition of this element.

X.valEq.X (Enumeration)

1 = SI units.
2 = Currency as per ISO 4217 per SI unit.
3 = SI unit per currency.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.valEq.X*. See Clause 5.7.4.5.1.2.13 for the complete definition of this element.

5.5.6. Elements Defined in the Generator Component Model

The following elements are defined in this model.

5.5.6.1. DERDatasheetEmissions (Class)

A class that summarizes data sheet emissions information for a generator.

UML Element Location: *Model\Device and Model Components\Generator Component\DERDatasheetEmissions*.

Table 5.22 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
datasheetEmissions	An EmissionsMeasurementSet containing rated emissions. Constraint: EmissionsMeasurementSet.perWhat = kWh	EmissionsMeasurementSet See Clause 6.6.4.8.12.	[1]
operatingMode	The operating mode of the containing generator Component.	ScheduleOperatingModeKind See Clause 5.7.4.5.1.2.1.	[1]

5.5.6.2. Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

Parent Class: ComponentElement (See Clause 5.2.3.1.)

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*.

Table 5.23 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
actualStoredEnergy	This attribute defines the amount of stored energy available at the present time. The value can be derived from the DERUnitGenerator.TotalWh and DERUnitGenerator.PeriodWh attributes.	StoreableEnergyQuantity See Clause 6.6.4.7.17.	[0..1]
actualSupply	This attribute defines the power being generated by the device at the present time in units specified by PowerMeasurementsSet. The value can be derived from the DERMeasurement.TotW, DERMeasurement.TotVA and, DERMeasurement.TotVA attributes.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[0..1]
downRamp	This attribute defines the Real power component reduction in power over time when a generator being partially or fully de-energized has a complex power ramp. It can be generated from the DERComplexOutputRamping.RmpDn attribute for complex generators. Simple generators can use	PowerRampSegmentType See Clause 5.4.2.4.	[0..*]

Attribute Name	Description	Attribute Type	Multiplicity
	DERDeviceControllerCharacteristics.LoadRampRate to generate the single ramp segment needed. For each element of the Generator.downRamp array, the generatorDownRamp[n].rate defines the amount of power increase and the generatorDownRamp[n].duration defines the length of time in seconds upon which the increase is in effect. If the generatorDownRamp[n].beginRamp attribute is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the downRamp attribute name implies that the rise is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be decreasing. The downRamp function shall measure the time from the generator being fully energized until the power is completely depleted. If downRamp is not present, the power decrease to 0 shall be instantaneous.		
locked	This attribute defines whether the generator is unavailable for use. It can be generated from the DERUnitStatus.ModeOffUnavailable attribute.	Boolean See Clause 6.6.3.3.	[0..1]
ratedStoredEnergy	This attribute defines the maximum amount of stored energy available. The value can be derived from the DERGeneratorRatings.WHRating attribute.	StoreableEnergyQuantity See Clause 6.6.4.7.17.	[0..1]
status	This attribute defines the current operational status of the generator. It can be generated from the DERUnitGenerator.GeneratorOperationStatus attribute.	GeneratorOperationStatusEnumeration See Clause 5.7.4.5.1.2.4.	[0..1]
storage	This attribute defines whether this instance of the GeneratorClass models electrical energy storage, thermal energy storage, or neither. The value of this attribute can be generated from DERDeviceControllerCharacteristics.DERType. If the value of DERDeviceControllerCharacteristics.DERType is "compressedAir", "pumpedHydroelectric", "flywheel", "electroChemicalBattery", "flowBattery", "capacitor", or "superconductingMagneticEnergy", then the value of this enumeration shall be electricalStorage. If the value of DERDeviceControllerCharacteristics.DERType is "thermalEnergyStorage", then the value of this enumeration shall be "thermalStorage."	StorageType See Clause 5.5.6.5.	[1]
supplyLimits	The operational demand characteristics and their associated end points for the generator.	PowerRatings See Clause 6.6.4.6.4.	[0..1]
type	This attribute defines the type of generator or storage system	DRCT.DERTyp.ING See Clause 5.7.4.5.1.2.7.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
upRamp	This attribute defines the Real power component increase in power over time when a generator being partially or fully energized has a complex power ramp. It can be determined from the DERComplexOutputRamping.RmpUp attribute for complex generators. Simple generators can use DERDeviceControllerCharacteristics.LoadRampRate to derive the single ramp segment needed. For each element of the Generator.upRamp array, the upRamp[n].rate defines the amount of power increase and the upRamp[n].duration defines the length of time in seconds upon which the increase is in effect. If the upRamp[n].beginRamp attribute is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the upRamp attribute name implies that the rise is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing. The upRamp shall measure the time from the generator being stopped until the power is completely restored as defined by DERGeneratorRatings.MaximumWOutput. If this attribute is not present, the power increase upon restoration shall be instantaneous.	PowerRampSegmentType See Clause 5.4.2.4.	[0..*]

This element has the following connections to other elements:

Table 5.24 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	DERComplexOutputRamping	Generator ramping behavior	DERComplexOutputRamping See Clause 5.7.4.5.4.3.2.	[0..1]
Association	DERCostParameters	Key economic statistics on generator operation.	DERCostParameters See Clause 5.7.4.5.1.3.1.	[0..1]
Association	DERDeviceControllerCharacteristics	Device control characteristics and control setpoints.	DERDeviceControllerCharacteristics See Clause 5.7.4.5.1.3.2.	[0..1]
Association	DERGeneratorRatings	Nameplate ratings of the generator/storage unit.	DERGeneratorRatings See Clause 5.7.4.5.1.3.3.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	DERMeasurement	Basic generator measurements.	DERMeasurement See Clause 5.7.4.5.4.3.8.	[0..1]
Association	DEROperationalModeControls	Control of the operational modes of the generator/storage.	DEROperationalModeControls See Clause 5.7.4.5.1.3.4.	[0..1]
Association	DERThermalStorage	Characteristics of the thermal storage.	DERThermalStorage See Clause 5.7.4.5.1.3.5.	[0..1]
Association	DERUnitControlActions	Basic generator control operations.	DERUnitControlActions See Clause 5.7.4.5.1.3.6.	[0..1]
Association	DERUnitGenerator	Detailed status of generator/storage.	DERUnitGenerator See Clause 5.7.4.5.1.3.7.	[0..1]
Association	DERUnitStatus	Operational status of the generator/storage.	DERUnitStatus See Clause 5.7.4.5.1.3.8.	[0..1]
Association	nameplateEmissions	Generator emissions per operating mode.	DERDatasheetEmissions See Clause 5.5.6.1.	[0..11]
Composition	output	The connection point associated with the output of this Generator.	ConnectionPoint See Clause 6.6.5.2.	[1]

5.5.6.3. DispatchableGenerator (Class)

The DispatchableGenerator class defines the attributes that are unique to power generating devices that can supply power when needed. The attributes define the information needed to represent supply policies based on criteria including priorities, cost constraints, protection of physical equipment, and safety considerations. The DispatchableGenerator class also provides attributes for orderly reduction of generated supply.

Parent Class: Generator (See Clause 5.5.6.2.)

UML Element Location: *Model\Device and Model Components\Generator Component\DispatchableGenerator.*

Table 5.25 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
eligibleSupply	This attribute defines the amount of supply still eligible to be added by the generator. For two-state generators this value will either be the value of maximumSupply as defined in the Generator class or zero. For multi-state generators this value is the total remaining supply that can be added as defined in the supplyRatingsLevel attribute. Constraint: The value of this attribute shall be greater or equal to 0. Constraint: The value of this attribute shall not be greater than the value of the Load.maximumLoad attribute.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
generatedSupplySetpoint	This attribute defines the maximum power threshold of the generator at the present time. Constraint: This attribute's value shall be greater than or equal to 0.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
lastSupplyCycleDateTime	This attribute defines the date and time on which the generator last supplied power. For multistage generators, each time a stage is switched, this attribute shall be updated.	UTCDateTime See Clause 6.6.1.11.	[0..1]
maximumSupplyCyclesInPeriod	This attribute defines the maximum number of times this generator may supply energy within the specified period. When the supplyCyclesInPeriod reaches this attribute's value, the generatorStatus attribute will be set to maxCyclesThisPeriod rendering the generator ineligible to supply energy until the period information is reset. The duration of the period and the resetting of this attribute is a local matter. Constraint: If the actualCurtailmentsInPeriod attribute is defined, then this attribute shall be defined. Constraint: This attribute's value shall be greater than or equal to 0.	Integer See Clause 6.6.3.5.	[0..1]
maximumSupplyTime	This attribute specifies the maximum number of seconds that a generator may supply power before it shall be released. This duration is often used to manage emission levels, or meet safety code requirements. It is used to set the queueTimeRemaining attribute when placing the generator in the maxSupplyQueue state. Constraint: This attribute's value shall be greater than 0. Constraint: This attribute's value shall be greater than or equal to the minimumCurtailTime attribute's value.	Duration See Clause 6.6.1.9.	[1]
minimumSupplyTime	This attribute specifies the minimum number of seconds that a generator shall be powered up before it may be released. This duration may be used to protect the generator's hardware from excess cycling or to meet safety code requirements. It is used to set the QueueTimeRemaining attribute when placing the load in the minCurtailQueue state. Constraint: This attribute's value shall be greater than 0. Constraint: This attribute's value shall be less than or equal to the maximumCurtailTime attribute's value.	Duration See Clause 6.6.1.9.	[0..1]
overridden	This attribute is typically controlled by an external entity to affect eligibility of the generator. For example, the Energy Manager may override this generator if the present price of fuel is outside a defined economic threshold. This attribute differs from the generatorLocked attribute in that this attribute is meant to	Boolean See Clause 6.6.3.3.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
	be connected to an automation system and only persists while the override is in effect. The generatorLocked attribute persists indefinitely and meant to be set by manual intervention.		
priceThreshold	This attribute defines the marginal operational cost per unit of power basis for this generator. The priority is not created within the generator; this is a placeholder for the Energy Manager. This attribute can be derived from the DERCostParameters.OperatingWCost attribute. Constraint: This attribute's value shall be greater than or equal to 0.	Real See Clause 6.6.3.6.	[0..1]
priority	This attribute defines the order upon which a generator starts across a set of generators. If the attribute is not present, it shall be assumed to have the lowest priority. The priority is not created within the generator, this is a placeholder for the Energy Manager. Constraint: How eligible loads of the same priority curtail is a local matter. Constraint: Priorities range from 1 to 1000 Constraint: Priority 1 loads curtail last Constraint: Priority 1000 loads curtail earliest	Integer See Clause 6.6.3.5.	[0..1]
queueTimeRemaining	This attribute specifies the duration of time that a dispatchable generator has remaining on either the minimum supply time, the maximum supply time or release time queue. When entered in the queue, this value will be set to the minimumSupplyTime, the (maximumSupplyTime – minimumSupplyTime) or the releaseTime respectively. This value will be down counted at a rate of not less than once per minute until the timer expires. At the expiration time, the generator state will transition to the next state and the queue time reset. Constraint: actualQueueTimeRemaining ≥ 0	Duration See Clause 6.6.1.9.	[0..1]
releaseTime	This attribute specifies the number of seconds that a generator shall be turned off before it is again eligible to supply power. This duration may be used to protect the generator's hardware from excess cycling or meet safety code requirements. It is used to set the queueTimeRemaining attribute when placing the generator in the releaseQueue state. If this attribute is not present, the releaseTime shall be considered to be zero (0) and immediately become available to supply power. Constraint: This attribute's value shall be greater than 0.	Duration See Clause 6.6.1.9.	[0..1]
requestedSupplyLevel	This attribute defines the power level to which the Energy Manager is requesting the generator to operate. The request by the Energy Manager is always a supply amount; however, the generator will compare this supply requested amount to the present supply of the generator and may then a) supply more power, b) reduce the power or c) leave the generator at its present supply state. The request is made in one of four ways defined by the value requestedSupplyLevel.signalType :	FSGIMEventSignalType See Clause 5.7.1.2.1.15.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
	1) When the requestedSupplyLevel.signalType attribute is set to 'setpoint', the requestedSupplyLevel.currentValue attribute describes the absolute amount of supply the generator may deliver. The currentValue must be a positive number or zero. The currentValue is compared against the generatedSupplySetpoint and maximumSupply to determine the adjustment needed to the generation supply setpoint. a) If the currentValue equals the generatedSupplySetpoint, no further adjustment will occur. The supplyStatus will be set to supplyCompliant. b) If the currentValue is zero, the generator will be powered down if the generator has surpassed its minSupplyTime and the supplyStatus will be set to supplyCompliant. If the minSupplyTime has not yet been surpassed, the generator will be powered down as soon as it reaches the minSupplyTime. The supplyStatus will be set to minSupplyQueue until the minSupplyTime is reached where upon the supplyStatus will then be set to supplyCompliant. c) If the currentValue is less than the generatedSupplySetpoint but not zero, the generator setpoint will be reduced if the generator has surpassed its minSupplyTime. If not, the generator will reduce the power setpoint as soon as it reaches the minSupplyTime. The supplyStatus will be set to supplyCompliant. d) If the currentValue is greater than the generatedSupplySetpoint but less than the maximumSupply, the generator setpoint will be increased if the generator has surpassed its minSupplyTime. If not, the generator setpoint will be increased once the generator reaches the minSupplyTime. e) If the currentValue is greater than the maximumSupply, the generator setpoint will be increased to the maximumSupply and the supplyStatus will be set to supplyNoncompliant. 2) When the requestedSupplyLevel.signalType attribute is set to 'delta', the requestedSupplyLevel.currentValue attribute defines the relative amount of supply the generator may deliver. The currentValue may be a positive or negative number. The sum of currentValue and actualGeneratedSupply, is compared against zero and maximumSupply to determine the adjustment needed to the generation supply setpoint. a) If the currentValue is zero no further curtailment or restoration will occur. The supplyStatus will be set to supplyCompliant. b) If the currentValue is positive, the generatedSupplySetpoint will be increased to the sum of the currentValue plus the generatedSupplySetpoint, but not to exceed the maximumSupply. If the sum of the currentValue and the generatedSupplySetpoint is less than the maximumSupply the supplyStatus will be set to supplyCompliant. If the sum of the currentValue and the generatedSupplySetpoint is greater than the maxSupply the supplyStatus will be set to supplyNonCompliant.		

Attribute Name	Description	Attribute Type	Multiplicity
	c) If the currentValue is negative, the generatedSupplySetpoint will be decreased to the sum of the currentValue plus the generatedSupplySetpoint, but not less than zero. If the sum of the currentValue and the generatedSupplySetpoint is greater than zero, the supplyStatus will be set to supplyCompliant. If the sum of the currentValue and the generatedSupplySetpoint is less than zero, generatedSupplySetpoint will be set to zero and the supplyStatus will be set to supplyCompliant. 3) When the requestedSupplyLevel.signalType attribute is set to 'multiplier', the requestedSupplyLevel.currentValue attribute divided by 100 defines the percentage of the generatedSupplySetpoint value to which the generator should be adjusted. That is: $\text{generatedSupplySetpoint} \leq \text{generatedSupplySetpoint} * (\text{currentValue} / 100)$ where the currentValue must be a positive number in the range of 0 to 100. The supplyStatus will be set to supplyCompliant. 4) When the requestedSupplyLevel.signalType attribute is set to 'level', the requestedSupplyLevel.currentValue attribute describes the predefined level to which the generator will be adjusted. The currentValue must be a positive number within the range of 1 to n, where n is defined as the number of entries in the supplyRatingsLevel array. When the request is received, the generator will adjust its generatedSupplySetpoint to the level defined by supplyRatingsLevel(currentValue).supplyAmount. If the value of currentValue is not within the range of n supported by the generator, the request will not be honored.		
supplyCost	This attribute defines the monetary value of the fixed and variable cost associated with supplying power (when positive) or the savings associated with supplying power (when negative) on a per unit of power basis. This attribute is derived from DERCostParameters.OperatingCost, DERCostParameters.StartCost, DERCostParameters.RampCost, and DERCostParameters.StopCost.	Real See Clause 6.6.3.6.	[0..1]
supplyCyclesInPeriod	This attribute defines the number of times the generator has supplied power since this attribute was last reset. This attribute shall be increased by one each time a dispatchable generator becomes operational and supplies power to a load. This attribute is derived from DERUnitGenerator.PeriodStartCount. Constraint: If the maximumCurtailmentsInPeriod attribute is defined, then this attribute shall be defined. Constraint: This attribute's value shall be greater than or equal to 0.	Integer See Clause 6.6.3.5.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
supplyRatingsLevel	This attribute defines the array of supply when the requestedSupplyLevel.signalType equals 'level'. This array transforms the level requested by the Energy Manager as specified by the attribute requestedSupplyLevel.level into the supply to which the generator shall be controlled. This attribute is derived from DERGeneratorRatings.MaximumWOutput, DERGeneratorRatings.MinimumWOutput, DERComplexRamping.StepPs, and DERComplexRamping.StepNg. Constraint: This attribute is required if requestedCurtailmentLevel.signalType may equal 'level'	SupplyArrayElement See Clause 5.5.6.4.	[0..*]
supplyStatus	This attribute indicates whether the generator is presently supplying power and, if so, is in compliance with the power request received. This attribute is derived from DERUnitGenerator.GeneratorOperationStatus and DERUnitStatus.ModeOnConnected.	SupplyStatusType See Clause 5.5.6.6.	[1]

5.5.6.4. SupplyArrayElement (Class)

This class defines a single array element of the supplyRatingsLevel attribute. Each array element defines the fixed supply to which the generator shall be controlled as a function of the stated level. The array is ordered by the supplyRatingsLevel.levelIndex attribute.

UML Element Location: *Model\Device and Model Components\Generator Component\SupplyArrayElement*.

Table 5.26 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
supplyAmount	This attribute defines the absolute supply to be achieved upon receipt of a request from an energy manager as specified in the requestedSupplyLevel.currentValue attribute. When deployed in the supplyRatingsLevel array, this attribute defines the absolute supply the generator should target for each of the defined levels.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
levelIndex	This attribute defines the level or percentage index when used in the supplyRatingsLevel or supplyRatingsMultiplier attributes respectively. When used in the supplyRatingsLevel array, this attribute is matched to the requestedSupplyLevel.currentValue as supplied by the Energy Manager to transform the level request to the absolute supply requested.	Integer See Clause 6.6.3.5.	[1]

5.5.6.5. StorageType (Enumeration)

An enumeration that defines the energy storage characteristics of an instance of the Generator Class.

UML Element Location: *Model\Device and Model Components\Generator Component\StorageType*.

Table 5.27 – StorageType Enumerated Values

Enumerated Value	Definition
none	This value indicates that this instance of the Generator Class models a device that does not produce energy from storage.
electricalStorage	This value indicates that this instance of the Generator Class models a device that produces electricity from storage.
thermalStorage	This value indicates that this instance of the Generator Class models a device that produces thermal energy from storage.

5.5.6.6. SupplyStatusType (Enumeration)

This enumeration indicates if the load is presently curtailed and if curtailed is in compliance with the curtailment request received.

UML Element Location: *Model\Device and Model Components\Generator Component\SupplyStatusType*.

Table 5.28 – SupplyStatusType Enumerated Values

Enumerated Value	Definition
supplyInactive	This generator is not presently operating
supplyRequestPending	A request has been received and is pending.
supplyCompliant	The generator operation is compliant with the last request.
supplyNoncompliant	The generator is not compliant with the last request.

5.6. Energy Manager Component

The Energy Manager Component is the portion of the FSGIM that provides an abstract representation of energy management system functionality. The functionality may be implemented in any device that performs analysis and/or makes energy-related decisions involving meters, loads, generators, or energy storage devices. The Energy Manager Component includes a representation of information used to make these decisions, including cost, emissions, and other information. The Energy Manager Component also supports tracking historical energy data, projecting future energy demand, and interfacing to third party sources of energy related information such as weather.

An energy manager fulfills two roles in a facility:

1. Managing/monitoring the energy aspects of some portion of a facility and
2. interfacing with entities outside a facility.

An energy manager that only fulfills the first role is called a "local energy manager" or "local EM." An energy manager that interfaces outside of the facility, regardless of any local energy management functions, is called an "energy services interface energy manager" or "ESI EM."

UML Package Location: *Model\Device and Model Components\Energy Manager Component*

5.6.1. EMTopLevel Diagram

This diagram presents a summary overview of the Energy Manager.

A key feature of the EM Class is to enable the packaging of sequences of time-delimited data. The top section of the diagram shows how this data can be either historical or forecast.

The middle section of the diagram shows how the EM class maintains associations with loads, meters, and generator components.

Finally, the bottom section shows how the EM data is contained in two principle classes -- EMPresentData for present data sets, and EMIntervalData for historic or projected time-delimited data. Shown are EnergySupplierData and UsageSummary classes, additional data classes defined in FSGIM can be found in the more detailed diagram "EM, PresentData, and EMIntervalData".

The balance of this package provides detailed descriptions of these classes and relationships.

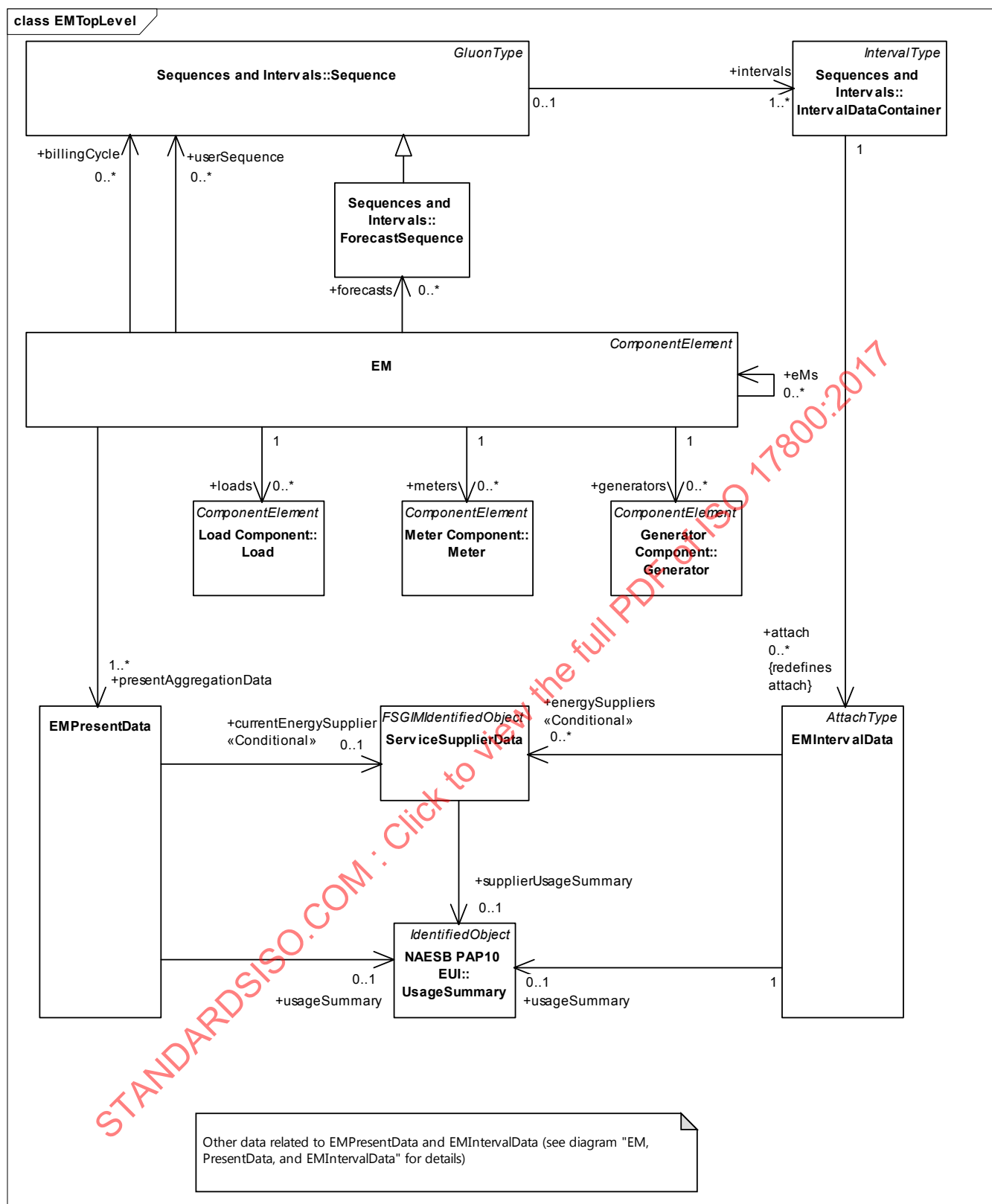


Figure 5.12 - EMTopLevel

The EMTopLevel Diagram makes use of the following elements:

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

EMIntervalData (Class)

This class represents data in the domain of concern of an Energy Manager over some complete interval of time. This time interval may be in the past or future.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMIntervalData*. See Clause 5.6.7.4 for the complete definition of this element.

EMPresentData (Class)

The container that directly holds all of the aggregated present data and indirectly holds the associated present context.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPresentData*. See Clause 5.6.7.7 for the complete definition of this element.

ForecastSequence (Class)

The Sequence that is used for storing all forecasts, including forecasts for time periods where the originally forecasted time period has now passed.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\ForecastSequence*. See Clause 5.6.7.10.2 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

IntervalDataContainer (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval. This is a specialization of the WSCalendar PIM IntervalType.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\IntervalDataContainer*. See Clause 5.6.7.10.3 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be surmised by monitoring the time references.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*. See Clause 5.6.7.10.4 for the complete definition of this element.

ServiceSupplierData (Class)

This class contains information associated with the generated emissions, quality of power, and energy-price units associated with an identified supplier of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\ServiceSupplierData*. See Clause 5.6.7.9 for the complete definition of this element.

UsageSummary (Class)

Summary of usage for a billing period.

Note: This model was extended using the work of OpenADE 2015 for ESPI. These extensions are proposed for the next revision of the NAESB standard REQ.21. Also extended are SummaryMeasurement and LineDetail.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\UsageSummary. See Clause 5.7.5.2.1.27 for the complete definition of this element.

5.6.2. Sequences and Intervals

This clause supports the need to relate facility management data with time intervals and sequences of time intervals. These sequences of intervals can be either forecasts or historical data.

UML Package Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals*

5.6.2.1. Sequences and Intervals Diagram

This diagram illustrates the structure of intervals and the relationships between intervals used in the FSGIM. The EM Class supports sequences of intervals for billing cycles, forecasts, and user determined other needs. The intervals in the sequence may be time-ordered (SimpleInterval). Alternatively, they may enable the expression of complex relationships between intervals (FullInterval). These intervals represent time delimited containers of data. Examples of simple intervals are shown in Annex D.

The portion of the diagram enclosed by the dashed box is derived from a platform independent model (PIM) of WS-Calendar (OASIS WS-Calendar v1.0). A PIM is a model of a system designed to be independent of the details needed for particular implementation technologies.

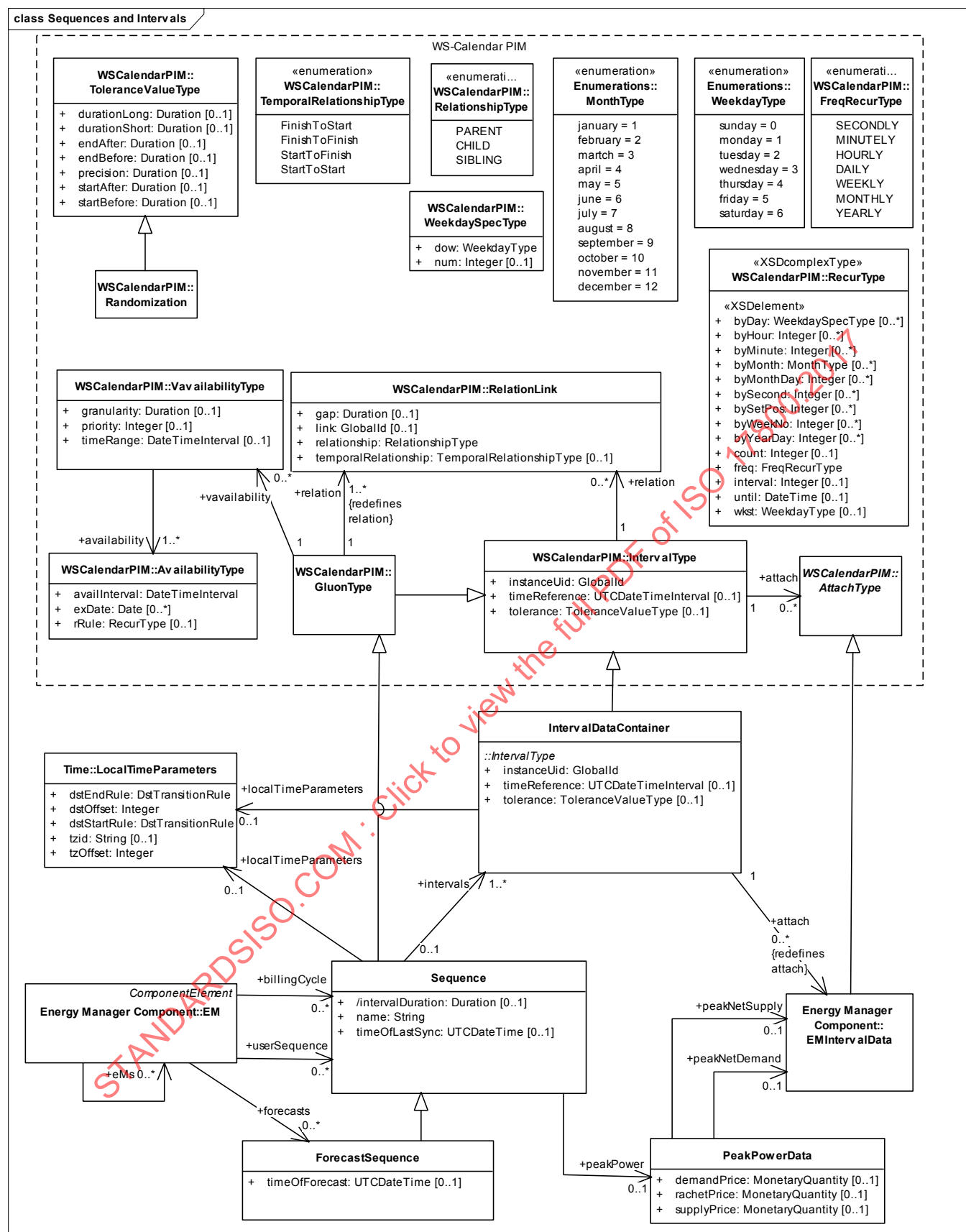


Figure 5.13 - Sequences and Intervals

The Sequences and Intervals Diagram makes use of the following elements:

AttachType (Abstract Class)

This class represents data in the domain of concern over some interval of time.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WScalendarPIM\AttachType*. See Clause 5.7.3.2.4 for the complete definition of this element.

AvailabilityType (Class)

This class expresses the range of times in which an Interval or Sequence can be Scheduled.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WScalendarPIM\AvailabilityType*. See Clause 5.7.3.2.5 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

EMIntervalData (Class)

This class represents data in the domain of concern of an Energy Manager over some complete interval of time. This time interval may be in the past or future.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMIntervalData*. See Clause 5.6.7.4 for the complete definition of this element.

ForecastSequence (Class)

The Sequence that is used for storing all forecasts, including forecasts for time periods where the originally forecasted time period has now passed.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\ForecastSequence*. See Clause 5.6.7.10.2 for the complete definition of this element.

FreqRecurType (Enumeration)

Used to identify the unit of time to be used in specifying the frequency of a recurring event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WScalendarPIM\FreqRecurType*. See Clause 5.7.3.2.11 for the complete definition of this element.

GluonType (Class)

A GluonType influences the serialization of IntervalTypes in a sequence. A GluonType is often used as a pointer to the beginning interval in a sequence, although in general it can point to any interval in a sequence, or simply describe an isolated interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WScalendarPIM\GluonType*. See Clause 5.7.3.2.6 for the complete definition of this element.

IntervalDataContainer (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval. This is a specialization of the WScalendar PIM IntervalType.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\IntervalDataContainer*. See Clause 5.6.7.10.3 for the complete definition of this element.

IntervalType (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WScalendarPIM\IntervalType*. See Clause 5.7.3.2.7 for the complete definition of this element.

LocalTimeParameters (Class)

Contains attributes related to the configuration of the time service.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\LocalTimeParameters*. See Clause 6.6.1.5 for the complete definition of this element.

MonthType (Enumeration)

An enumerated list of months of the year.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\MonthType*. See Clause 6.6.2.2 for the complete definition of this element.

PeakPowerData (Class)

This class contains information associated with peak power intervals within a Sequence.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\PeakPowerData*. See Clause 5.6.7.10.1 for the complete definition of this element.

Randomization (Class)

A specialized version of ToleranceValueType used to indicate the degree of randomization required. In this use, the startafter and startbefore attributes indicate the range of the starting times.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WCalendarPIM\Randomization*. See Clause 5.7.3.2.1 for the complete definition of this element.

RecurType (Class)

This value type is used to identify properties that contain a recurrence rule specification.

From RFC5545 section 3.3.10. Recurrence Rule.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WCalendarPIM\RecurType*. See Clause 5.7.3.2.2 for the complete definition of this element.

RelationLink (Class)

The RelationLink class serves to link a set of intervals together, in support of a linked-list method of creating a sequence. It is also used to link a GluonType instance to a specific interval within a sequence, typically used to point to the beginning interval of a sequence.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WCalendarPIM\RelationLink*. See Clause 5.7.3.2.8 for the complete definition of this element.

RelationshipType (Enumeration)

Specifies the type of hierarchical relationship associated with the calendar component specified by the property. The parameter value can be PARENT, to indicate that the referenced calendar component is a superior of calendar component; CHILD to indicate that the referenced calendar component is a subordinate of the calendar component; or SIBLING to indicate that the referenced calendar component is a peer of the calendar component. If this parameter is not specified on an allowable property, the default relationship type is PARENT.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WCalendarPIM\RelationshipType*. See Clause 5.7.3.2.12 for the complete definition of this element.

Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be surmised by monitoring the time references.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*. See Clause 5.6.7.10.4 for the complete definition of this element.

TemporalRelationshipType (Enumeration)

The temporal relation of one interval to the interval it is linked to.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\TemporalRelationshipType*. See Clause 5.7.3.2.13 for the complete definition of this element.

ToleranceValueType (Class)

A tolerance value is a set of durations which indicate the allowed tolerance for the indicated value. Note that there may be many tolerance values associated with a Duration.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\ToleranceValueType*. See Clause 5.7.3.2.9 for the complete definition of this element.

VavailabilityType (Class)

The VavailabilityType class is intended to establish a baseline range of time to scope a sequence. The class can specify a start time and an end time or duration, using the timeRange attribute. If "start" is not present, then the start time is unbounded. If "end" or "duration" is not present, then the end time is unbounded. Within the specified time period, availability defaults to "not available", except for any time periods corresponding to associated AvailabilityType instances.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\VavailabilityType*. See Clause 5.7.3.2.10 for the complete definition of this element.

WeekdaySpecType (Class)

Specifies a week day and a recurrence number. For example, within a MONTHLY rule:

+1,MO represents the first Monday within the month, whereas -1,MO represents the last Monday of the month.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\WeekdaySpecType*. See Clause 5.7.3.2.3 for the complete definition of this element.

WeekdayType (Enumeration)

An enumerated list of day of the week.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\WeekdayType*. See Clause 6.6.2.4 for the complete definition of this element.

5.6.3. Collections, Rule Sets, and Aggregations

Facilities will need to record and report total values for energy related quantities such as energy used, net demand, or emissions. Facility managers will want to identify and manage collections of entities within the facility that contribute to these total values.

There are three major concepts involved:

- Collection – a set of entities that is identified as a group because the members share some trait;
- Ruleset – an expression of directives and constraints used to filter a Collection or derive an aggregated value; and
- Aggregation – a value derived by combining several other values according to a Ruleset.

Some examples of Collections and Aggregations include:

- At the facility level: total demand for all of the equipment in the facility;
- At the building level: the energy output from the photovoltaic arrays for the day; and
- At the floor level: total energy used by loads on the fourth floor during the month.

UML Package Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations*

5.6.3.1. Collection and Aggregation Diagram

This diagram illustrates the constructs provided to support collections and aggregate quantities. A Collection has an attribute, part, that is used to identify its members. The Collection members can be instances of the Meter Class,

Generator Class, EM Class, Load Class, or other instances of the Collection Class. A FilteredCollection derives its members by applying a filtering Ruleset to one or more input Collections. An Aggregation Class associates an AggregateQuantity with the Collection contributing to the aggregate quantity and the Ruleset governing its calculation.

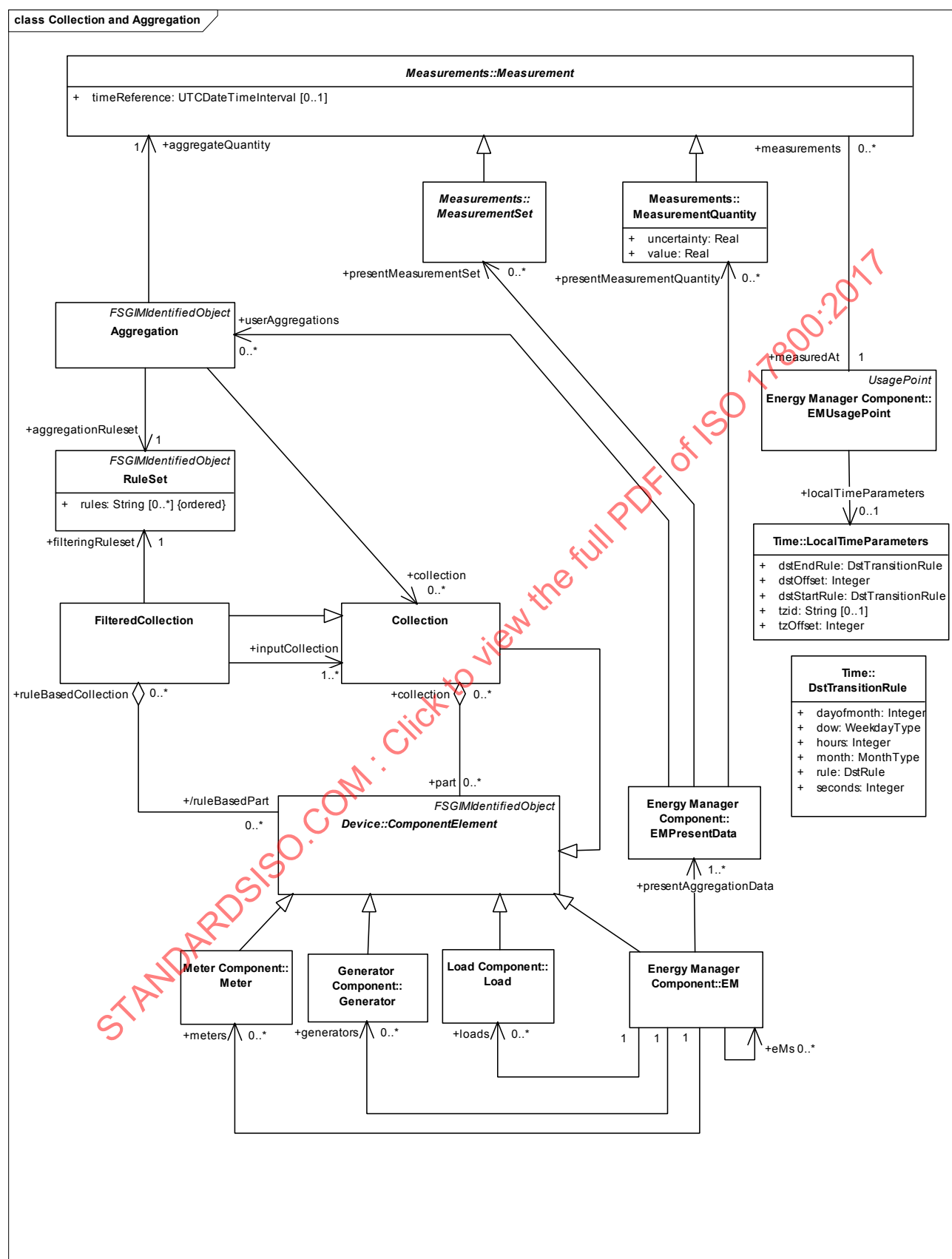


Figure 5.14 - Collection and Aggregation

The Collection and Aggregation Diagram makes use of the following elements:

Aggregation (Class)

An aggregation allows the assembly of statistics of interest arrived at through defined algorithms applied to other parts of the model. The description of an aggregation includes all of the inputs that comprise the algorithms as well as the outputs of the algorithm. An aggregation instance represents one or more aggregateQuantities that are computed by applying aggregation rules for each quantity against a Collection or FilteredCollection of Components.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Aggregation*. See Clause 5.6.7.11.1 for the complete definition of this element.

Collection (Class)

This class defines a set of entities that is identified as a group because the members share some trait. A Collection can contain other Collections.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Collection*. See Clause 5.6.7.11.3 for the complete definition of this element.

ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*. See Clause 5.2.3.1 for the complete definition of this element.

DstTransitionRule (Class)

Rules for DST transition.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\DstTransitionRule*. See Clause 6.6.1.2 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

EMPresentData (Class)

The container that directly holds all of the aggregated present data and indirectly holds the associated present context.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPresentData*. See Clause 5.6.7.7 for the complete definition of this element.

EMUsagePoint (Class)

Logical point on a network at which consumption or production is either measured (e.g., metered) or estimated (e.g., unmetered street lights).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMUsagePoint*. See Clause 5.6.7.8 for the complete definition of this element.

FilteredCollection (Class)

A FilteredCollection "filters" one or more inputCollections according to a set of filteringRules to produce a new instance of a Collection with the subset defined by the Ruleset.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\FilteredCollection*. See Clause 5.6.7.11.4 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

LocalTimeParameters (Class)

Contains attributes related to the configuration of the time service.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\LocalTimeParameters*. See Clause 6.6.1.5 for the complete definition of this element.

Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*. See Clause 6.6.4.1 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

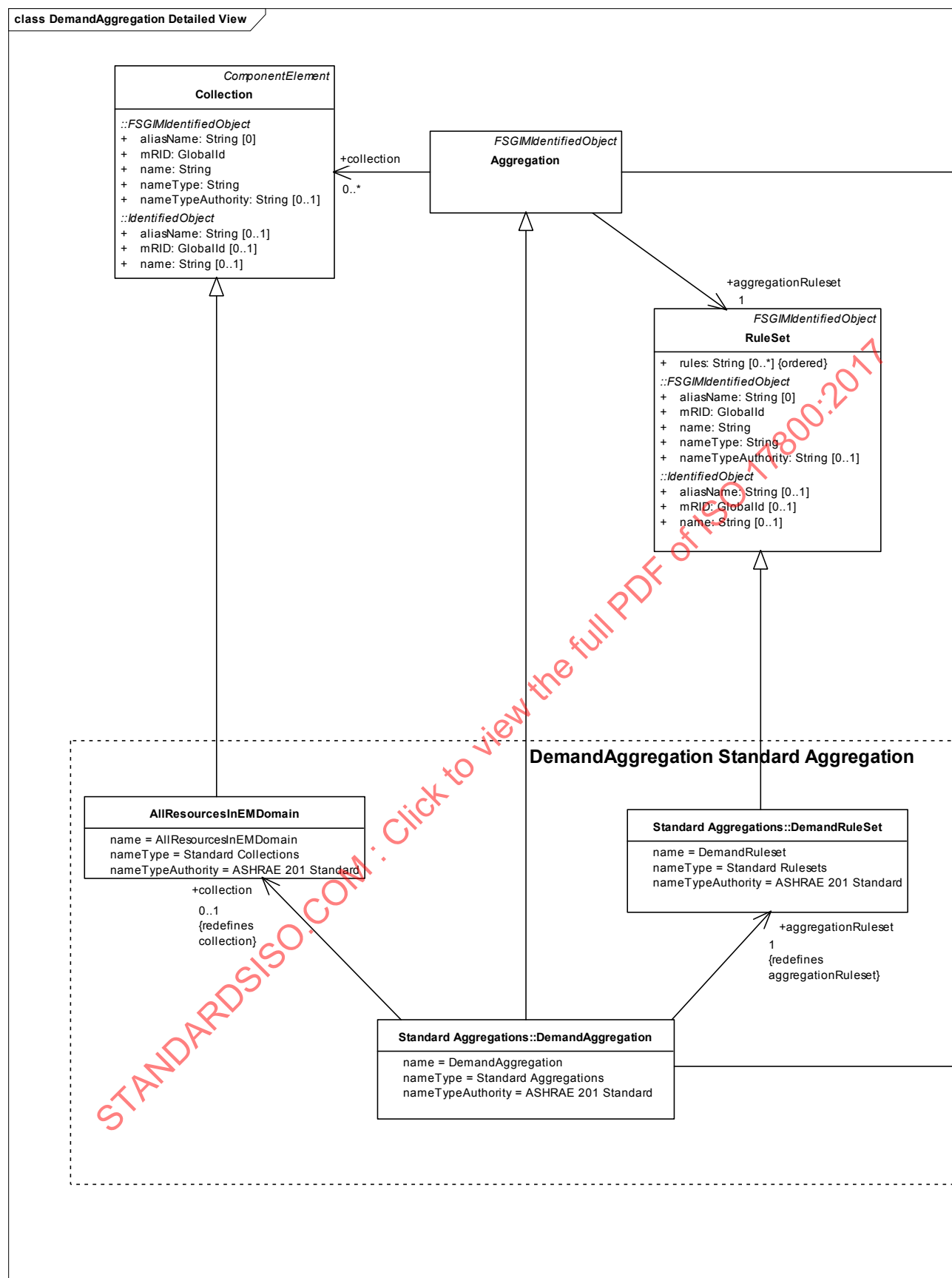
5.6.3.2. DemandAggregation Detailed View Diagram

The FSGIM creates a number of Standard Aggregations along with their associated Standard Rulesets. All of these Standard Aggregations and Standard Rulesets are derived from the general Aggregation/Collection/Ruleset model.

An example showing one of these derivations is shown in the following figure. The portion of the diagram that is not enclosed in a dotted box are taken directly from the general Aggregation/Collection/Ruleset model. The portion of the diagram enclosed in the dotted box show how a DemandAggregation is determined from the AllResourcesInEMDomain Collection and the DemandRuleset.

Since all Standard Aggregations, Standard Collections, and Standard Rulesets follow this same basic pattern, only the portion of the diagram in the dotted box is shown when documenting each of the Standard Aggregations and their associated Standard Rulesets.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



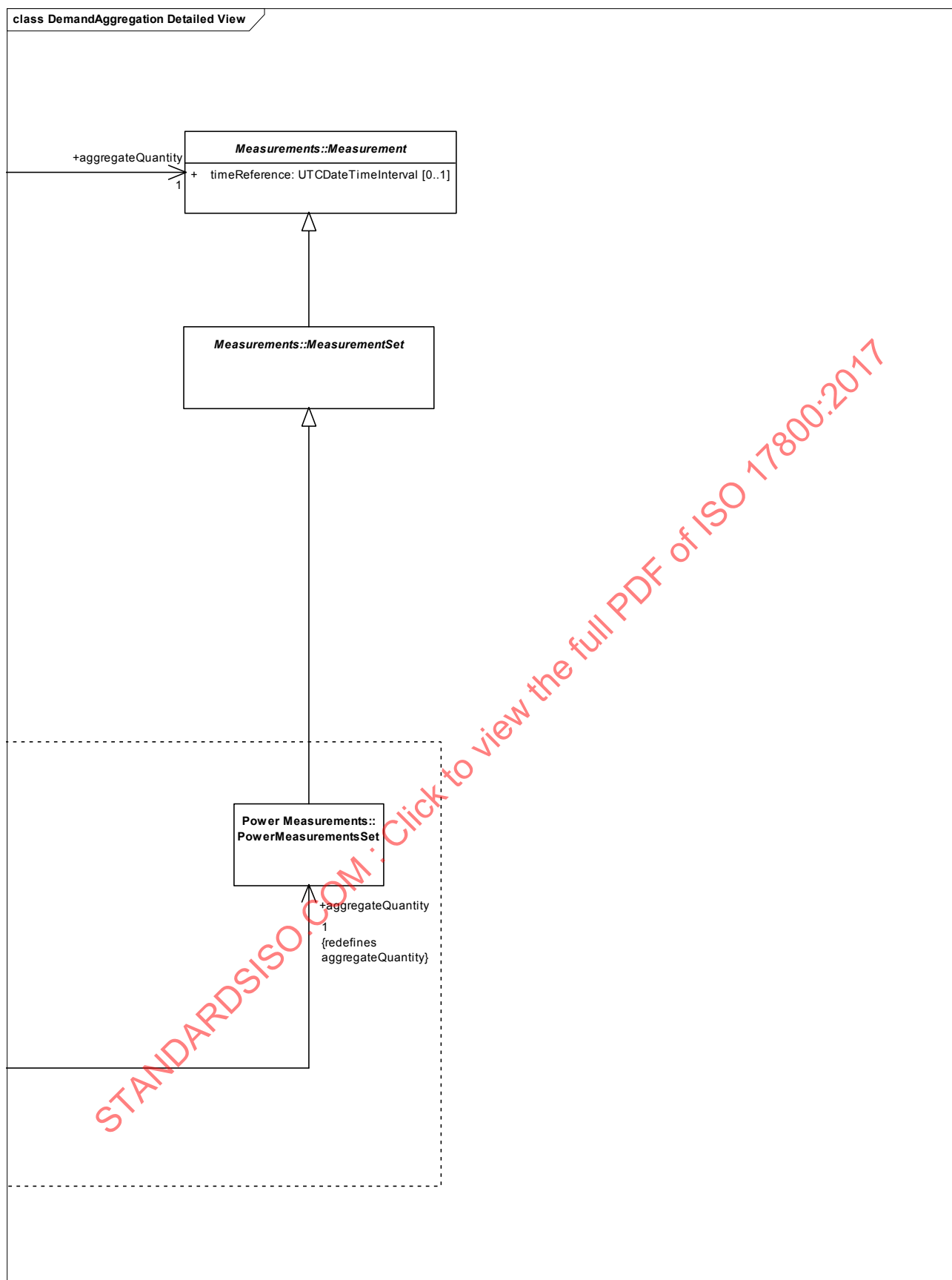


Figure 5.15 - DemandAggregation Detailed View

The DemandAggregation Detailed View Diagram makes use of the following elements:

Aggregation (Class)

An aggregation allows the assembly of statistics of interest arrived at through defined algorithms applied to other parts of the model. The description of an aggregation includes all of the inputs that comprise the algorithms as well as the outputs of the algorithm. An aggregation instance represents one or more aggregateQuantities that are computed by applying aggregation rules for each quantity against a Collection or FilteredCollection of Components.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Aggregation*. See Clause 5.6.7.11.1 for the complete definition of this element.

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

Collection (Class)

This class defines a set of entities that is identified as a group because the members share some trait. A Collection can contain other Collections.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Collection*. See Clause 5.6.7.11.3 for the complete definition of this element.

DemandAggregation (Class)

For historical aggregations, this is the aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate present subinterval demand of loads directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\DemandAggregation*. See Clause 5.6.7.11.6.9 for the complete definition of this element.

DemandRuleSet (Class)

Calculate "metered demand":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Load Class (including instances of the CurtailableLoad Class) and/or
 2. direct subordinate instances of the EM Class where

EM.presentAggregationData.aggregationMetadata.hasElectricalGenerators is False
are connected to its output ConnectionPoint and where ElectricMeter.powerReading is available, sum
ElectricMeter.powerReading.

Calculate "demand from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered demand" and where Load.actualDemand is available, sum
Load.actualDemand.

Calculate "demand from energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered demand" and where EMPresentData.presentAggregateDemand is available, sum the subordinate
EMPresentData.presentAggregateDemand.aggregateQuantity.

Calculate final result:

- Sum "metered demand", "demand from unmetered loads", and "demand from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\DemandRuleSet*. See Clause 5.6.7.11.6.10 for the complete definition of this element.

Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*. See Clause 6.6.4.1 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

5.6.3.3. Standard Aggregations

This clause documents the Standard Aggregations and associated Standard Rulesets that are prescribed in the FSGIM.

Note: In the following diagrams, the rules that make up the standard Rulesets are shown in the Notes field to make the diagrams more compact.

Use of the Standard Aggregations requires a logical tree structure for Energy Managers consistent with the structure of the circuits in the facility. In order to get a Standard Aggregation that correctly represents an entire Circuit, the designer must ensure that there is one Energy Manager with direct or indirect visibility of all contributors to that Circuit.

For quantities to be suitable for standard aggregations, they must have consistent values for their measurement time reference, meaning the measurements must have the same start times and durations. When it is not practical to precisely align time references, it is a local matter to decide how to adapt actual measurement values to make them consistent.

The standard rulesets are written from the perspective of present data. When applying the ruleset to historical or forecast data, substitutions of attributes from the EMIntervalData Class instead of the EMPresentData Class are required.

For example, the DemandRuleset refers to

“EM.presentAggregationData.aggregationMetadata.hasElectricalGenerators.”

For a historical sequence, this is replaced by

“EM.billingCycle.intervals.attach.aggregationMetadata.hasElectricalGenerators.”

UML Package Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations*

5.6.3.3.1. AdjustedFullDRDemandAggregation Diagram

This diagram shows how the AdjustedFullDRDemandAggregation is derived by applying the AdjustedFullDRDemandRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as a PowerMeasurementsSet.

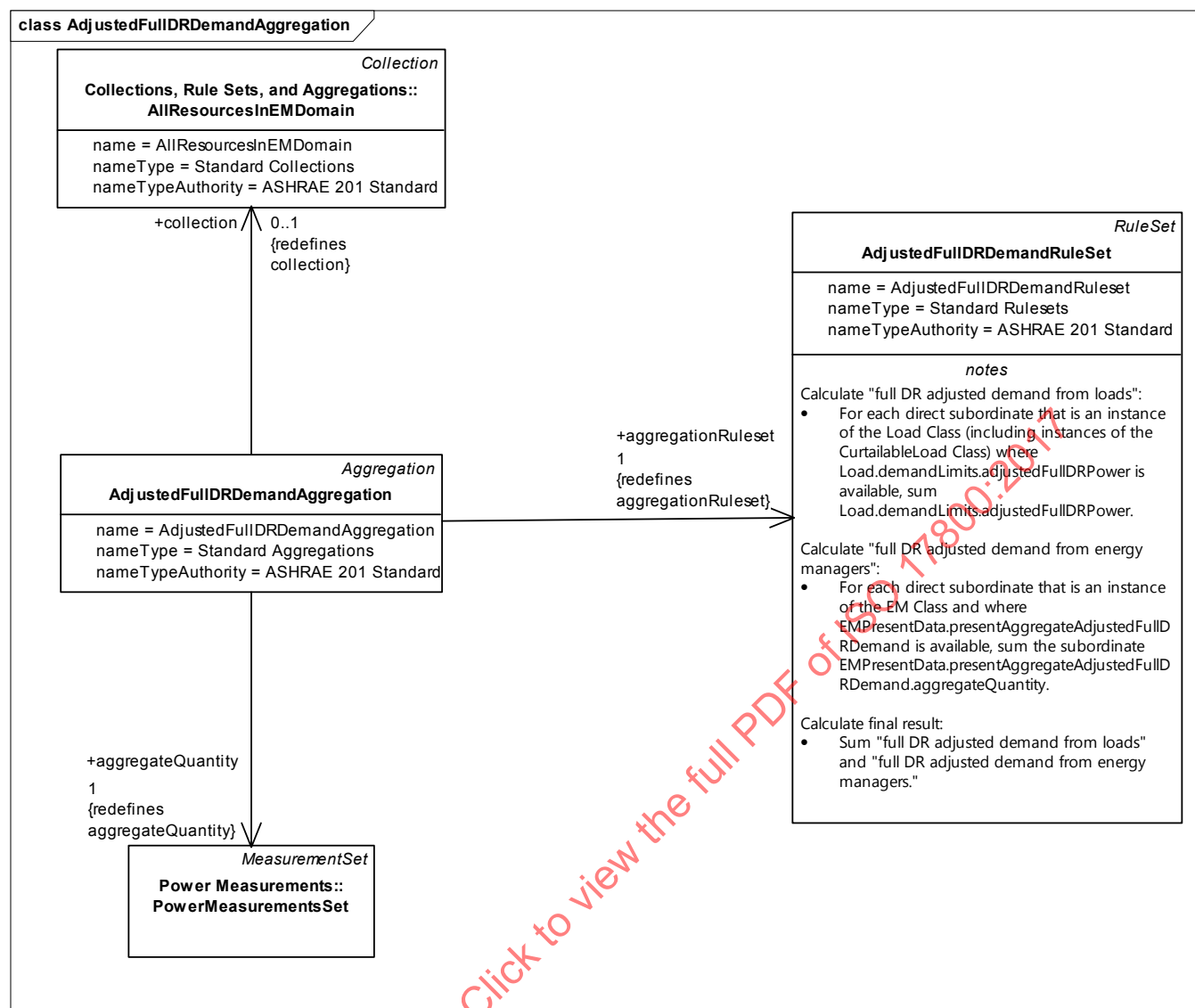


Figure 5.16 - AdjustedFullDRDemandAggregation

The AdjustedFullDRDemandAggregation Diagram makes use of the following elements:

AdjustedFullDRDemandAggregation (Class)

For historical aggregations, this is the aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval assuming the largest acceptable response to demand response events occurred.

For a partially completed (filling) demand interval, this is the expected demand of loads directly or indirectly managed by the associated EM as if the largest acceptable response to a demand response event was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads, with the largest acceptable demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

For example, in a production line this could be the demand of the production line when it was providing the largest acceptable demand reduction in response to a DR request. For an HVAC system, this could be the demand that still meets code requirements and acceptable comfort conditions assuming a certain set of weather conditions and occupancy levels.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRDemandAggregation*. See Clause 5.6.7.11.6.1 for the complete definition of this element.

AdjustedFullDRDemandRuleSet (Class)

Calculate "full DR adjusted demand from loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) where Load.demandLimits.adjustedFullDRPower is available, sum Load.demandLimits.adjustedFullDRPower.

Calculate "full DR adjusted demand from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedFullDRDemand is available, sum the subordinate EMPresentData.presentAggregateAdjustedFullDRDemand.aggregateQuantity.

Calculate final result:

- Sum "full DR adjusted demand from loads" and "full DR adjusted demand from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRDemandRuleSet*. See Clause 5.6.7.11.6.2 for the complete definition of this element.

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

5.6.3.3.2. AdjustedFullDRSupplyAggregation Diagram

This diagram shows how the AdjustedFullDRSupplyAggregation is derived by applying the AdjustedFullDRSupplyRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as a PowerMeasurementsSet.

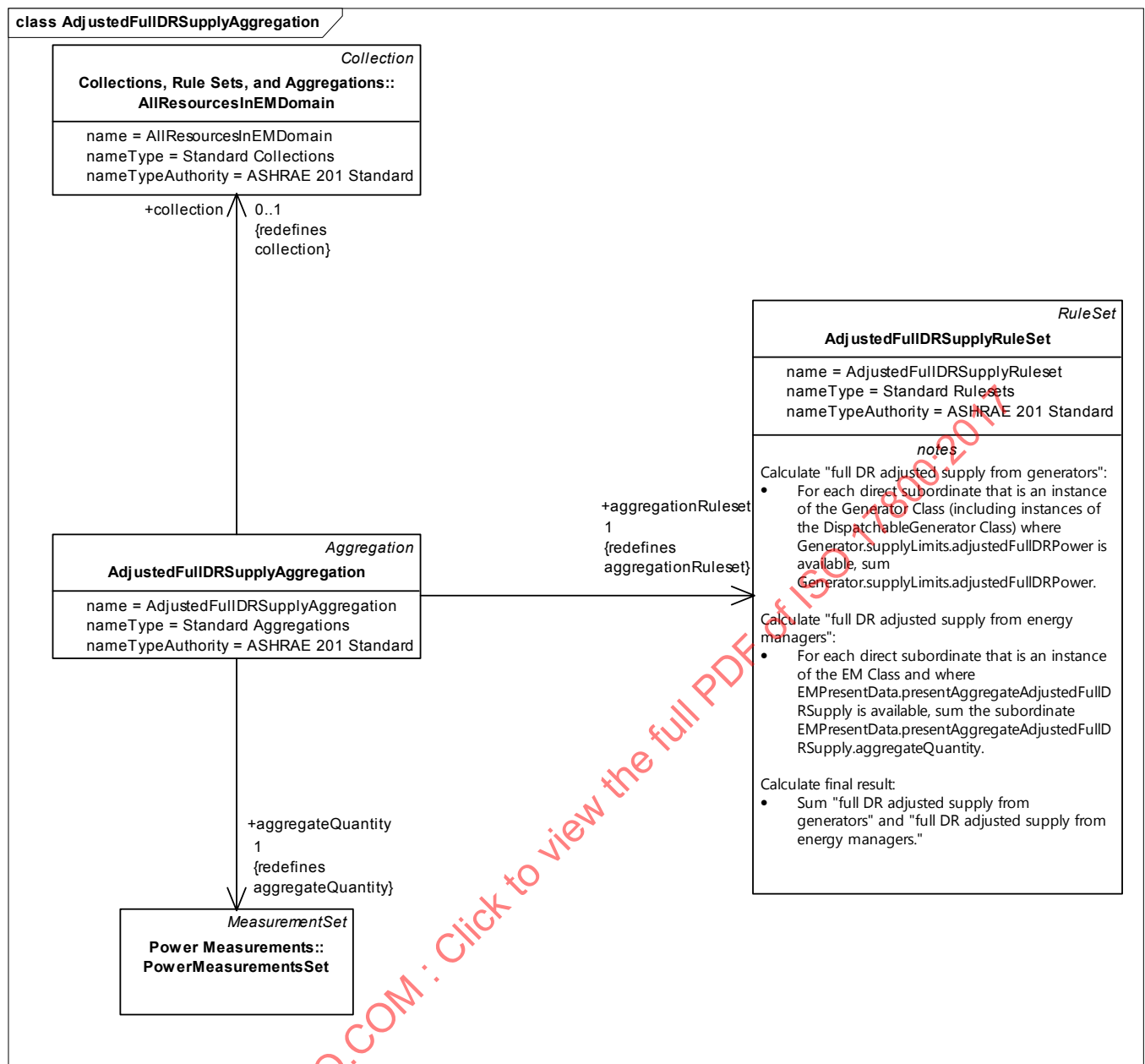


Figure 5.17 - AdjustedFullDRSupplyAggregation

The AdjustedFullDRSupplyAggregation Diagram makes use of the following elements:

AdjustedFullDRSupplyAggregation (Class)

For historical aggregations, this is the aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval assuming the largest acceptable response to demand response events occurred.

For a partially completed (filling) demand interval, this is the expected supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM as if the largest acceptable response to a demand response event was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of electrical generators and electrical storage devices, with the largest acceptable demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRSupplyAggregation*. See Clause 5.6.7.11.6.3 for the complete definition of this element.

AdjustedFullDRSupplyRuleSet (Class)

Calculate "full DR adjusted supply from generators":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.supplyLimits.adjustedFullDRPower is available, sum Generator.supplyLimits.adjustedFullDRPower.

Calculate "full DR adjusted supply from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedFullDRSupply is available, sum the subordinate EMPresentData.presentAggregateAdjustedFullDRSupply.aggregateQuantity.

Calculate final result:

- Sum "full DR adjusted supply from generators" and "full DR adjusted supply from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRSupplyRuleSet*. See Clause 5.6.7.11.6.4 for the complete definition of this element.

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

5.6.3.3.3. AdjustedNoDRDemandAggregation Diagram

This diagram shows how the AdjustedNoDRDemandAggregation is derived by applying the AdjustedNoDRDemandRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as a PowerMeasurementsSet.

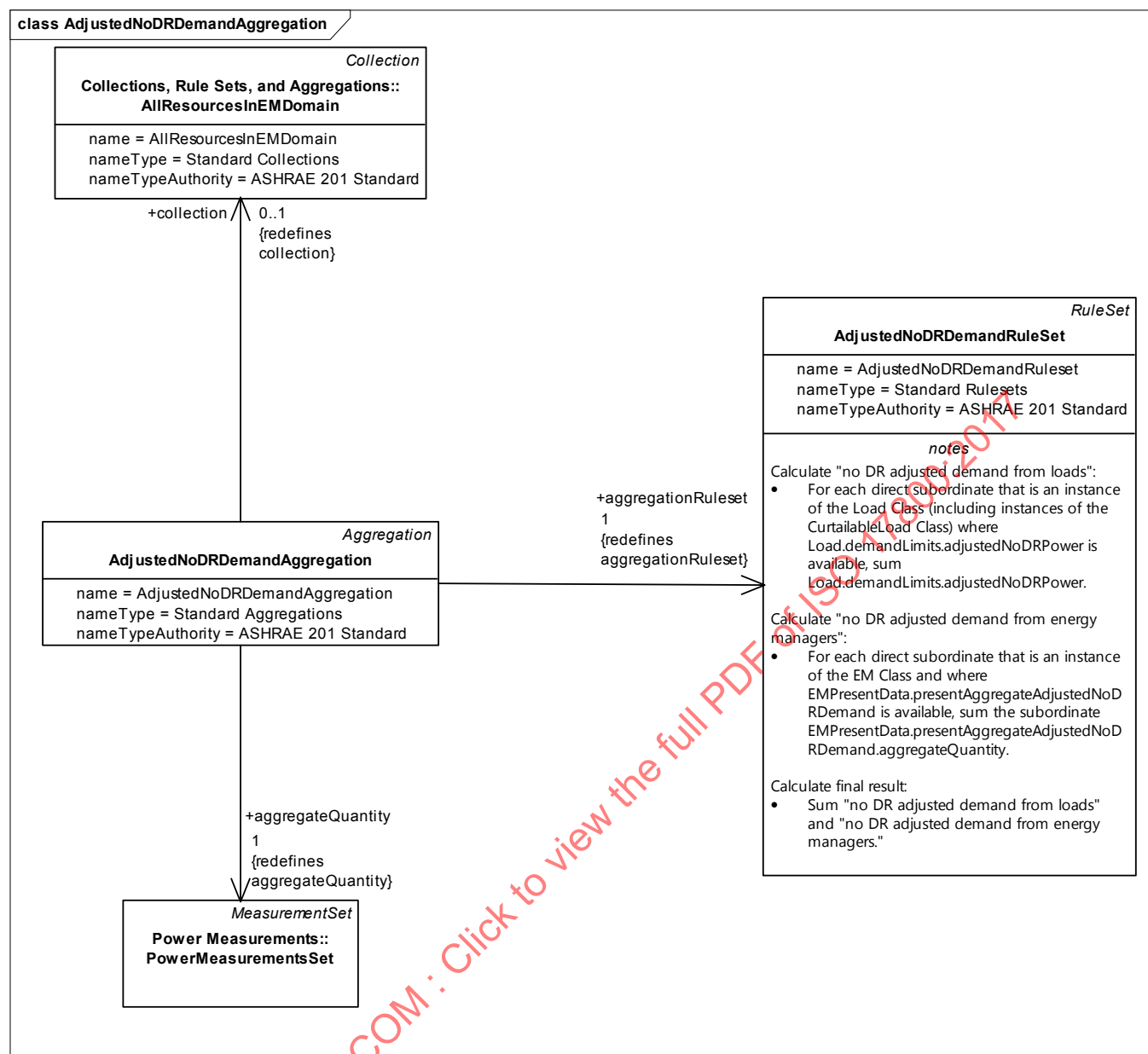


Figure 5.18 - AdjustedNoDRDemandAggregation

The AdjustedNoDRDemandAggregation Diagram makes use of the following elements:

AdjustedNoDRDemandAggregation (Class)

For historical aggregations, this is the aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval assuming no demand response events occurred.

For a partially completed (filling) demand interval, this is the expected demand of loads directly or indirectly managed by the associated EM as if no demand response was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads, without demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

For example, in a production line this could be the expected average demand of the production line assuming that the production line was operating at a certain capacity. For an HVAC system, this could be the expected average demand assuming a certain set of weather conditions and occupancy levels.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRDemandAggregation*. See Clause 5.6.7.11.6.5 for the complete definition of this element.

AdjustedNoDRDemandRuleSet (Class)

Calculate "no DR adjusted demand from loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) where Load.demandLimits.adjustedNoDRPower is available, sum Load.demandLimits.adjustedNoDRPower.

Calculate "no DR adjusted demand from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedNoDRDemand is available, sum the subordinate EMPresentData.presentAggregateAdjustedNoDRDemand.aggregateQuantity.

Calculate final result:

- Sum "no DR adjusted demand from loads" and "no DR adjusted demand from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRDemandRuleSet*. See Clause 5.6.7.11.6.6 for the complete definition of this element.

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

5.6.3.3.4. AdjustedNoDRSupplyAggregation Diagram

This diagram shows how the AdjustedNoDRSupplyAggregation is derived by applying the AdjustedNoDRSupplyRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as a PowerMeasurementsSet.

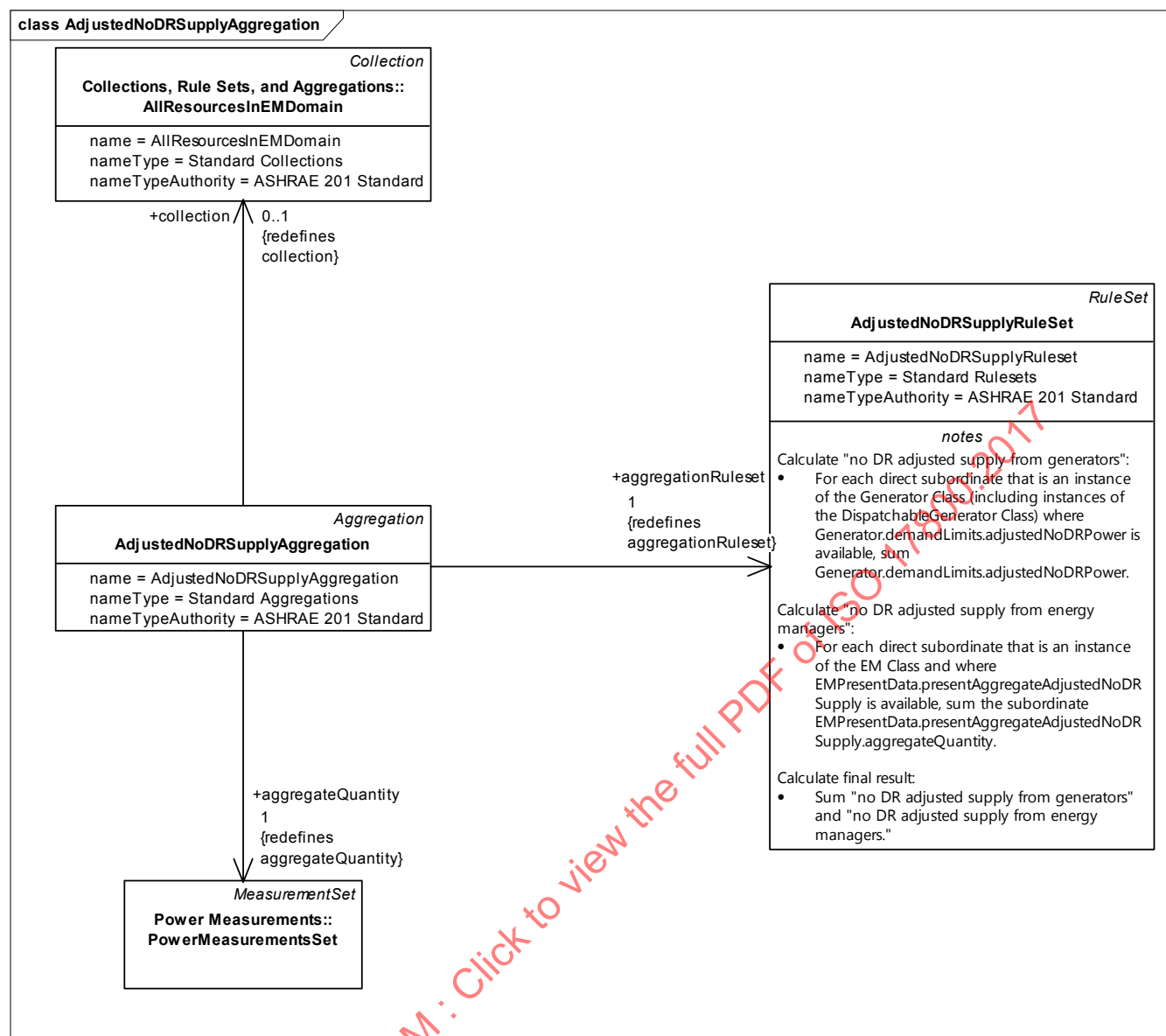


Figure 5.19 - AdjustedNoDRSupplyAggregation

The AdjustedNoDRSupplyAggregation Diagram makes use of the following elements:

AdjustedNoDRSupplyAggregation (Class)

For historical aggregations, this is the aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval assuming no demand response events occurred.

For a partially completed (filling) demand interval, this is the expected supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM as if no demand response was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate supply of electrical generators and electrical storage devices, without demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRSupplyAggregation*. See Clause 5.6.7.11.6.7 for the complete definition of this element.

AdjustedNoDRSupplyRuleSet (Class)

Calculate "no DR adjusted supply from generators":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.demandLimits.adjustedNoDRPower is available, sum Generator.demandLimits.adjustedNoDRPower.

Calculate "no DR adjusted supply from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedNoDRSupply is available, sum the subordinate EMPresentData.presentAggregateAdjustedNoDRSupply.aggregateQuantity.

Calculate final result:

- Sum "no DR adjusted supply from generators" and "no DR adjusted supply from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRSupplyRuleSet*. See Clause 5.6.7.11.6.8 for the complete definition of this element.

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

5.6.3.3.5. DemandAggregation Diagram

This diagram shows how the DemandAggregation is derived by applying the DemandRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as a PowerMeasurementsSet.

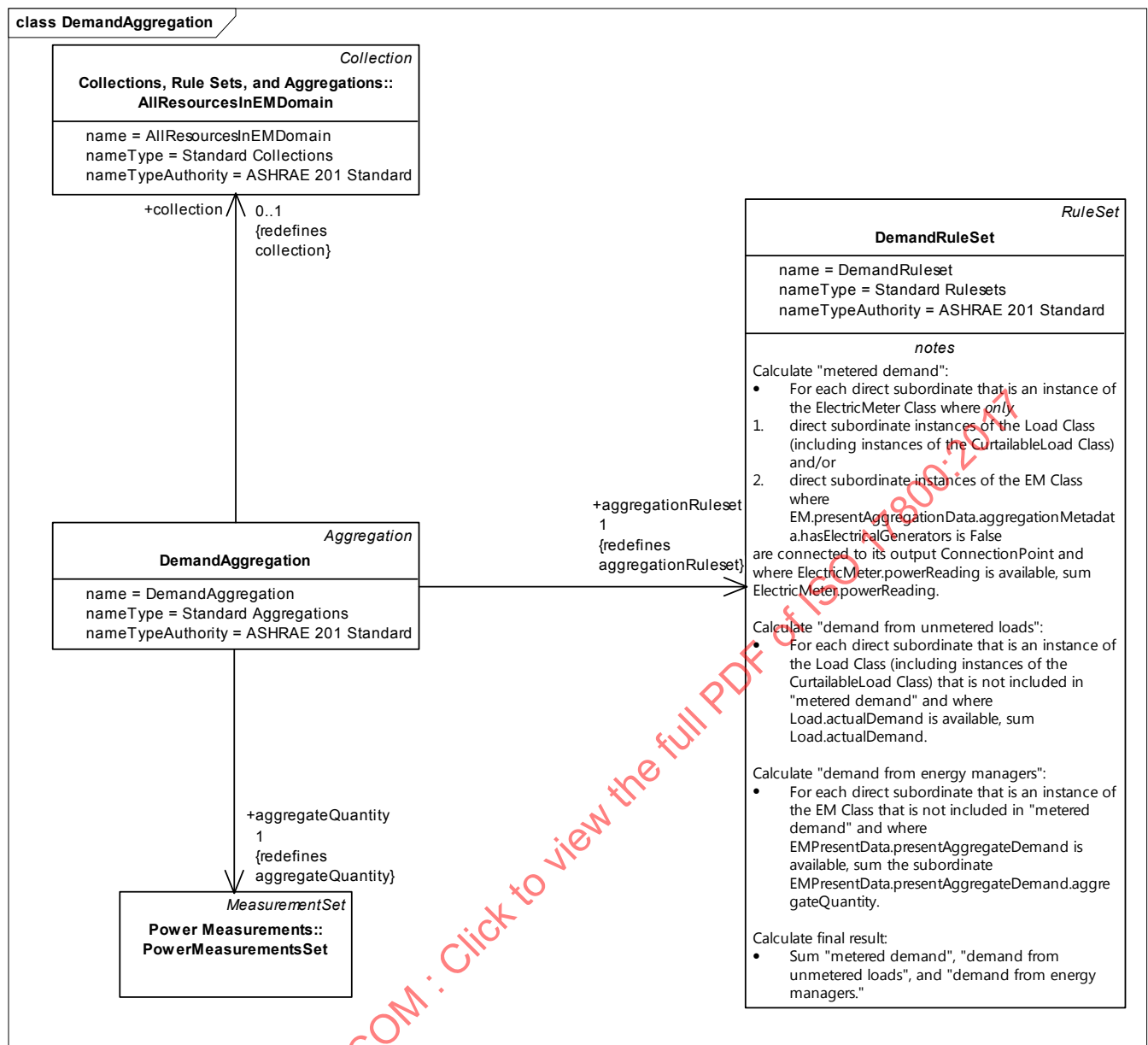


Figure 5.20 - DemandAggregation

The DemandAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

DemandAggregation (Class)

For historical aggregations, this is the aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate present subinterval demand of loads directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\DemandAggregation*. See Clause 5.6.7.11.6.9 for the complete definition of this element.

DemandRuleSet (Class)

Calculate "metered demand":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Load Class (including instances of the CurtailableLoad Class) and/or
 2. direct subordinate instances of the EM Class where
EM.presentAggregationData.aggregationMetadata.hasElectricalGenerators is False

are connected to its output ConnectionPoint and where ElectricMeter.powerReading is available, sum ElectricMeter.powerReading.

Calculate "demand from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered demand" and where Load.actualDemand is available, sum Load.actualDemand.

Calculate "demand from energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered demand" and where EMPresentData.presentAggregateDemand is available, sum the subordinate
EMPresentData.presentAggregateDemand.aggregateQuantity.

Calculate final result:

- Sum "metered demand", "demand from unmetered loads", and "demand from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\DemandRuleSet*. See Clause 5.6.7.11.6.10 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

5.6.3.3.6. ElectricalEnergyStoredAggregation Diagram

This diagram shows how the ElectricalEnergyStoredAggregation is derived by applying the ElectricalEnergyStoredRuleSet to the ElectricalEnergyStoredCollection to obtain an aggregateQuantity that is represented as an EnergyMeasurementsSet.

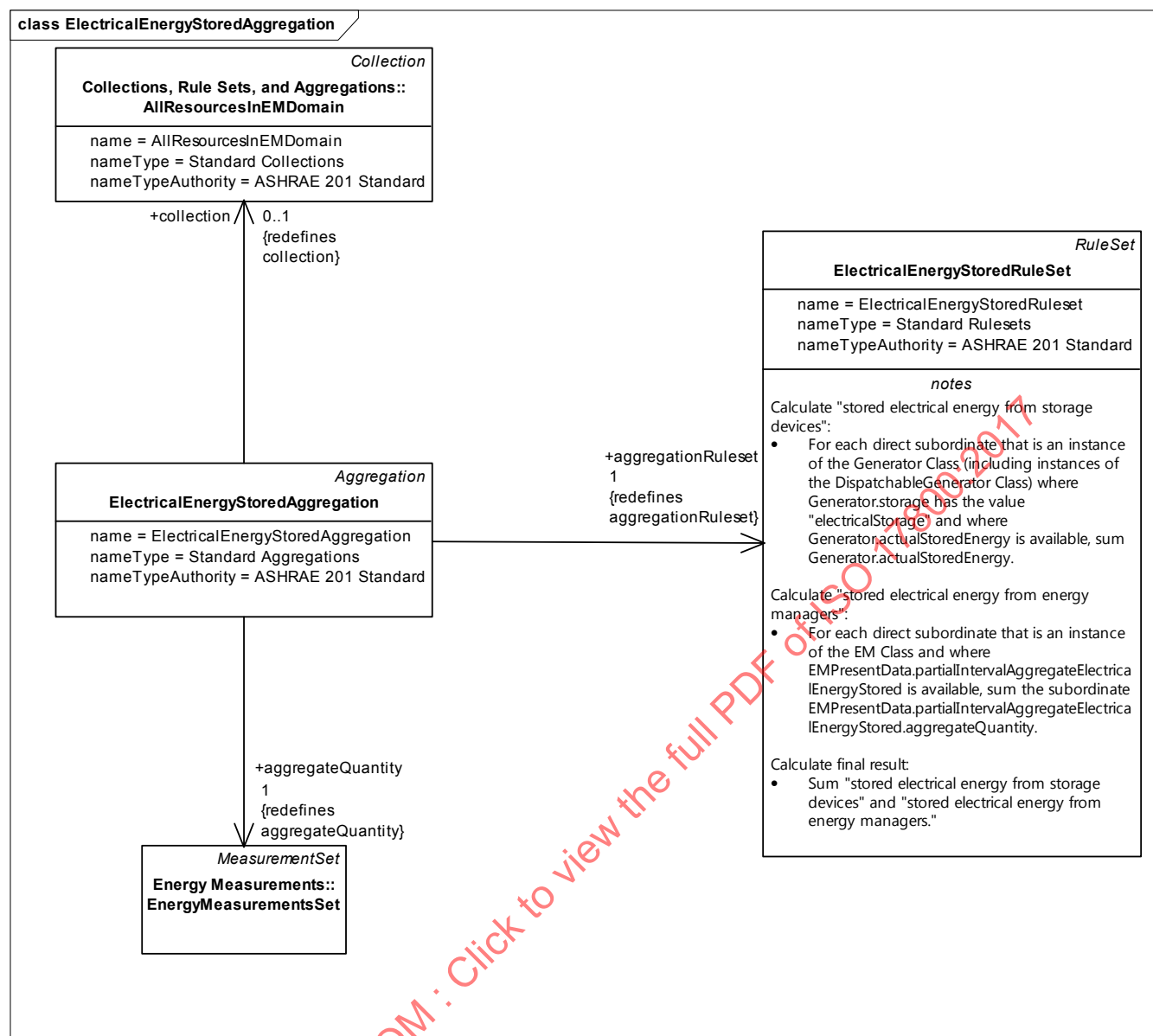


Figure 5.21 - ElectricalEnergyStoredAggregation

The ElectricalEnergyStoredAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

ElectricalEnergyStoredAggregation (Class)

For historical or forecast aggregations, this is the energy stored in electrical storage directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the energy stored in electrical storage directly or indirectly managed by the associated EM since the beginning of the interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ElectricalEnergyStoredAggregation*. See Clause 5.6.7.11.6.11 for the complete definition of this element.

ElectricalEnergyStoredRuleSet (Class)

Calculate "stored electrical energy from storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.storage has the value "electricalStorage" and where Generator.actualStoredEnergy is available, sum Generator.actualStoredEnergy.

Calculate "stored electrical energy from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.partialIntervalAggregateElectricalEnergyStored is available, sum the subordinate EMPresentData.partialIntervalAggregateElectricalEnergyStored.aggregateQuantity.

Calculate final result:

- Sum "stored electrical energy from storage devices" and "stored electrical energy from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ElectricalEnergyStoredRuleSet*. See Clause 5.6.7.11.6.12 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

5.6.3.3.7. EmissionsGeneratedAggregation Diagram

This diagram shows how the EmissionsGeneratedAggregation is derived by applying the EmissionsGeneratedRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as an EmissionsMeasurementsSet.

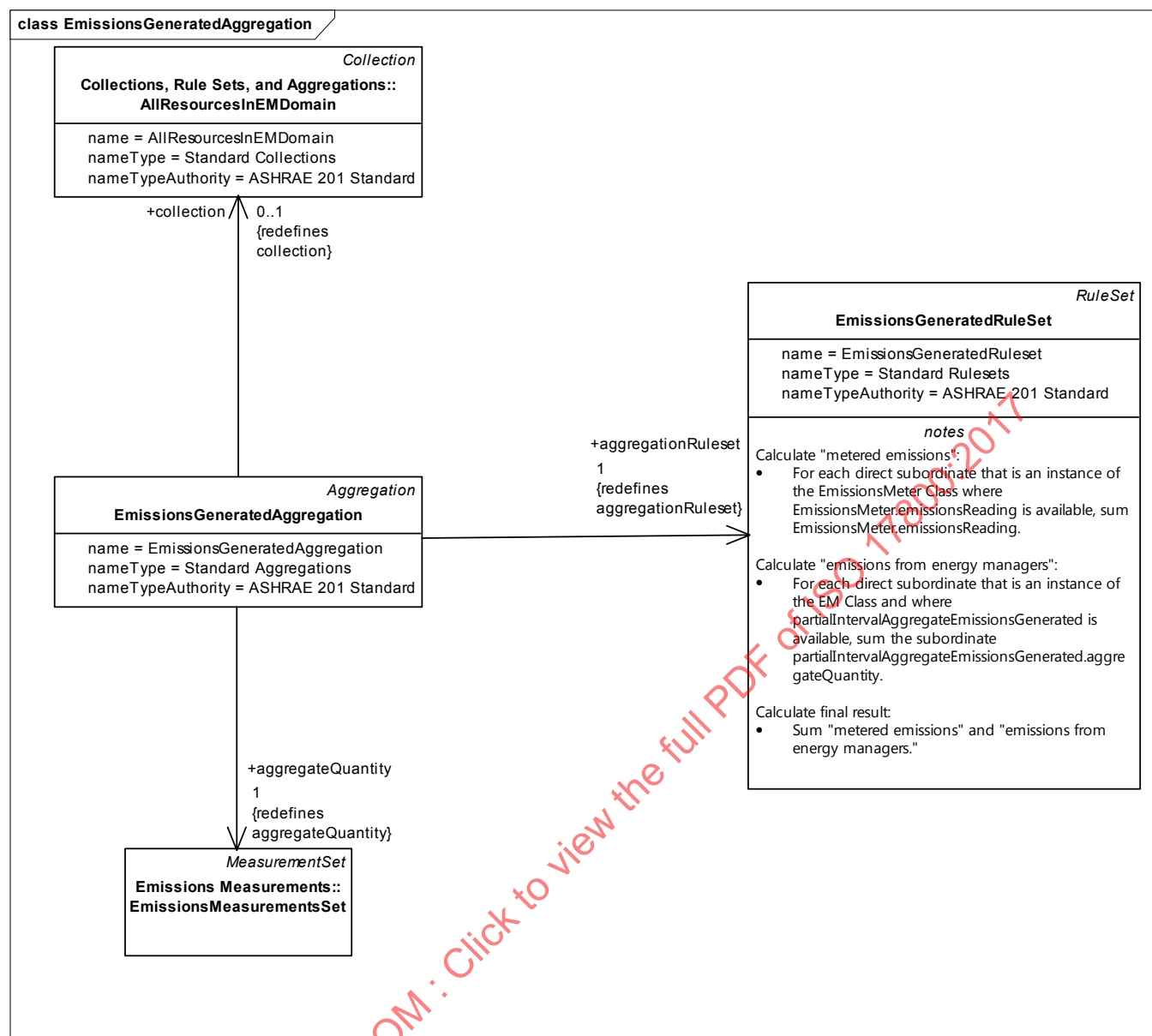


Figure 5.22 - EmissionsGeneratedAggregation

The EmissionsGeneratedAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

EmissionsGeneratedAggregation (Class)

For historical aggregations, this is the aggregate emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGeneratedAggregation*. See Clause 5.6.7.11.6.13 for the complete definition of this element.

EmissionsGeneratedRuleSet (Class)

Calculate "metered emissions":

- For each direct subordinate that is an instance of the EmissionsMeter Class where EmissionsMeter.emissionsReading is available, sum EmissionsMeter.emissionsReading.

Calculate "emissions from energy managers":

- For each direct subordinate that is an instance of the EM Class and where partialIntervalAggregateEmissionsGenerated is available, sum the subordinate partialIntervalAggregateEmissionsGenerated.aggregateQuantity.

Calculate final result:

- Sum "metered emissions" and "emissions from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGeneratedRuleSet*. See Clause 5.6.7.11.6.16 for the complete definition of this element.

EmissionsMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions quantity measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMeasurementsSet*. See Clause 6.6.4.8.12 for the complete definition of this element.

5.6.3.3.8. EmissionsGenerationRateAggregation Diagram

This diagram shows how the EmissionsGenerationRateAggregation is derived by applying the EmissionsGenerationRateRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as an EmissionsRateMeasurementsSet.

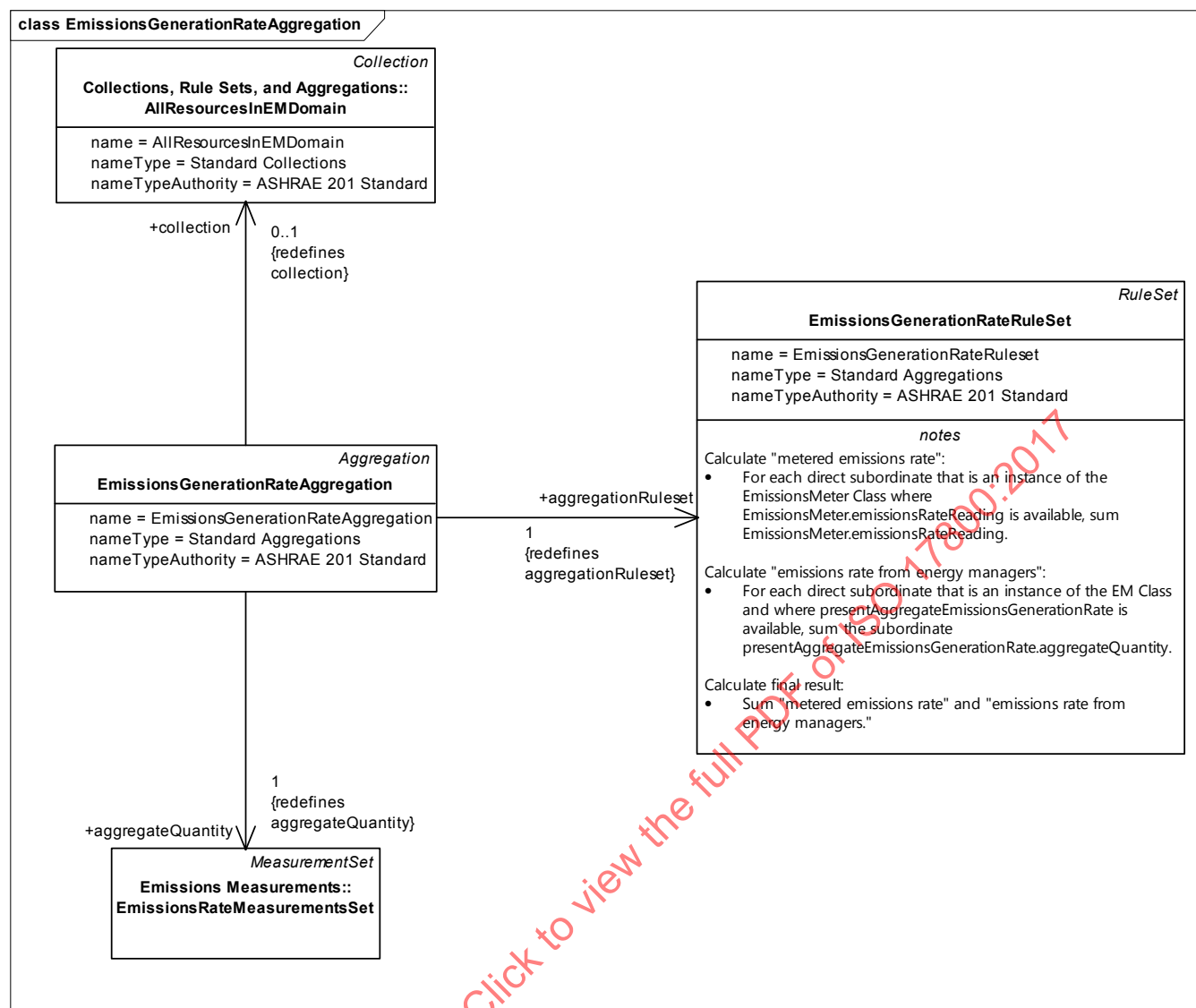


Figure 5.23 - EmissionsGenerationRateAggregation

The EmissionsGenerationRateAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

EmissionsGenerationRateAggregation (Class)

For historical aggregations, this is the aggregate rate of emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) emissions interval, this is the aggregate present subinterval rate of emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate rate of emissions generation of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the indentified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGenerationRateAggregation*. See Clause 5.6.7.11.6.14 for the complete definition of this element.

EmissionsGenerationRateRuleSet (Class)

Calculate "metered emissions rate":

- For each direct subordinate that is an instance of the EmissionsMeter Class where EmissionsMeter.emissionsRateReading is available, sum EmissionsMeter.emissionsRateReading.

Calculate "emissions rate from energy managers":

- For each direct subordinate that is an instance of the EM Class and where presentAggregateEmissionsGenerationRate is available, sum the subordinate presentAggregateEmissionsGenerationRate.aggregateQuantity.

Calculate final result:

- Sum "metered emissions rate" and "emissions rate from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGenerationRateRuleSet*. See Clause 5.6.7.11.6.15 for the complete definition of this element.

EmissionsRateMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions rate measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsRateMeasurementsSet*. See Clause 6.6.4.8.15 for the complete definition of this element.

5.6.3.3.9. EnergyConsumedAggregation Diagram

This diagram shows how the EnergyConsumedAggregation is derived by applying the EnergyConsumedRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as an EnergyMeasurementsSet.

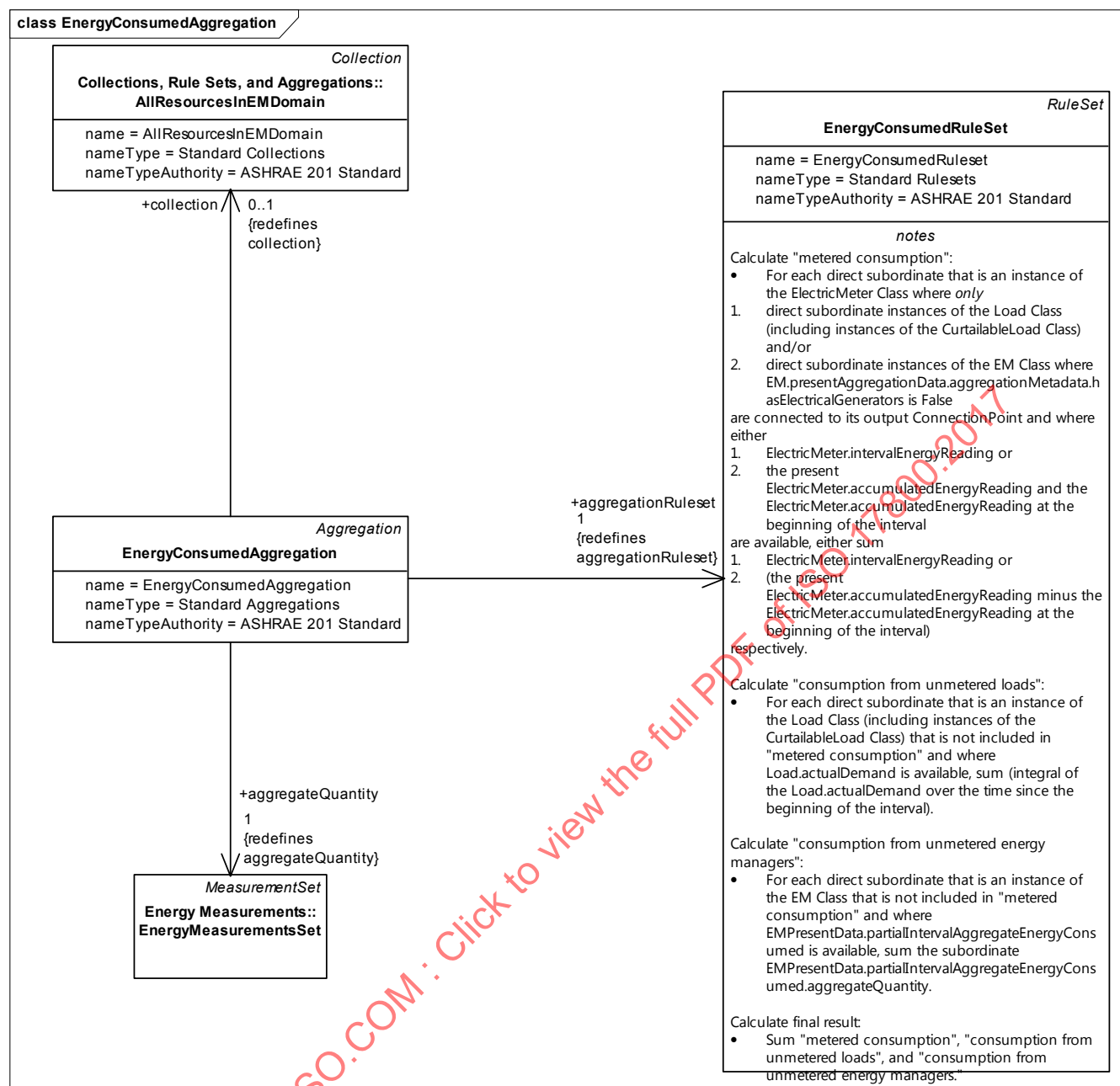


Figure 5.24 - EnergyConsumedAggregation

The EnergyConsumedAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

EnergyConsumedAggregation (Class)

For historical aggregations, this is the aggregate energy consumed by loads directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate energy consumed by loads directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate energy consumed by loads directly or indirectly managed by the associated EM during the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergyConsumedAggregation*. See Clause 5.6.7.11.6.17 for the complete definition of this element.

EnergyConsumedRuleSet (Class)

Calculate "metered consumption":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Load Class (including instances of the CurtailableLoad Class) and/or
 2. direct subordinate instances of the EM Class where
EM.presentAggregationData.aggregationMetadata.hasElectricalGenerators is False

are connected to its output ConnectionPoint and where either

1. ElectricMeter.intervalEnergyReading or
2. the present ElectricMeter.accumulatedEnergyReading and the ElectricMeter.accumulatedEnergyReading at the beginning of the interval

are available, either sum

1. ElectricMeter.intervalEnergyReading or
2. (the present ElectricMeter.accumulatedEnergyReading minus the ElectricMeter.accumulatedEnergyReading at the beginning of the interval)

respectively.

Calculate "consumption from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered consumption" and where Load.actualDemand is available, sum (integral of the Load.actualDemand over the time since the beginning of the interval).

Calculate "consumption from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered consumption" and where EMPresentData.partialIntervalAggregateEnergyConsumed is available, sum the subordinate EMPresentData.partialIntervalAggregateEnergyConsumed.aggregateQuantity.

Calculate final result:

- Sum "metered consumption", "consumption from unmetered loads", and "consumption from unmetered energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergyConsumedRuleSet*. See Clause 5.6.7.11.6.18 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

5.6.3.3.10. EnergySuppliedAggregation Diagram

This diagram shows how the EnergySuppliedAggregation is derived by applying the EnergySuppliedRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as an EnergyMeasurementsSet.

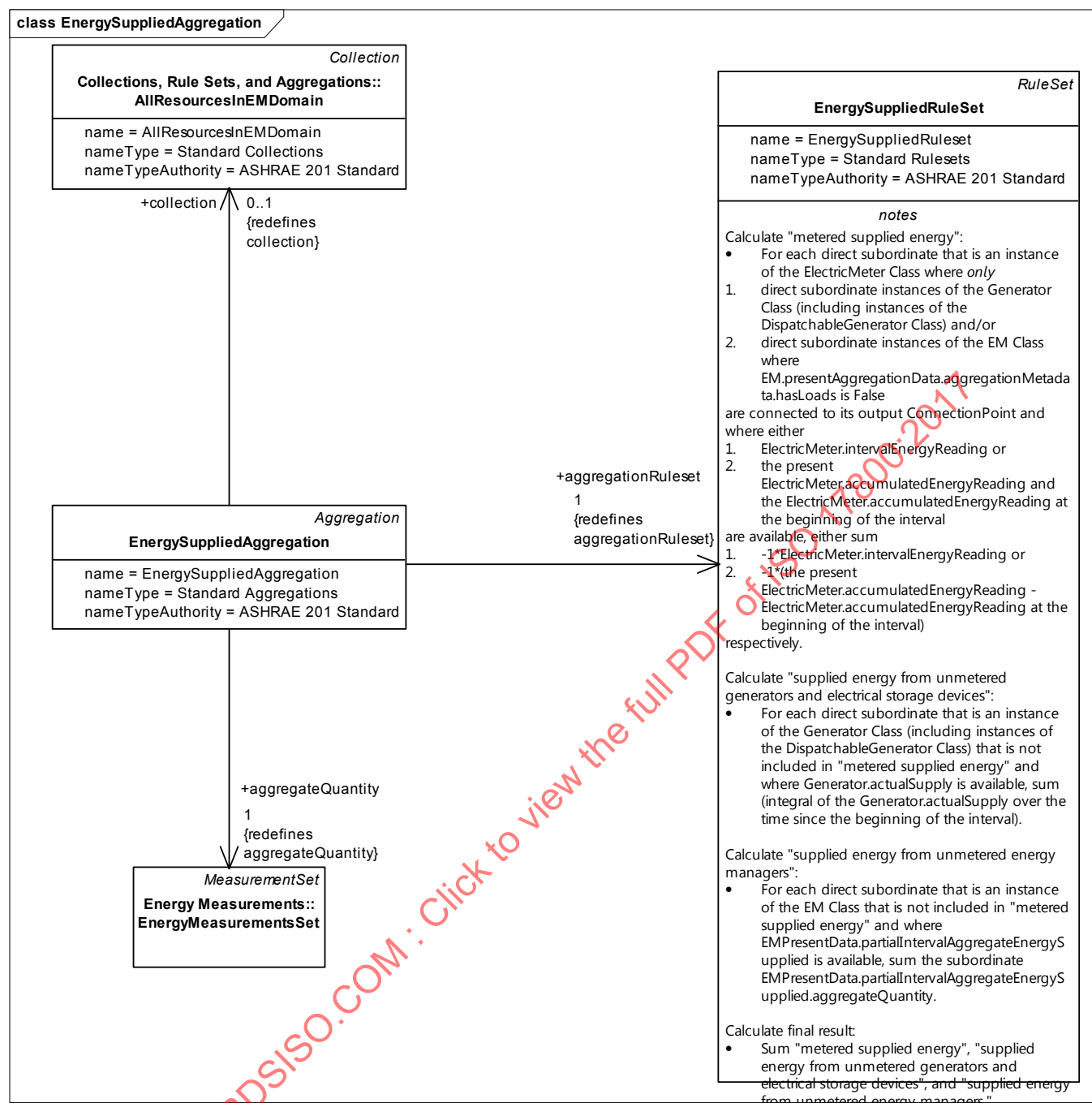


Figure 5.25 - EnergySuppliedAggregation

The EnergySuppliedAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

EnergySuppliedAggregation (Class)

For historical aggregations, this is the aggregate energy supplied by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate energy supplied by electrical generators and electrical storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate energy supplied by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergySuppliedAggregation*. See Clause 5.6.7.11.6.19 for the complete definition of this element.

EnergySuppliedRuleSet (Class)

Calculate "metered supplied energy":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Generator Class (including instances of the DispatchableGenerator Class) and/or
 2. direct subordinate instances of the EM Class where EM.presentAggregationData.aggregationMetadata.hasLoads is False

are connected to its output ConnectionPoint and where either

1. ElectricMeter.intervalEnergyReading or
2. the present ElectricMeter.accumulatedEnergyReading and the ElectricMeter.accumulatedEnergyReading at the beginning of the interval

are available, either sum

1. $-1 * \text{ElectricMeter.intervalEnergyReading}$ or
2. $-1 * (\text{the present ElectricMeter.accumulatedEnergyReading} - \text{ElectricMeter.accumulatedEnergyReading at the beginning of the interval})$

respectively.

Calculate "supplied energy from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered supplied energy" and where Generator.actualSupply is available, sum (integral of the Generator.actualSupply over the time since the beginning of the interval).

Calculate "supplied energy from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered supplied energy" and where EMPresentData.partialIntervalAggregateEnergySupplied is available, sum the subordinate EMPresentData.partialIntervalAggregateEnergySupplied.aggregateQuantity.

Calculate final result:

- Sum "metered supplied energy", "supplied energy from unmetered generators and electrical storage devices", and "supplied energy from unmetered energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergySuppliedRuleSet*. See Clause 5.6.7.11.6.20 for the complete definition of this element.

5.6.3.3.11. NetDemandAggregation Diagram

This diagram shows how the NetDemandAggregation is derived by applying the NetDemandRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as a PowerMeasurementsSet.

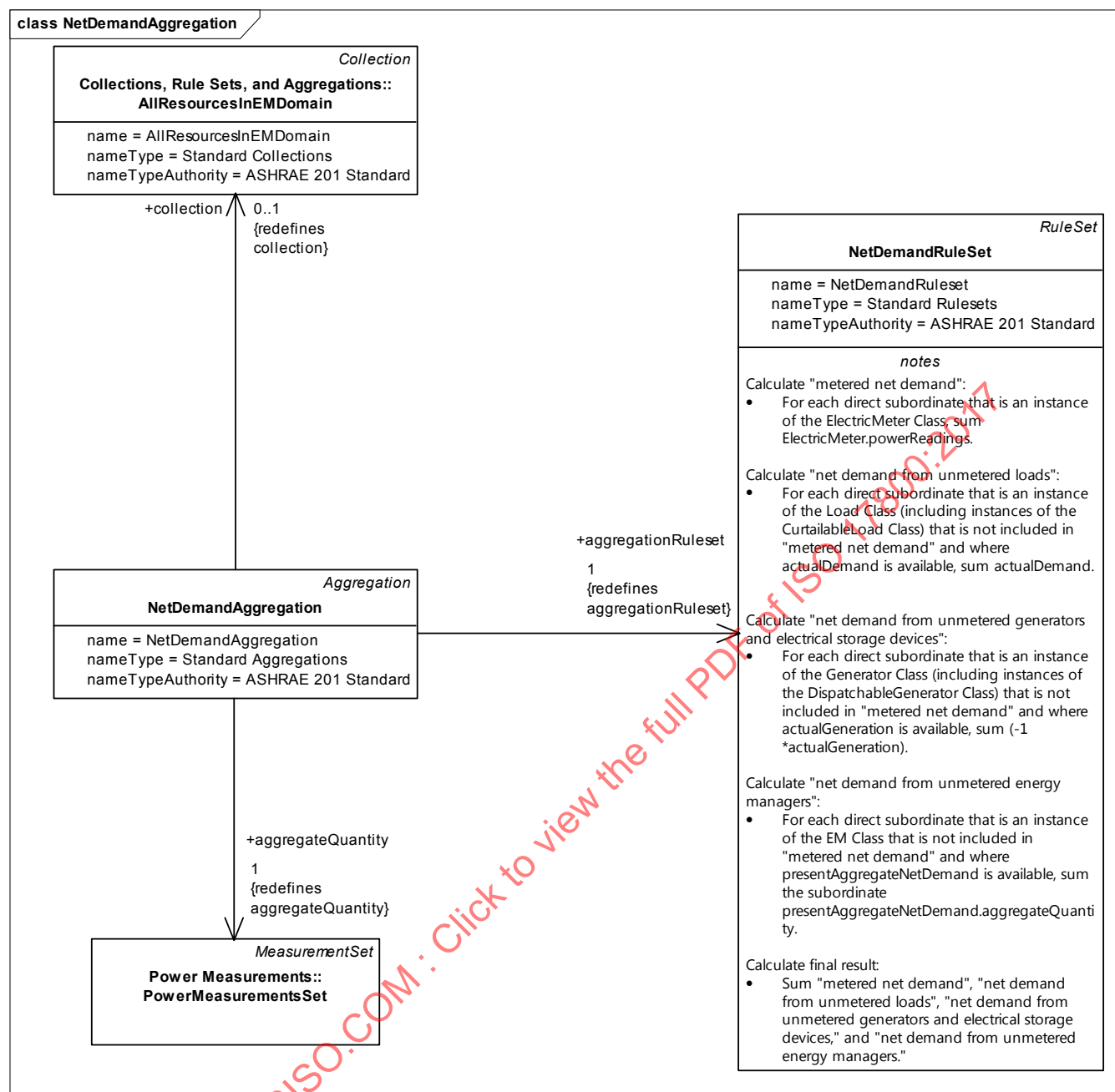


Figure 5.26 - NetDemandAggregation

The NetDemandAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

NetDemandAggregation (Class)

For historical aggregations, this is the aggregate net demand of loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate present subinterval net demand of loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate net demand of loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetDemandAggregation*. See Clause 5.6.7.11.6.21 for the complete definition of this element.

NetDemandRuleSet (Class)

Calculate "metered net demand":

- For each direct subordinate that is an instance of the ElectricMeter Class, sum ElectricMeter.powerReadings.

Calculate "net demand from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered net demand" and where actualDemand is available, sum actualDemand.

Calculate "net demand from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered net demand" and where actualGeneration is available, sum $(-1 * \text{actualGeneration})$.

Calculate "net demand from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered net demand" and where presentAggregateNetDemand is available, sum the subordinate presentAggregateNetDemand.aggregateQuantity.

Calculate final result:

- Sum "metered net demand", "net demand from unmetered loads", "net demand from unmetered generators and electrical storage devices," and "net demand from unmetered energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetDemandRuleSet*. See Clause 5.6.7.11.6.22 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

5.6.3.3.12. NetEnergyConsumedAggregation Diagram

This diagram shows how the NetEnergyConsumedAggregation is derived by applying the NetEnergyConsumedRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as an EnergyMeasurementsSet.

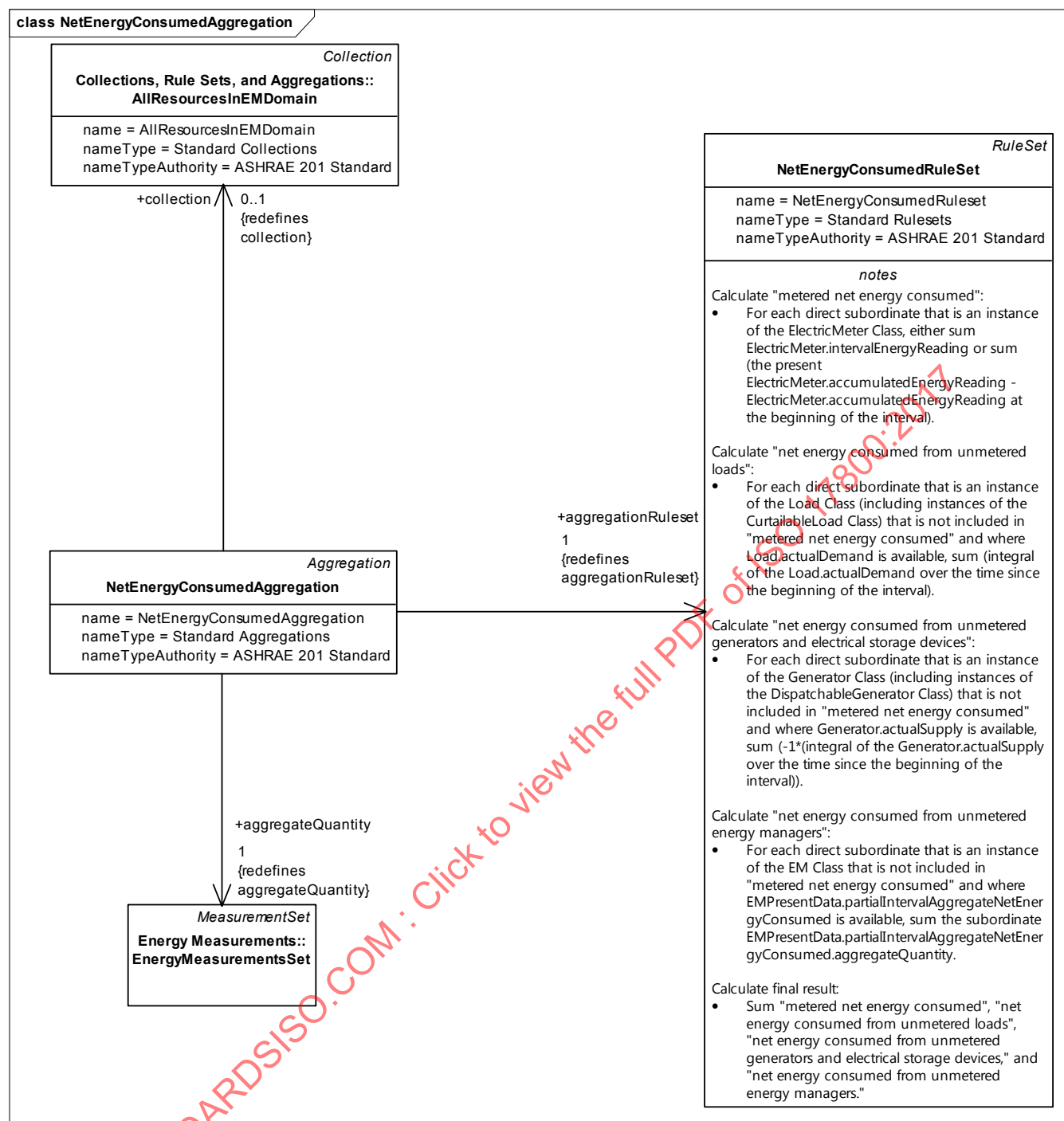


Figure 5.27 - NetEnergyConsumedAggregation

The NetEnergyConsumedAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

NetEnergyConsumedAggregation (Class)

For historical aggregations, this is the aggregate net energy consumed by loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate net energy consumed by loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate net energy consumed by loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetEnergyConsumedAggregation*. See Clause 5.6.7.11.6.23 for the complete definition of this element.

NetEnergyConsumedRuleSet (Class)

Calculate "metered net energy consumed":

- For each direct subordinate that is an instance of the ElectricMeter Class, either sum ElectricMeter.intervalEnergyReading or sum (the present ElectricMeter.accumulatedEnergyReading - ElectricMeter.accumulatedEnergyReading at the beginning of the interval).

Calculate "net energy consumed from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered net energy consumed" and where Load.actualDemand is available, sum (integral of the Load.actualDemand over the time since the beginning of the interval).

Calculate "net energy consumed from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered net energy consumed" and where Generator.actualSupply is available, sum $(-1 * (\text{integral of the Generator.actualSupply over the time since the beginning of the interval}))$.

Calculate "net energy consumed from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered net energy consumed" and where EMPresentData.partialIntervalAggregateNetEnergyConsumed is available, sum the subordinate EMPresentData.partialIntervalAggregateNetEnergyConsumed.aggregateQuantity.

Calculate final result:

- Sum "metered net energy consumed", "net energy consumed from unmetered loads", "net energy consumed from unmetered generators and electrical storage devices," and "net energy consumed from unmetered energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetEnergyConsumedRuleSet*. See Clause 5.6.7.11.6.24 for the complete definition of this element.

5.6.3.3.13. SupplyAggregation Diagram

This diagram shows how the SupplyAggregation is derived by applying the SupplyRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as a PowerMeasurementsSet.

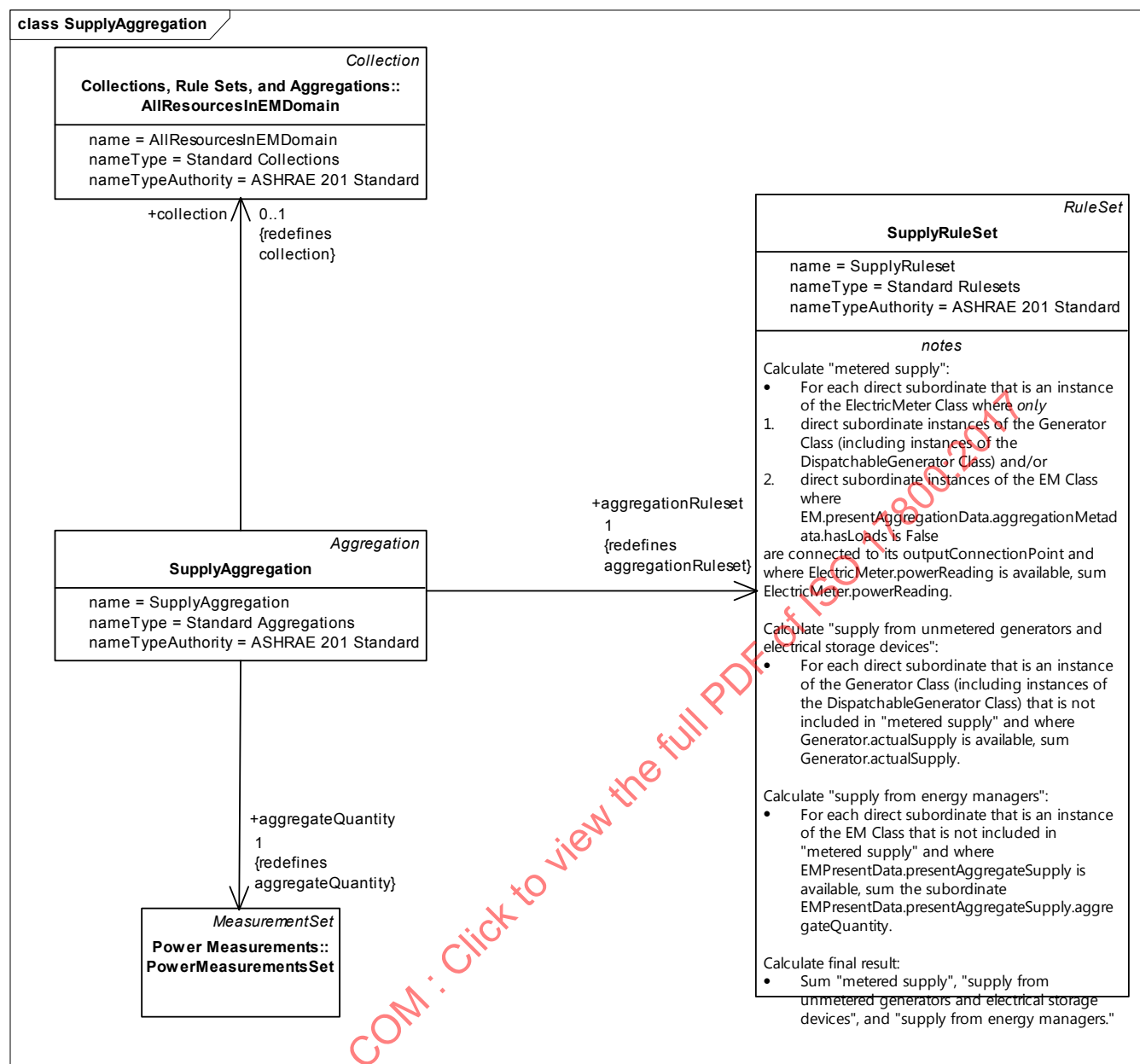


Figure 5.28 - SupplyAggregation

The SupplyAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

SupplyAggregation (Class)

For historical aggregations, this is the aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate present subinterval supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\SupplyAggregation*. See Clause 5.6.7.11.6.25 for the complete definition of this element.

SupplyRuleSet (Class)

Calculate "metered supply":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Generator Class (including instances of the DispatchableGenerator Class) and/or
 2. direct subordinate instances of the EM Class where EM.presentAggregationData.aggregationMetadata.hasLoads is False

are connected to its outputConnectionPoint and where ElectricMeter.powerReading is available, sum ElectricMeter.powerReading.

Calculate "supply from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered supply" and where Generator.actualSupply is available, sum Generator.actualSupply.

Calculate "supply from energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered supply" and where EMPresentData.presentAggregateSupply is available, sum the subordinate EMPresentData.presentAggregateSupply.aggregateQuantity.

Calculate final result:

- Sum "metered supply", "supply from unmetered generators and electrical storage devices", and "supply from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\SupplyRuleSet*. See Clause 5.6.7.11.6.26 for the complete definition of this element.

5.6.3.3.14. ThermalEnergyStoredAggregation Diagram

This diagram shows how the ThermalEnergyStoredAggregation is derived by applying the ThermalEnergyStoredRuleset to the AllResourcesInEMDomain collection to obtain an aggregateQuantity that is represented as an EnergyThermalQuantity.

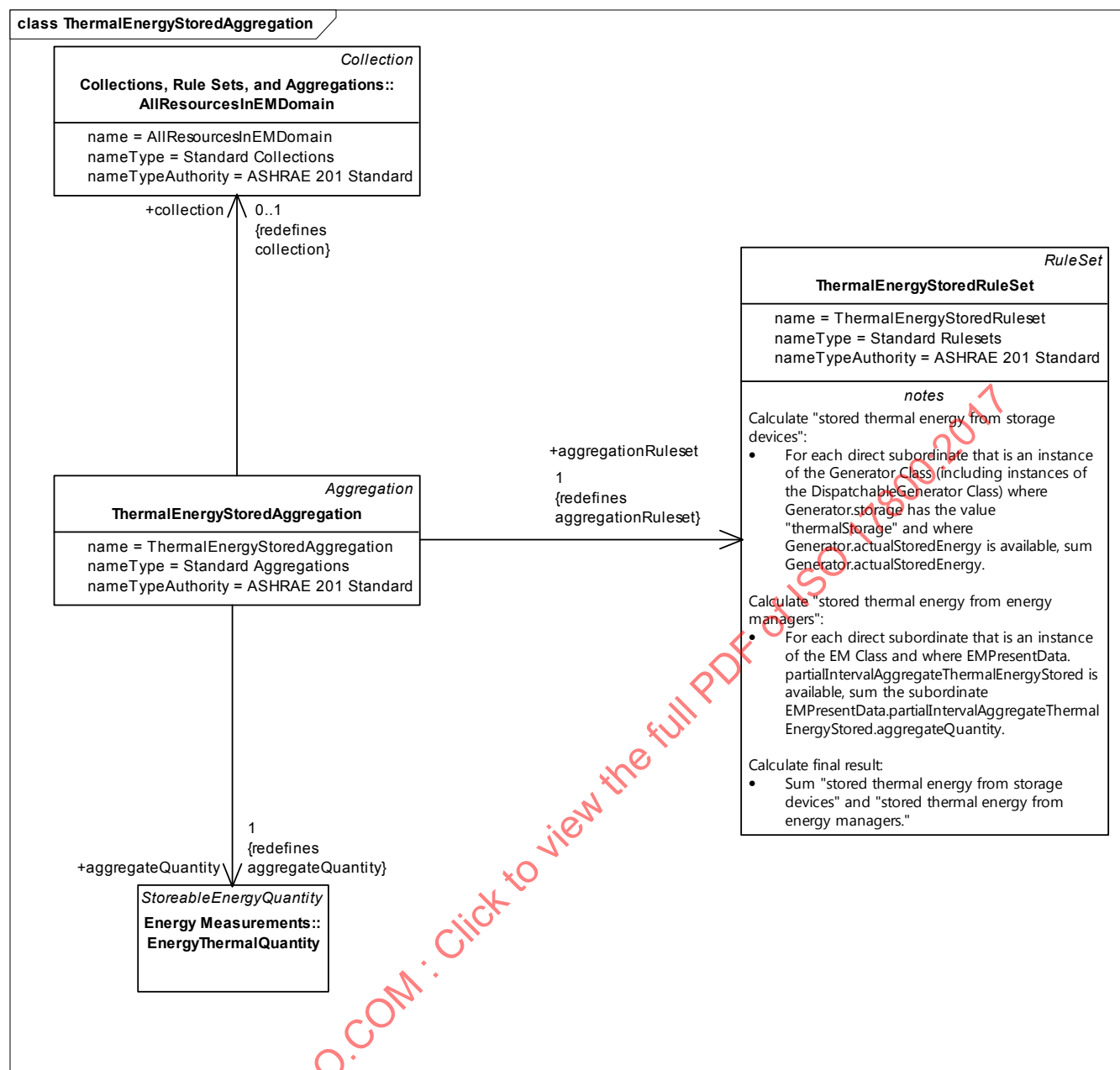


Figure 5.29 - ThermalEnergyStoredAggregation

The ThermalEnergyStoredAggregation Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

EnergyThermalQuantity (Class)

Quantity/metadata pair used to represent thermal energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyThermalQuantity*. See Clause 6.6.4.7.15 for the complete definition of this element.

ThermalEnergyStoredAggregation (Class)

For historical aggregations, this is the aggregate thermal energy stored by thermal storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate thermal energy stored by thermal storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate thermal energy stored by thermal storage devices directly or indirectly managed by the associated EM during the identified time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ThermalEnergyStoredAggregation*. See Clause 5.6.7.11.6.27 for the complete definition of this element.

ThermalEnergyStoredRuleSet (Class)

Calculate "stored thermal energy from storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.storage has the value "thermalStorage" and where Generator.actualStoredEnergy is available, sum Generator.actualStoredEnergy.

Calculate "stored thermal energy from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.partialIntervalAggregateThermalEnergyStored is available, sum the subordinate EMPresentData.partialIntervalAggregateThermalEnergyStored.aggregateQuantity.

Calculate final result:

- Sum "stored thermal energy from storage devices" and "stored thermal energy from energy managers."

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ThermalEnergyStoredRuleSet*. See Clause 5.6.7.11.6.28 for the complete definition of this element.

5.6.4. FSGIMWeather

The FSGIMWeather model describes an abstract representation of weather information that may be used by or exchanged with the facility. It includes weather information and forecasts that may be received from outside providers, and information generated through observations (measurements) performed at the facility. The weather model also includes weather phenomena such as wind, rain, and storms.

UML Package Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather*

5.6.4.1. FSGIMWeatherEnumerations Diagram

Many attributes of the FSGIMWeather model are represented as enumerated values. This diagram provides a summary of the enumerations of the model.

The FSGIMWeatherEnumerations Diagram makes use of the following elements:

Air temperature.

138

CI_Address (Class)

Location of the responsible individual or organisation.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Address*. See Clause 5.7.6.6.4.6.2 for the complete definition of this element.

CI_Citation (Class)

Standardized resource reference.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Citation*. See Clause 5.7.6.6.4.6.3 for the complete definition of this element.

CI_Contact (Class)

Information required enabling contact with the responsible person and/or organisation

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Contact*. See Clause 5.7.6.6.4.6.4 for the complete definition of this element.

CI_OnLineFunctionCode (Enumeration)

Function performed by the resource

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_OnLineFunctionCode*. See Clause 5.7.6.6.4.6.9 for the complete definition of this element.

CI_OnlineResource (Class)

Information about online sources from which the dataset, specification, or community profile name and extended metadata elements can be obtained.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_OnlineResource*. See Clause 5.7.6.6.4.6.5 for the complete definition of this element.

CI_PresentationFormCode (Enumeration)

Mode in which the data is represented

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_PresentationFormCode*. See Clause 5.7.6.6.4.6.10 for the complete definition of this element.

CI_ResponsibleParty (Class)

Identification of, and means of communication with, person(s) and organisations associated with the dataset.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_ResponsibleParty*. See Clause 5.7.6.6.4.6.6 for the complete definition of this element.

CI_RoleCode (Enumeration)

Function performed by the responsible party.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_RoleCode*. See Clause 5.7.6.6.4.6.11 for the complete definition of this element.

CI_Telephone (Class)

Telephone numbers for contacting the responsible individual or organisation.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Telephone*. See Clause 5.7.6.6.4.6.8 for the complete definition of this element.

DQ_Element (Abstract Class)

Used to provide user specific quality information about a dataset's suitability for a particular application.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Data quality information\DQ_Element*. See Clause 5.7.6.6.4.6.13 for the complete definition of this element.

DQ_EvaluationMethodTypeCode (Enumeration)

Evaluation method type code.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Data quality information\DQ_EvaluationMethodTypeCode*. See Clause 5.7.6.6.4.6.15 for the complete definition of this element.

GM_Envelope (Class)

GM_Envelope is often referred to as a minimum bounding box or rectangle. Regardless of dimension, a GM_Envelope can be represented without ambiguity as two direct positions (coordinate points). To encode a GM_Envelope, it is sufficient to encode these two points. This is consistent with all of the data types in this standard, their state is represented by their publicly accessible attributes.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19107 Spatial Schema\Geometry\Coordinate geometry\GM_Envelope*. See Clause 5.7.6.6.4.4.4 for the complete definition of this element.

MD_CharacterSetCode (Enumeration)

Character set code list.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Identification information\MD_CharacterSetCode*. See Clause 5.7.6.6.4.6.19 for the complete definition of this element.

MD_Identifier (Class)

Generic identifier class used to associate a reference code with an identification authority.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Reference system information\MD_Identifier*. See Clause 5.7.6.6.4.6.25 for the complete definition of this element.

MD_Metadata (Class)

Information about the metadata.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Metadata entity set information\MD_Metadata*. See Clause 5.7.6.6.4.6.23 for the complete definition of this element.

MD_ScopeCode (Enumeration)

Class of information to which the referencing entity applies.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Maintenance information\MD_ScopeCode*. See Clause 5.7.6.6.4.6.21 for the complete definition of this element.

Percentage (Class)

Generic percentage class; value is limited to between 0 and 100 inclusive.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Percentage*. See Clause 5.7.6.6.2.1.6 for the complete definition of this element.

PercentageRange (Class)

Percentage range.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\PercentageRange*. See Clause 5.7.6.6.2.5.1 for the complete definition of this element.

Pressure (Class)

Generic pressure class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Pressure*. See Clause 5.7.6.6.2.1.7 for the complete definition of this element.

PT_Locale (Class)

A class describing the cultural locale for a location including language, country, and character set.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19139 Metadata\Cultural and linguistic adaptability\PT_Locale*. See Clause 5.7.6.6.4.2.2 for the complete definition of this element.

RelativeHumidity (Class)

The relative humidity, the ratio of the partial pressure of water vapor to the equilibrium vapor pressure at constant temperature.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\RelativeHumidity*. See Clause 5.6.7.12.7 for the complete definition of this element.

ReportProcess (Class)

The process related to the distribution or collection of weather reports

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Process\ReportProcess*. See Clause 5.7.6.6.2.12.1 for the complete definition of this element.

5.6.4.2. InheritanceFromWXXM Diagram

The FSGIMInheritance diagram illustrates how the FSGIMWeather class is derived from existing WXXM classes and adds its own specialization for FSGIM.

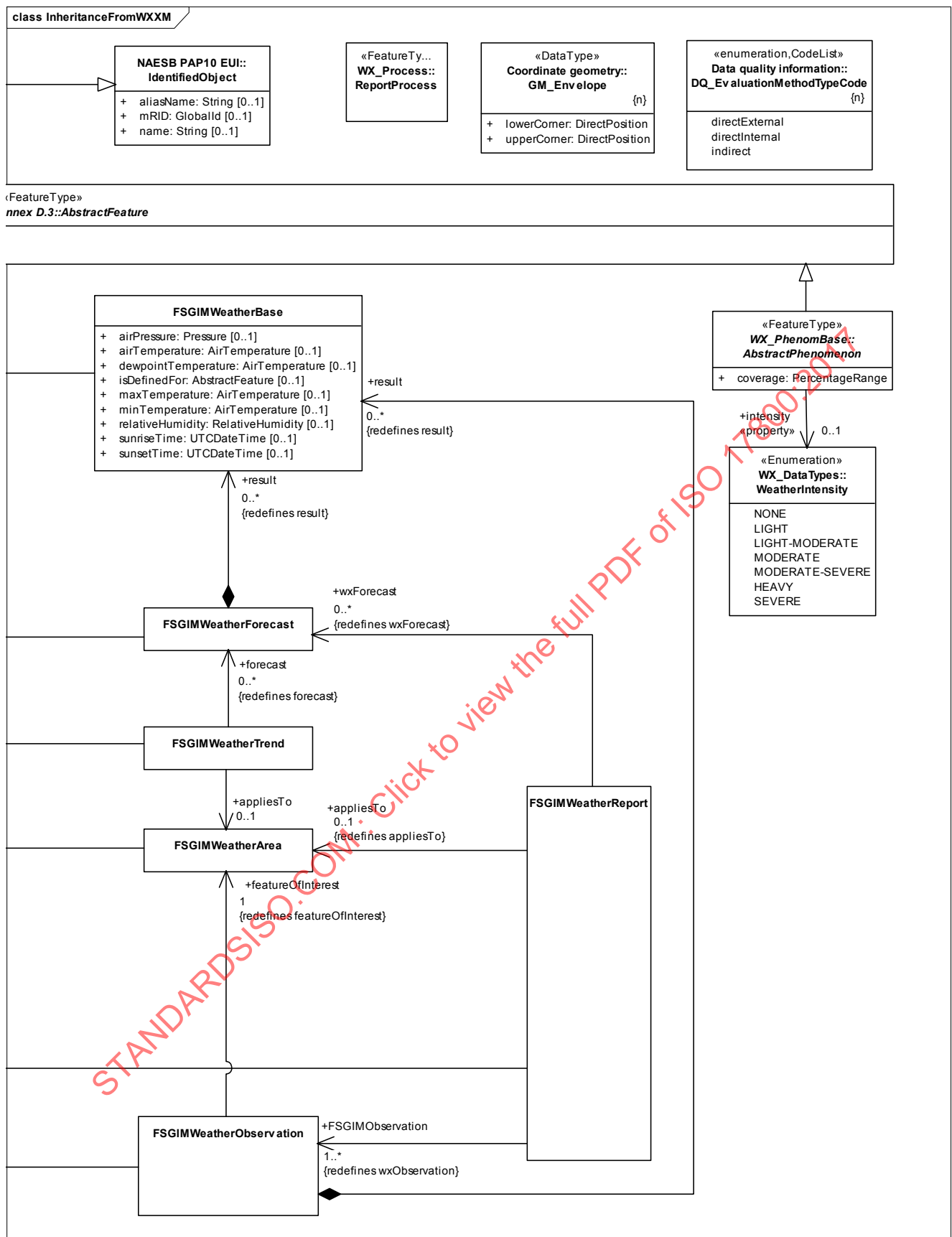


Figure 5.31 - InheritanceFromWXXM

The InheritanceFromWXXM Diagram makes use of the following elements:

AbstractFeature (Abstract Class)

Abstract class representing all classes stereotyped <<FeatureType>> i.e., classes that realize GF_FeatureType, representing features of all types from all application schemas. This class may be used as a "wildcard" when it is necessary to refer to a feature of any type.

N.B. Any class stereotyped <<FeatureType>> is implicitly substitutable for AnyFeature.

Implemented by gml:AbstractFeature (gml 3.2) or gml:_Feature (earlier versions).

UML Element Location: *Model\Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractFeature*. See Clause 5.7.6.6.3.1.1 for the complete definition of this element.

AbstractGML (Abstract Class)

Abstract class representing all classes stereotyped <<ObjectType>> i.e., classes that represent objects with identity that may be externally referenced. Implemented by gml:AbstractGML (gml 3.2) or gml:_GML (earlier versions).

UML Element Location: *Model\Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractGML*. See Clause 5.7.6.6.3.1.2 for the complete definition of this element.

AbstractObject (Class)

Abstract object class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractObject*. See Clause 5.7.6.6.3.1.3 for the complete definition of this element.

AbstractPhenomenon (Abstract Class)

Abstract class intended to act as head of substitution group for all weather phenomena classes.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\AbstractPhenomenon*. See Clause 5.7.6.6.2.11.1 for the complete definition of this element.

AbstractReport (Abstract Class)

Abstract report base class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\ICAO_Annex_3_Base\AbstractReport*. See Clause 5.7.6.6.1.9.1 for the complete definition of this element.

AbstractWxFeature (Abstract Class)

Abstract weather feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Base\AbstractWxFeature*. See Clause 5.7.6.6.2.2.1 for the complete definition of this element.

Aerodrome (Class)

Stand-in class for AIXM equivalent (Airport/Heliport).

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Features\Aerodrome*. See Clause 5.7.6.6.1.3.1 for the complete definition of this element.

DirectPosition (Class)

DirectPosition object data types (Figure 14) hold the coordinates for a position within some coordinate reference system. The coordinate reference system is described in ISO 19111. Since DirectPositions, as data types, will often be included in larger objects (such as GM_Objects) that have references to ISO19111::SC_CRS, the DirectPosition::coordinateReferenceSystem may be left NULL if this particular DirectPosition is included in a larger object with such a reference to a SC_CRS. In this case, the DirectPosition::coordinateReferenceSystem is implicitly assumed to take on the value of the containing object's SC_CRS.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19107 Spatial Schema\Geometry\Coordinate geometry\DirectPosition*. See Clause 5.7.6.6.4.4.3 for the complete definition of this element.

DQ_EvaluationMethodTypeCode (Enumeration)

Evaluation method type code.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Data quality information\DQ_EvaluationMethodTypeCode*. See Clause 5.7.6.6.4.6.15 for the complete definition of this element.

Forecast (Class)

Generic weather forecast class. Forecast inherits from O&M observation. This is a temporary fix as this inheritance violates Liskov substitution principle (subclass can't be used as a substitute for the super class).

For a Forecast the samplingTime implies the time period associated with the observations used to produce the forecast.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Forecast\Forecast*. See Clause 5.7.6.6.2.7.1 for the complete definition of this element.

ForecastChangeIndicator (Enumeration)

Enumeration for forecast change indicator descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\ForecastChangeIndicator*. See Clause 5.7.6.6.2.5.7 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

FSGIMWeatherArea (Class)

Defines the area of interest for a weather observation or forecast.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherArea*. See Clause 5.6.7.12.1 for the complete definition of this element.

FSGIMWeatherBase (Class)

This class defines the basic weather information including temperature and humidity data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherBase*. See Clause 5.6.7.12.8 for the complete definition of this element.

FSGIMWeatherForecast (Class)

The weather forecast of observations at a future time and with a defined confidence and probability.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherForecast*. See Clause 5.6.7.12.2 for the complete definition of this element.

FSGIMWeatherObservation (Class)

Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherObservation*. See Clause 5.6.7.12.3 for the complete definition of this element.

FSGIMWeatherReport (Class)

A container class for weather observations, forecasts, and trends.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherReport*. See Clause 5.6.7.12.4 for the complete definition of this element.

FSGIMWeatherTrend (Class)

A sequence of forecasts of weather in increments of a defined duration such as hourly.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherTrend*. See Clause 5.6.7.12.5 for the complete definition of this element.

GM_Envelope (Class)

GM_Envelope is often referred to as a minimum bounding box or rectangle. Regardless of dimension, a GM_Envelope can be represented without ambiguity as two direct positions (coordinate points). To encode a GM_Envelope, it is sufficient to encode these two points. This is consistent with all of the data types in this standard, their state is represented by their publicly accessible attributes.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19107 Spatial Schema\Geometry\Coordinate geometry\GM_Envelope*. See Clause 5.7.6.6.4.4.4 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject*. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Observation (Class)

Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

UML Element Location: *Model\Model Elements from External Sources\WXXM\SWE\OMI_Observation\observation\Observation*. See Clause 5.7.6.6.5.1.2 for the complete definition of this element.

Observation (Class)

Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Observation\Observation*. See Clause 5.7.6.6.2.10.1 for the complete definition of this element.

ReportProcess (Class)

The process related to the distribution or collection of weather reports

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Process\ReportProcess*. See Clause 5.7.6.6.2.12.1 for the complete definition of this element.

TREND (Class)

TREND forecast.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\ICAO_Annex_3_Base\TREND*. See Clause 5.7.6.6.1.9.2 for the complete definition of this element.

WeatherIntensity (Enumeration)

Enumeration for weather intensity descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\WeatherIntensity*. See Clause 5.7.6.6.2.5.12 for the complete definition of this element.

5.6.4.3. WXXMForFSGIM Diagram

This diagram illustrates the main classes of the FSGIMWeather model. Shown are the principal building blocks of:

FSGIMWeatherBase -- a class that contains measurements and related phenomena comprising a state of weather information.

FSGIMPrecipitation extends the base to include precipitation data.
 FSGIMSolar extends the base to include solar data.
 FSGIMWind extends the base to include wind data.
 FSGIMWeatherArea -- the geographic area of interest.
 FSGIMObservation -- a set of FSGIMWeather objects.
 FSGIMForecast -- a set of forecasted FSGIMWeather objects.
 FSGIMTrend -- a sequence of FSGIMForecast objects.

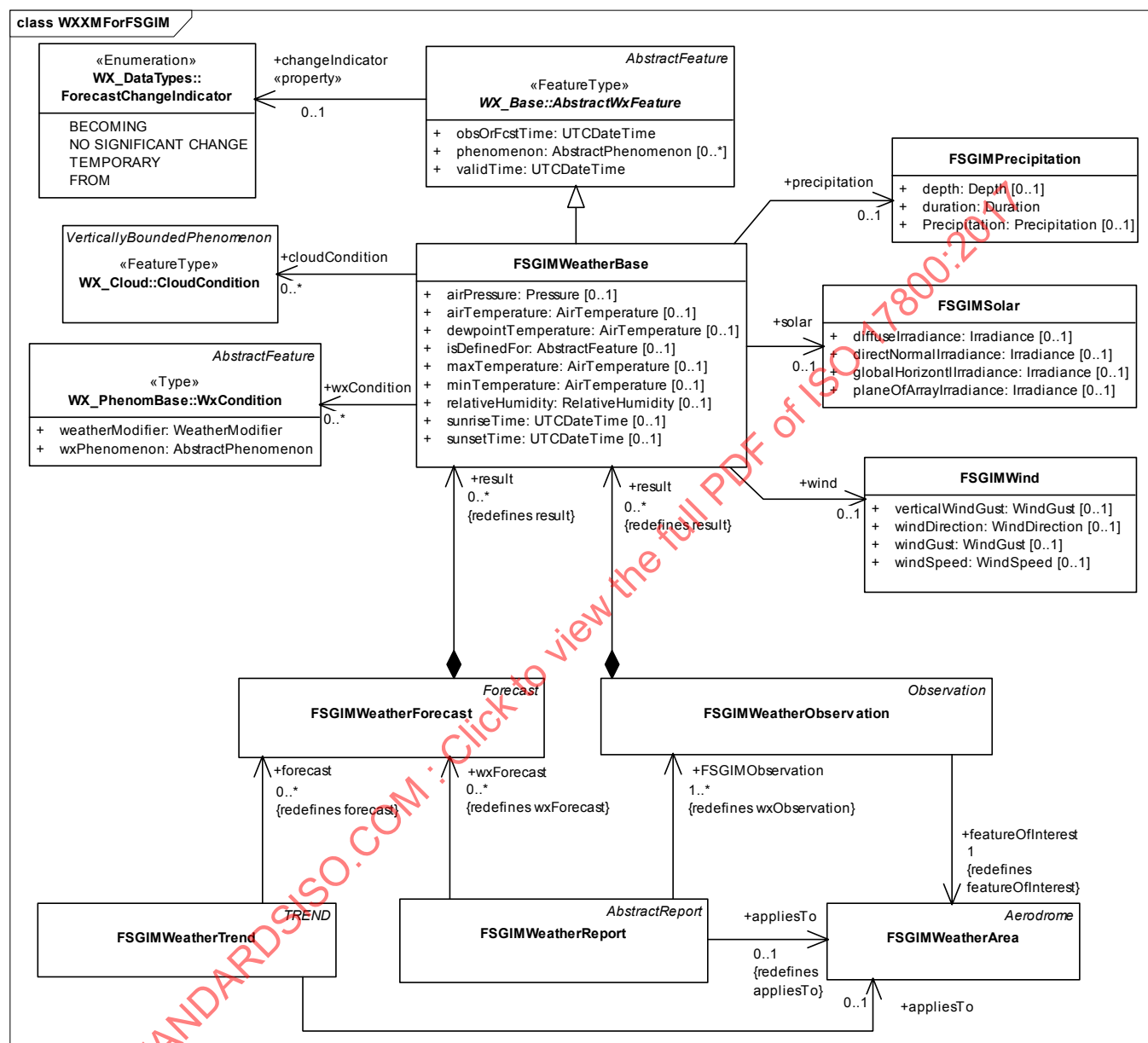


Figure 5.32- WXXMForFSGIM

The WXXMForFSGIM Diagram makes use of the following elements:

AbstractWxFeature (Abstract Class)

Abstract weather feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Base\AbstractWxFeature*. See Clause 5.7.6.2.2.1 for the complete definition of this element.

CloudCondition (Class)

Cloud condition.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\CloudCondition*. See Clause 5.7.6.2.3.1 for the complete definition of this element.

ForecastChangeIndicator (Enumeration)

Enumeration for forecast change indicator descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\ForecastChangeIndicator*. See Clause 5.7.6.2.5.7 for the complete definition of this element.

FSGIMPrecipitation (Class)

This class extends the basic weather data with precipitation data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMPrecipitation*. See Clause 5.6.7.12.9 for the complete definition of this element.

FSGIMSolar (Class)

This class extends the basic weather data with solar data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMSolar*. See Clause 5.6.7.12.10 for the complete definition of this element.

FSGIMWeatherArea (Class)

Defines the area of interest for a weather observation or forecast.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherArea*. See Clause 5.6.7.12.1 for the complete definition of this element.

FSGIMWeatherBase (Class)

This class defines the basic weather information including temperature and humidity data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherBase*. See Clause 5.6.7.12.8 for the complete definition of this element.

FSGIMWeatherForecast (Class)

The weather forecast of observations at a future time and with a defined confidence and probability.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherForecast*. See Clause 5.6.7.12.2 for the complete definition of this element.

FSGIMWeatherObservation (Class)

Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherObservation*. See Clause 5.6.7.12.3 for the complete definition of this element.

FSGIMWeatherReport (Class)

A container class for weather observations, forecasts, and trends.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherReport*. See Clause 5.6.7.12.4 for the complete definition of this element.

FSGIMWeatherTrend (Class)

A sequence of forecasts of weather in increments of a defined duration such as hourly.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherTrend*. See Clause 5.6.7.12.5 for the complete definition of this element.

FSGIMWind (Class)

This class extends the basic weather data with wind data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWind*. See Clause 5.6.7.12.11 for the complete definition of this element.

WxCondition (Class)

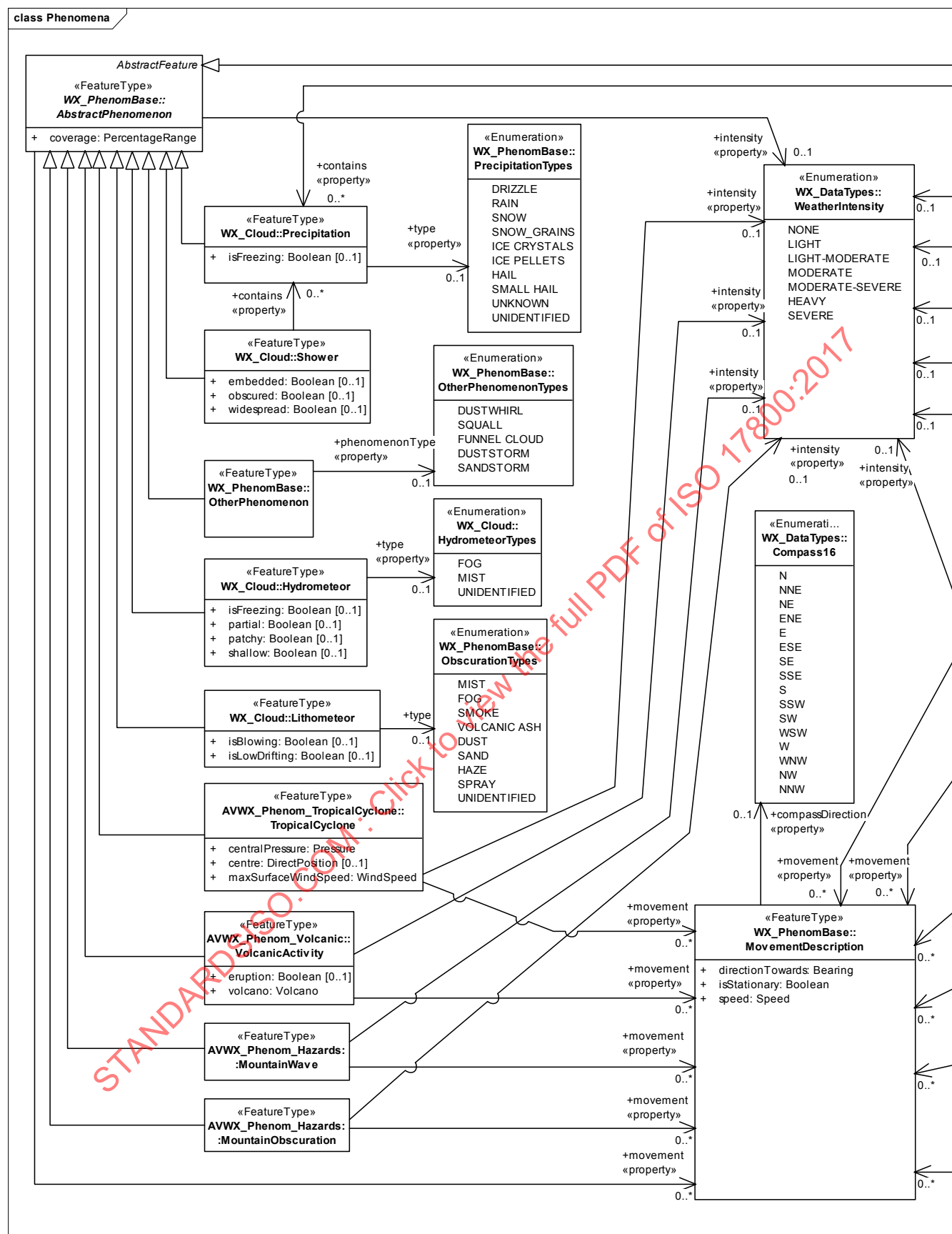
Weather condition description.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\WxCondition*. See Clause 5.7.6.6.2.11.5 for the complete definition of this element.

5.6.4.4. Phenomena Diagram

Phenomena are transient weather related events such as storms, tornados, storms, etc...

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



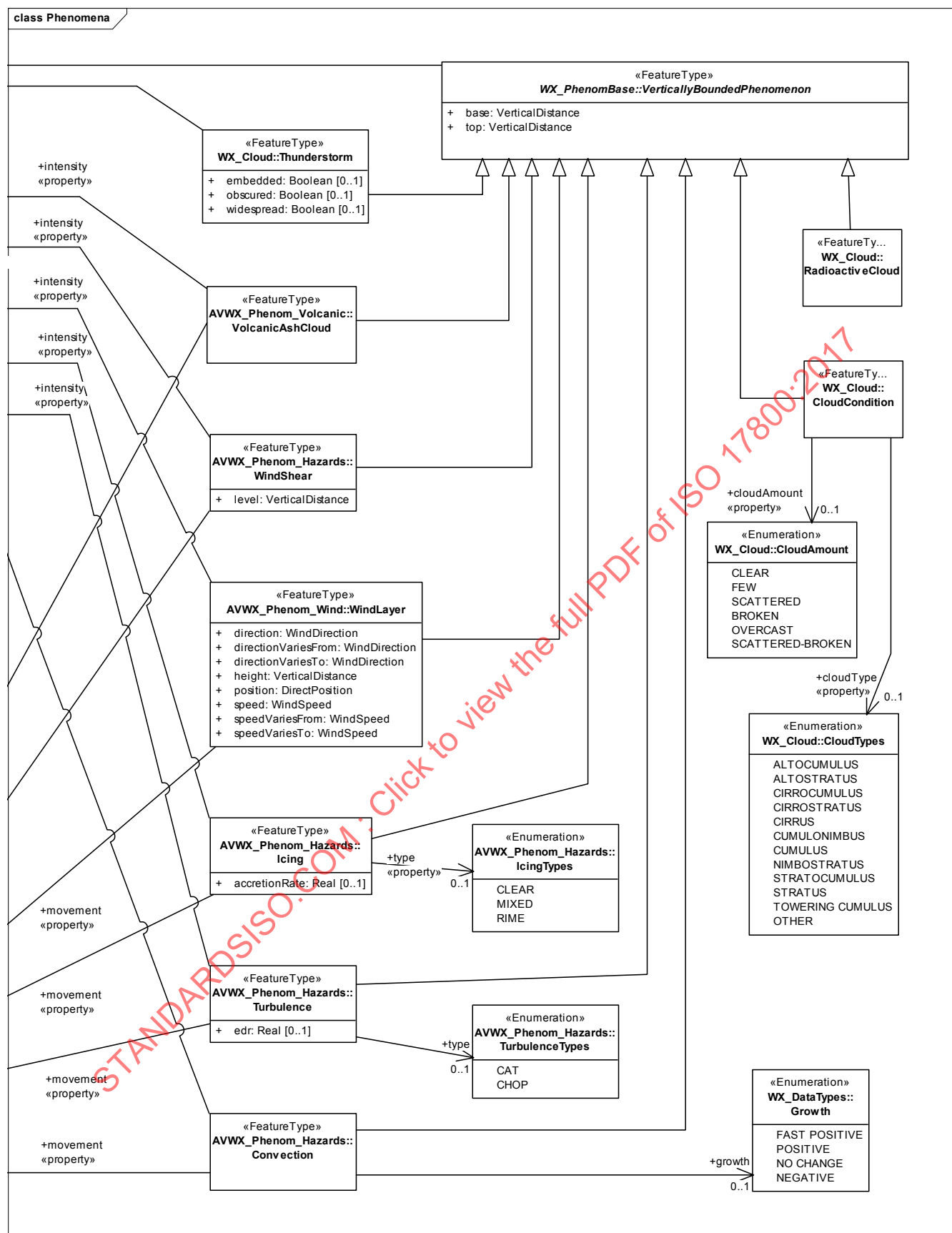


Figure 5.33 - Phenomena

The Phenomena Diagram makes use of the following elements:

AbstractPhenomenon (Abstract Class)

Abstract class intended to act as head of substitution group for all weather phenomena classes.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\AbstractPhenomenon*. See Clause 5.7.6.6.2.11.1 for the complete definition of this element.

CloudAmount (Enumeration)

Enumeration for cloud amounts.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\CloudAmount*. See Clause 5.7.6.6.2.3.8 for the complete definition of this element.

CloudCondition (Class)

Cloud condition.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\CloudCondition*. See Clause 5.7.6.6.2.3.1 for the complete definition of this element.

CloudTypes (Enumeration)

Enumeration for types of cloud.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\CloudTypes*. See Clause 5.7.6.6.2.3.9 for the complete definition of this element.

Compass16 (Enumeration)

Enumeration for compass points, including 12 intercardinal points.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\Compass16*. See Clause 5.7.6.6.2.5.4 for the complete definition of this element.

Convection (Class)

Convection hazard.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\Convection*. See Clause 5.7.6.6.1.5.1 for the complete definition of this element.

Growth (Enumeration)

Enumeration for storm growth (taken from CCFP).

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\Growth*. See Clause 5.7.6.6.2.5.8 for the complete definition of this element.

Hydrometeor (Class)

Hydrometeor feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\Hydrometeor*. See Clause 5.7.6.6.2.3.2 for the complete definition of this element.

HydrometeorTypes (Enumeration)

Enumeration for types of hydrometeor.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\HydrometeorTypes*. See Clause 5.7.6.6.2.3.10 for the complete definition of this element.

Icing (Class)

Icing hazard.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\Icing*. See Clause 5.7.6.6.1.5.2 for the complete definition of this element.

IcingTypes (Enumeration)

Enumeration for types of icing.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\IcingTypes*. See Clause 5.7.6.6.1.5.7 for the complete definition of this element.

Lithometeor (Class)

Lithometeor feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\Lithometeor*. See Clause 5.7.6.6.2.3.3 for the complete definition of this element.

MountainObscuration (Class)

Mountain obscuration hazard.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\MountainObscuration*. See Clause 5.7.6.6.1.5.3 for the complete definition of this element.

MountainWave (Class)

Mountain wave hazard.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\MountainWave*. See Clause 5.7.6.6.1.5.4 for the complete definition of this element.

MovementDescription (Class)

Movement description feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\MovementDescription*. See Clause 5.7.6.6.2.11.2 for the complete definition of this element.

ObscurationTypes (Enumeration)

Enumeration for obscuration types.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\ObscurationTypes*. See Clause 5.7.6.6.2.11.6 for the complete definition of this element.

OtherPhenomenon (Class)

Other phenomenon feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\OtherPhenomenon*. See Clause 5.7.6.6.2.11.3 for the complete definition of this element.

OtherPhenomenonTypes (Enumeration)

Enumeration for other types of phenomenon.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\OtherPhenomenonTypes*. See Clause 5.7.6.6.2.11.7 for the complete definition of this element.

Precipitation (Class)

Precipitation feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\Precipitation*. See Clause 5.7.6.6.2.3.4 for the complete definition of this element.

PrecipitationTypes (Enumeration)

Enumeration for precipitation types.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\WX\WX_PhenomBase\PrecipitationTypes. See Clause 5.7.6.6.2.11.8 for the complete definition of this element.

RadioactiveCloud (Class)

Radioactive cloud feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\RadioactiveCloud*. See Clause 5.7.6.6.2.3.5 for the complete definition of this element.

Shower (Class)

Shower feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\Shower*. See Clause 5.7.6.6.2.3.6 for the complete definition of this element.

Thunderstorm (Class)

Thunderstorm feature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud\Thunderstorm*. See Clause 5.7.6.6.2.3.7 for the complete definition of this element.

TropicalCyclone (Class)

Tropical cyclone hazard.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\AVWX\AVWX_Phenom_TropicalCyclone\TropicalCyclone. See Clause 5.7.6.6.1.6.1 for the complete definition of this element.

Turbulence (Class)

Turbulence hazard.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\AVWX\AVWX_Phenom_Hazards\Turbulence. See Clause 5.7.6.6.1.5.5 for the complete definition of this element.

TurbulenceTypes (Enumeration)

Enumeration for types of turbulence.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\AVWX\AVWX_Phenom_Hazards\TurbulenceTypes. See Clause 5.7.6.6.1.5.8 for the complete definition of this element.

VerticallyBoundedPhenomenon (Abstract Class)

A weather phenomenon that is vertically bounded with a base and/or top.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\WX\WX_PhenomBase\VerticallyBoundedPhenomenon. See Clause 5.7.6.6.2.11.4 for the complete definition of this element.

VolcanicActivity (Class)

Volcanic activity.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\AVWX\AVWX_Phenom_Volcanic\VolcanicActivity. See Clause 5.7.6.6.1.7.1 for the complete definition of this element.

VolcanicAshCloud (Class)

Volcanic ash cloud.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Volcanic\VolcanicAshCloud*. See Clause 5.7.6.6.1.7.2 for the complete definition of this element.

WeatherIntensity (Enumeration)

Enumeration for weather intensity descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\WeatherIntensity*. See Clause 5.7.6.6.2.5.12 for the complete definition of this element.

WindLayer (Class)

Wind layer hazard.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Wind\WindLayer*. See Clause 5.7.6.6.1.8.1 for the complete definition of this element.

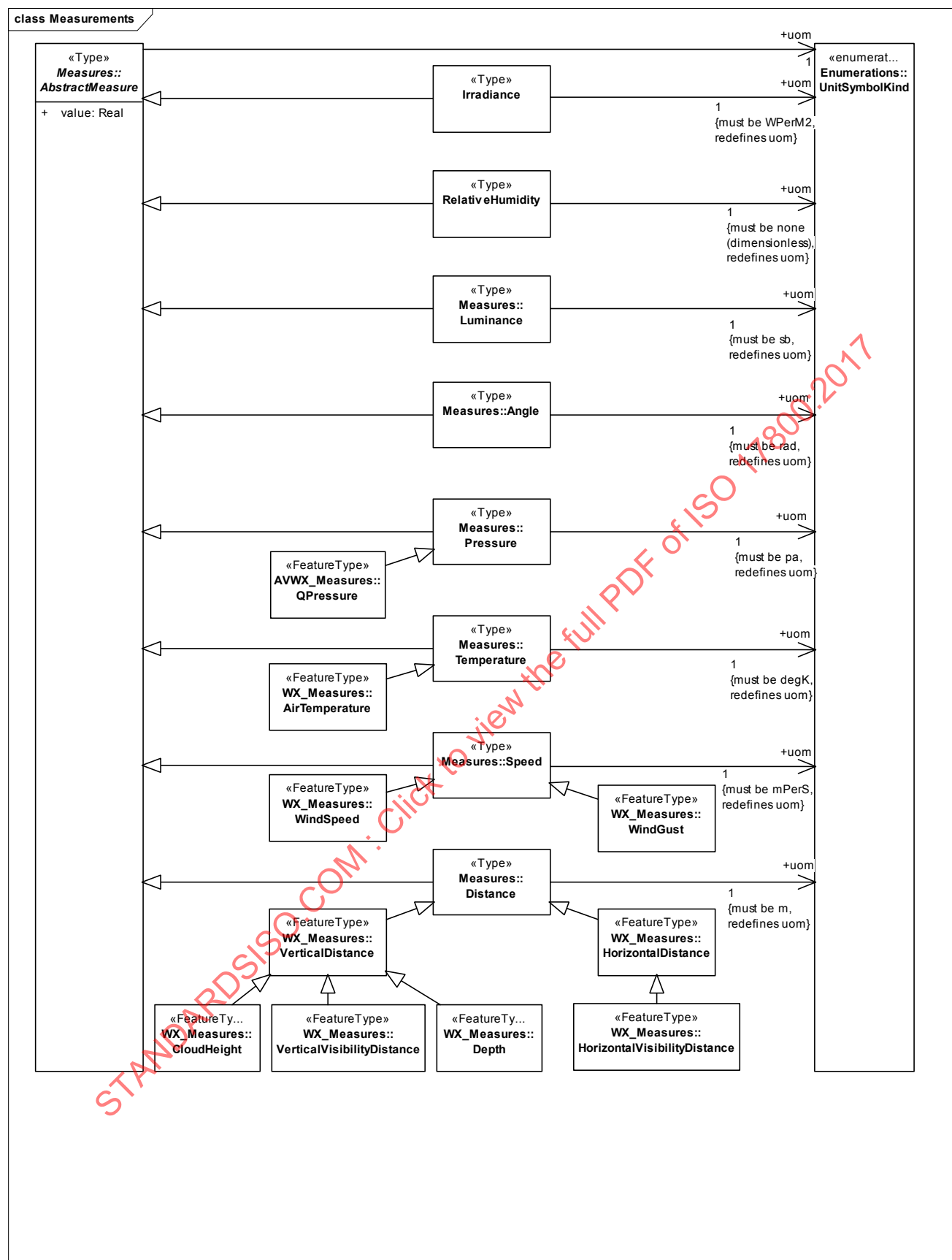
WindShear (Class)

Wind shear hazard.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\WindShear*. See Clause 5.7.6.6.1.5.6 for the complete definition of this element.

5.6.4.5. Measurements Diagram

Measurements make up the state information for an FSGIMWeather object.



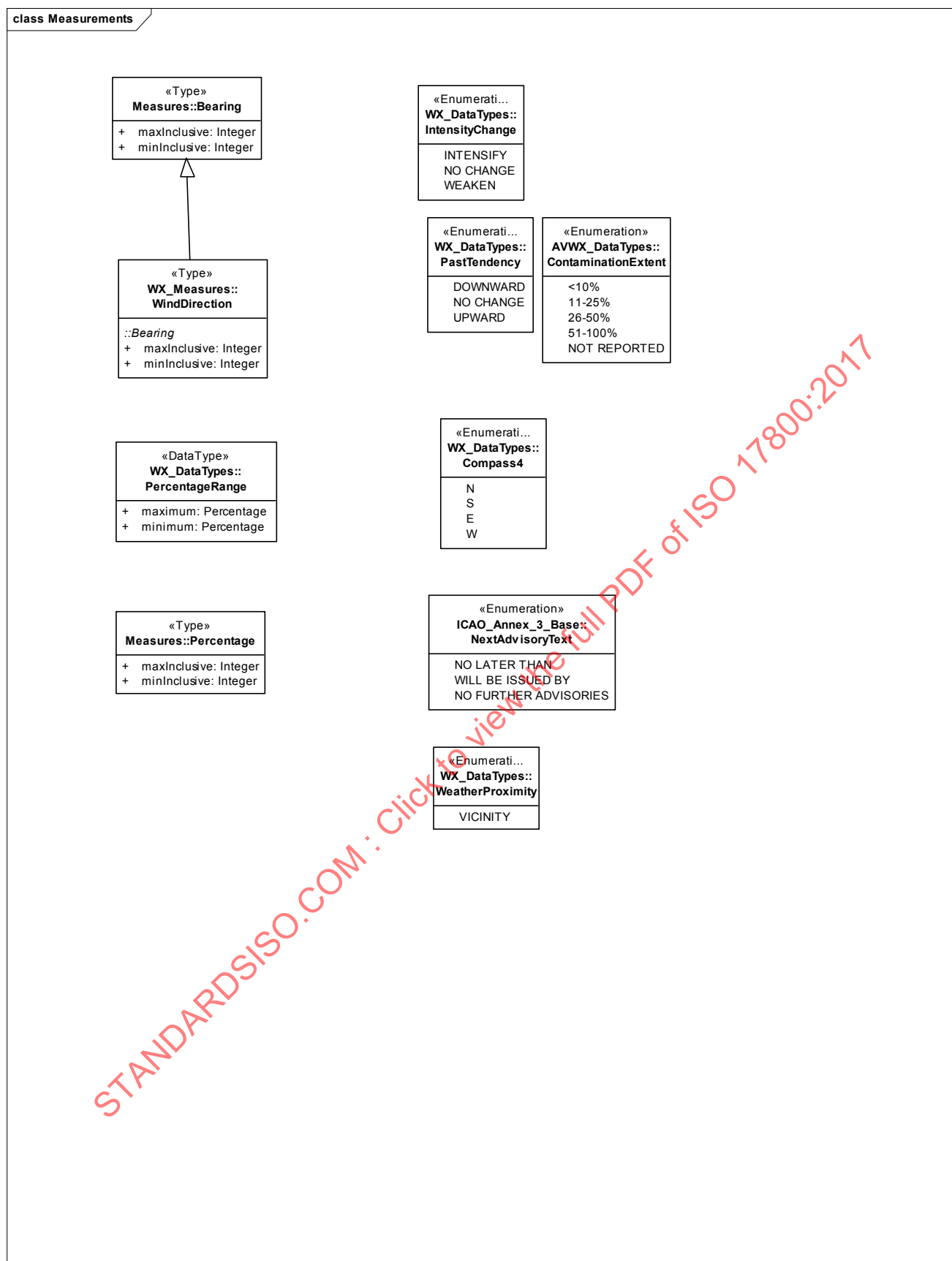


Figure 5.34 - Measurements

The Measurements Diagram makes use of the following elements:

AbstractMeasure (Abstract Class)

Abstract measurement class. Each measurement has a unit of measure, uom. WXXM allows more than one uom -- e.g., F/C/K for temperature. FSGIM only allows one choice of uom for a physical quantity.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\AbstractMeasure*. See Clause 5.7.6.6.2.1.1 for the complete definition of this element.

AirTemperature (Class)

Air temperature.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\AirTemperature*. See Clause 5.7.6.6.2.8.1 for the complete definition of this element.

Angle (Class)

Generic angle class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Angle*. See Clause 5.7.6.6.2.1.2 for the complete definition of this element.

Bearing (Class)

Bearing is used to represent direction in the 2D coordinate reference system. It is an "angle measured from true north". It is represented as an angle restricted to the range greater than 0 and up to and including 360 degrees which is true north.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Bearing*. See Clause 5.7.6.6.2.1.3 for the complete definition of this element.

CloudHeight (Class)

Cloud height.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\CloudHeight*. See Clause 5.7.6.6.2.8.2 for the complete definition of this element.

Compass4 (Enumeration)

Enumeration for cardinal compass points.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\Compass4*. See Clause 5.7.6.6.2.5.5 for the complete definition of this element.

ContaminationExtent (Enumeration)

Enumeration for contamination extent ranges.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_DataTypes\ContaminationExtent*. See Clause 5.7.6.6.1.2.2 for the complete definition of this element.

Depth (Class)

Depth.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\Depth*. See Clause 5.7.6.6.2.8.3 for the complete definition of this element.

Distance (Class)

Generic distance class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Distance*. See Clause 5.7.6.6.2.1.4 for the complete definition of this element.

HorizontalDistance (Class)

Horizontal distance.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\HorizontalDistance*. See Clause 5.7.6.6.2.8.4 for the complete definition of this element.

HorizontalVisibilityDistance (Class)

Restricts HorizontalDistance to an integer value greater than or equal to zero.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\WX\WX_Measures\HorizontalVisibilityDistance. See Clause 5.7.6.6.2.8.5 for the complete definition of this element.

IntensityChange (Enumeration)

Enumeration for intensity change descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\IntensityChange*.

See Clause 5.7.6.6.2.5.9 for the complete definition of this element.

Irradiance (Class)

The power of electromagnetic radiation per unit area incident on a surface, measured in W/m².

UML Element Location: *Model\Device and Model Components\Energy Manager*

Component\FSGIMWeather\Irradiance. See Clause 5.6.7.12.6 for the complete definition of this element.

Luminance (Class)

Density of luminous intensity.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Luminance*. See Clause

5.7.6.6.2.1.5 for the complete definition of this element.

NextAdvisoryText (Enumeration)

Enumeration of next advisory projection.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\AVWX\ICAO_Annex_3_Base\NextAdvisoryText. See Clause 5.7.6.6.1.9.3 for the complete definition of this element.

PastTendency (Enumeration)

Enumeration for past tendency descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\PastTendency*. See

Clause 5.7.6.6.2.5.10 for the complete definition of this element.

Percentage (Class)

Generic percentage class; value is limited to between 0 and 100 inclusive.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Percentage*. See Clause

5.7.6.6.2.1.6 for the complete definition of this element.

PercentageRange (Class)

Percentage range.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\PercentageRange*.

See Clause 5.7.6.6.2.5.1 for the complete definition of this element.

Pressure (Class)

Generic pressure class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Pressure*. See Clause

5.7.6.6.2.1.7 for the complete definition of this element.

QPressure (Class)

Restricts Pressure to an integer value between 500 and 1100 (inclusive).

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Measures\QPressure*. See

Clause 5.7.6.6.1.4.1 for the complete definition of this element.

RelativeHumidity (Class)

The relative humidity, the ratio of the partial pressure of water vapor to the equilibrium vapor pressure at constant temperature.

UML Element Location: *Model\Device and Model Components\Energy Manager*

Component\FSGIMWeather\RelativeHumidity. See Clause 5.6.7.12.7 for the complete definition of this element.

Speed (Class)

Generic speed class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Speed*. See Clause 5.7.6.6.2.1.8 for the complete definition of this element.

Temperature (Class)

Generic temperature class.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Temperature*. See Clause 5.7.6.6.2.1.9 for the complete definition of this element.

UnitSymbolKind (Enumeration)

This identifies the units of measure based on the NIST Special Publication 811 2008 edition -- Guide for the Use of the International System of Units (SI). Note that these reference document supports SI units as well as common units of measurements in practice including CGS units.

Only VA, W, var, VAh, Wh, varh, Btu, J, therm, BtuPerh, g, gPerM3, gPerS, m, mPerS, WPerM2, rad, sb, pa, and degK are normative for the FSGIM standard. The enumerated values of VA, W, var, VAh, Wh, and varh were changed from their original NAESB values of vA, w, vAr, vAH, wH, and vArH to make them consistent with the standard SI abbreviations for these units. The enumerated values of Btu and BtuPerh were changed from their original NAESB values of btu and btuPerH to make them consistent with the standard abbreviations for British thermal units and hours.

Note: The abbreviation MBtu is often used to represent thousands of Btu's (where M represents the Roman numeral for 1000). The FSGIM *always* uses the SI prefixes listed in the *SiScaleCodeType* enumeration and not the prefixes based on Roman numerals. This means that 1000 Btu's would be represented in the FSGIM as *EnergyThermalQuantity.quantity = 1, EnergyThermalType.itemUnits = Btu, and EnergyThermalType.siScaleCode = k, not EnergyThermalType.siScaleCode = M*.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\UnitSymbolKind*. See Clause 6.6.2.6 for the complete definition of this element.

VerticalDistance (Class)

Vertical distance.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\VerticalDistance*. See Clause 5.7.6.6.2.8.6 for the complete definition of this element.

VerticalVisibilityDistance (Class)

Restricts *VerticalDistance* to an integer value greater than or equal to zero.

UML Element Location: *Model\Model Elements from External*

Sources\WXXM\WX\WX_Measures\VerticalVisibilityDistance. See Clause 5.7.6.6.2.8.7 for the complete definition of this element.

WeatherProximity (Enumeration)

Enumeration for weather intensity descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\WeatherProximity*. See Clause 5.7.6.6.2.5.13 for the complete definition of this element.

WindDirection (Class)

WindDirection is a Bearing expressed as an integer.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\WindDirection*. See Clause 5.7.6.6.2.8.8 for the complete definition of this element.

WindGust (Class)

Wind gust.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\WindGust*. See Clause 5.7.6.6.2.8.9 for the complete definition of this element.

WindSpeed (Class)

Restriction of Speed to an integer value greater than or equal to one.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\WindSpeed*. See Clause 5.7.6.6.2.8.10 for the complete definition of this element.

5.6.5. Key Energy Manager Concepts

This clause covers the portion of the FSGIM that provides information elements supporting the key roles of managing, monitoring, and reporting the energy aspects of all or some portion of a facility, including interfacing within a hierarchy of energy managers. It ties together present measurement and control with forecasts of future needs and historical records of past performance. Some energy managers interface with entities outside of the facility to exchange information including weather, emissions, energy pricing and cost, and energy demand and supply.

UML Package Location: *Model\Device and Model Components\Energy Manager Component\Key Energy Manager Concepts*

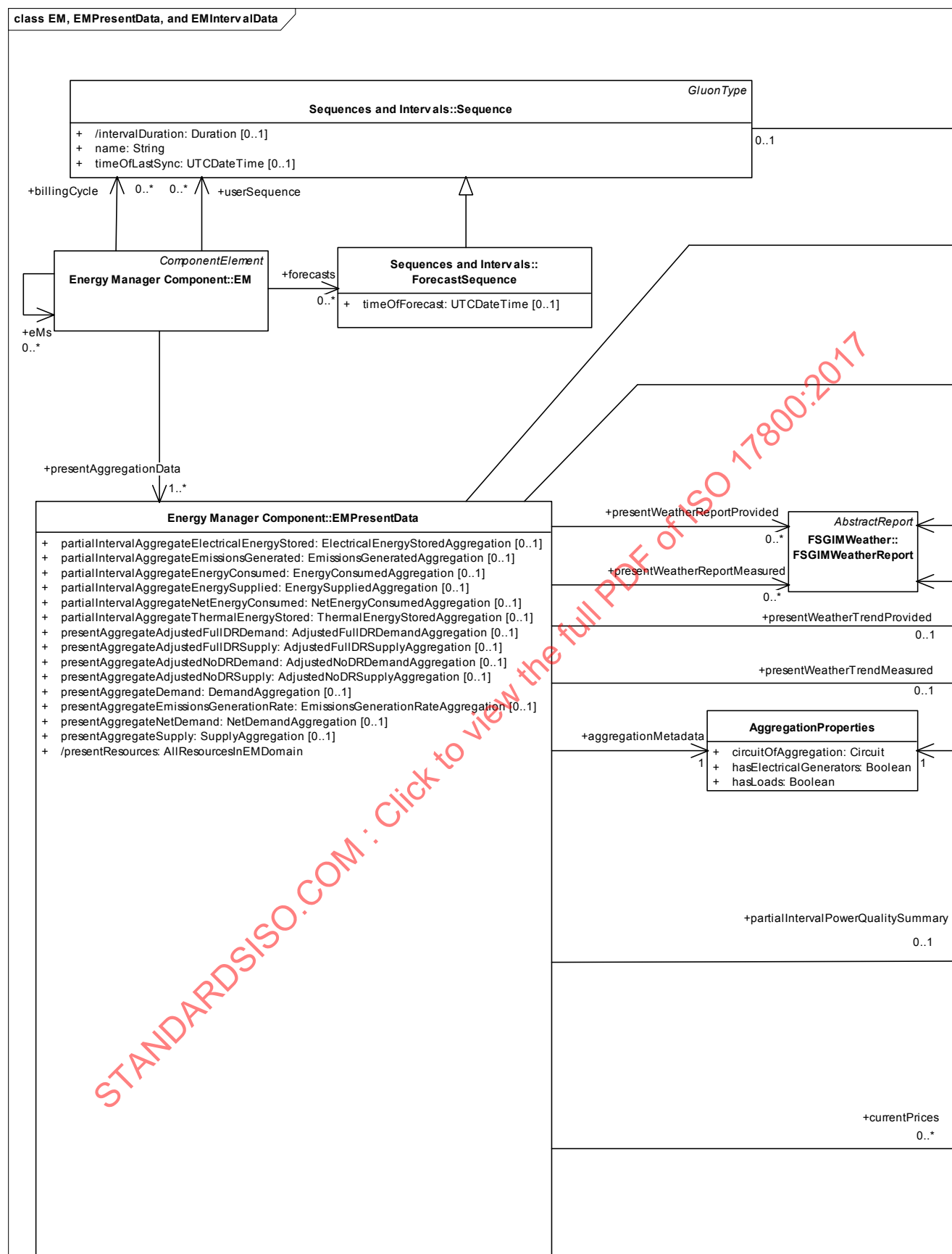
5.6.5.1. EM, EMPresentData, and EMIntervalData Diagram

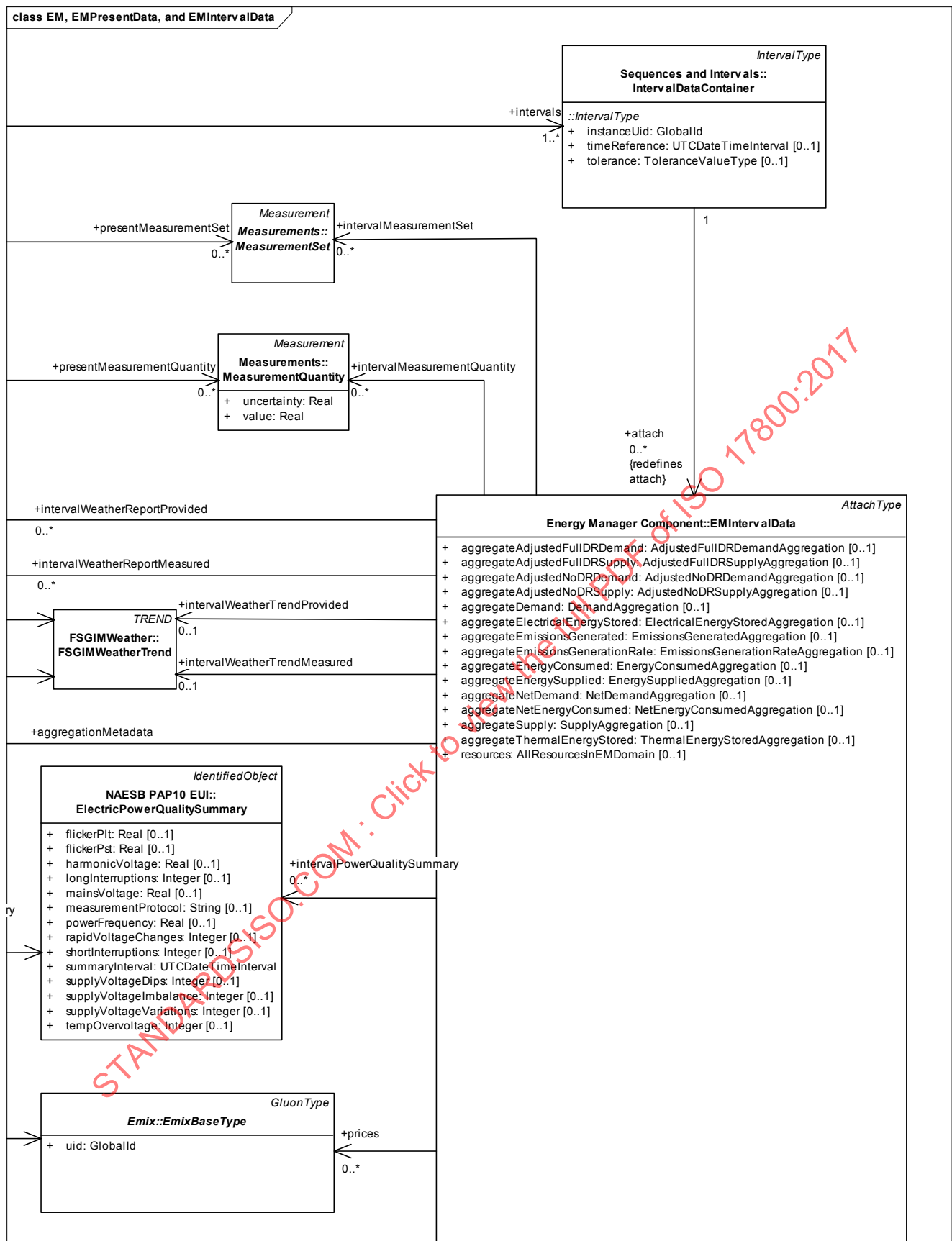
The EM, EMPresentData, and EMIntervalData diagram describes the information elements that support key energy management functions.

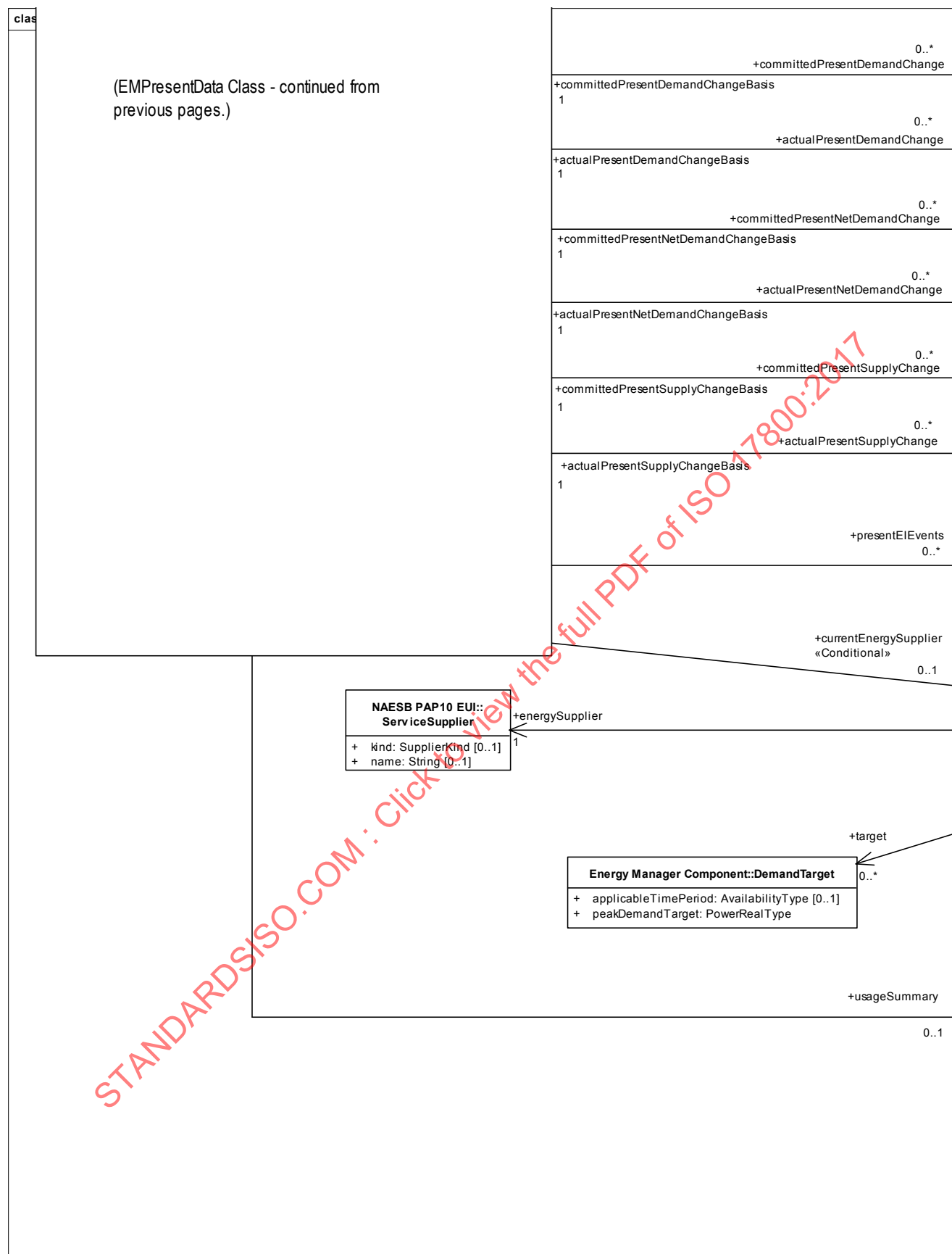
The EM Class serves as the main class for modeling an energy manager. It has associations, either directly or indirectly, to all of the subordinate resources in its domain, along with associations to the real-time, historical, and forecast data for this energy manager.

The EMPresentData Class contains key real-time measurements. Some measurements, such as PowerMeasurementsSets, contain instantaneous rate values, while others, such as EnergyMeasurementsSets, contain accumulated values for a partially completed (filling) time interval. When the partially completed interval finishes, a new filling interval begins.

The EMIntervalData Class contains measurements analogous to those in the EM Class, but they are associated with a complete historic or future interval of time.







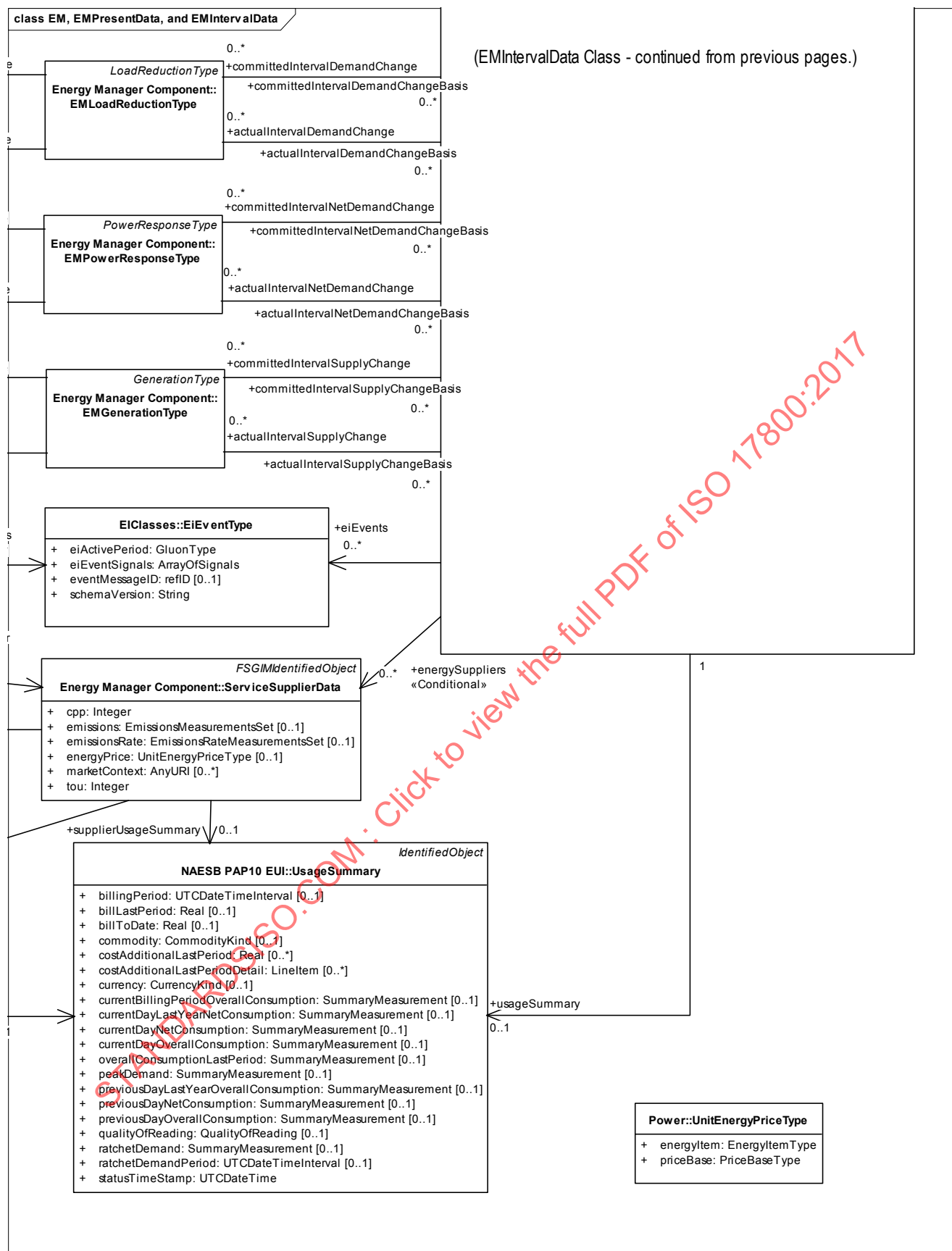


Figure 5.35 - EM, EMPresentData, and EMIntervalData

The EM, EMPresentData, and EMIntervalData Diagram makes use of the following elements:

AggregationProperties (Class)

This class defines a set of attributes that characterize a Circuit with respect to Loads and Generators that are directly connected to it or indirectly connected through other Circuits that contribute to aggregations performed on this Circuit.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Key Energy Manager Concepts\AggregationProperties*. See Clause 5.6.7.13.1 for the complete definition of this element.

DemandTarget (Class)

This class defines a demand target during a time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\DemandTarget*. See Clause 5.6.7.1 for the complete definition of this element.

EiEventType (Class)

Core element of event-based demand response that specifies the duration and applicability of an Event. Some deadlines, time periods, and transitions may not be applicable to all products or services.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiEventType*. See Clause 5.7.1.2.1.8 for the complete definition of this element.

ElectricPowerQualitySummary (Class)

A summary of power quality events. This information represents a summary of power quality information typically required by energy management systems. It is not intended to satisfy the detailed requirements of power quality monitoring. All values are as defined by measurementProtocol during the period. The standards typically also give ranges of allowed values. See referenced standards for definition, measurement protocol, and period.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB_PAP10 EUI\ElectricPowerQualitySummary*. See Clause 5.7.5.2.1.5 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

EMGenerationType (Class)

This class represents net demand reductions resulting from supply increases. The class supports power ramps that describe the power profile for a generation resource when it is energized (stagingRamps) or de-energized (recoveryRamps).

For each array element of a generation stagingRamp, the stagingRamp[n].rate defines the amount of power increase and the stagingRamp[n].duration defines the length of time in seconds upon which this increase is in effect. If the stagingRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the generation stagingRamp attribute name implies that the demand is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

For each array element of a generator recoveryRamp, the recoveryRamp[n].rate defines the amount of power decrease and the recoveryRamp[n].duration defines the length of time in seconds upon which this decrease is in effect. If the recoveryRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the generator recoveryRamp attribute name implies that the demand reduction is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMGenerationType*. See Clause 5.6.7.3 for the complete definition of this element.

EMIntervalData (Class)

This class represents data in the domain of concern of an Energy Manager over some complete interval of time. This time interval may be in the past or future.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMIntervalData*. See Clause 5.6.7.4 for the complete definition of this element.

EmixBaseType (Abstract Class)

iCalendar-derived object to host EMIX (OASIS EMIX v1.0) elements.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixBaseType*. See Clause 5.7.2.13.1.2 for the complete definition of this element.

EMLoadReductionType (Class)

This class represents demand reductions from Loads in response to a demand response event. The class supports power ramps that describe the power profile for a resource when it is de-energized (stagingRamps) or energized (recoveryRamps).

For each array element of a stagingRamp, the stagingRamp[n].rate defines the amount of power decrease and the stagingRamp[n].duration defines the length of time in seconds upon which this decrease is in effect. If the stagingRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the stagingRamp attribute name implies that the demand reduction is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be decreasing.

For each array element of a recoveryRamp, the recoveryRamp[n].rate defines the amount of power increase and the recoveryRamp[n].duration defines the length of time in seconds upon which this increase is in effect. If the recoveryRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the recoveryRamp attribute name implies that the demand is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMLoadReductionType*. See Clause 5.6.7.5 for the complete definition of this element.

EMPowerResponseType (Class)

This class represents net demand reductions resulting from a combination of reductions in demand (EMLoadReductionType) and increases in supply (EMGenerationType).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPowerResponseType*. See Clause 5.6.7.6 for the complete definition of this element.

EMPresentData (Class)

The container that directly holds all of the aggregated present data and indirectly holds the associated present context.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPresentData*. See Clause 5.6.7.7 for the complete definition of this element.

ForecastSequence (Class)

The Sequence that is used for storing all forecasts, including forecasts for time periods where the originally forecasted time period has now passed.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\ForecastSequence*. See Clause 5.6.7.10.2 for the complete definition of this element.

FSGIMWeatherReport (Class)

A container class for weather observations, forecasts, and trends.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherReport*. See Clause 5.6.7.12.4 for the complete definition of this element.

FSGIMWeatherTrend (Class)

A sequence of forecasts of weather in increments of a defined duration such as hourly.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherTrend*. See Clause 5.6.7.12.5 for the complete definition of this element.

IntervalDataContainer (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval. This is a specialization of the WSCalendar PIM IntervalType.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\IntervalDataContainer*. See Clause 5.6.7.10.3 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be surmised by monitoring the time references.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*. See Clause 5.6.7.10.4 for the complete definition of this element.

ServiceSupplier (Class)

Organization that provides services to customers.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\ServiceSupplier*. See Clause 5.7.5.2.1.23 for the complete definition of this element.

ServiceSupplierData (Class)

This class contains information associated with the generated emissions, quality of power, and energy-price units associated with an identified supplier of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\ServiceSupplierData*. See Clause 5.6.7.9 for the complete definition of this element.

UnitEnergyPriceType (Class)

Price per Unit of Energy, i.e., Power times Duration

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\UnitEnergyPriceType*. See Clause 5.7.2.13.4.8 for the complete definition of this element.

UsageSummary (Class)

Summary of usage for a billing period.

Note: This model was extended using the work of OpenADE 2015 for ESPI. These extensions are proposed for the next revision of the NAESB standard REQ.21. Also extended are SummaryMeasurement and LineDetail.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\UsageSummary. See Clause 5.7.5.2.1.27 for the complete definition of this element.

5.6.6. EnergyRouter

The EnergyRouter concept is used to handle situations where loads can receive power from more than one source. When this occurs, the Energy Manager Component must be able to determine which power source or combination of power sources are driving the loads in order for it to properly make energy management decisions. Examples of where this might occur include loads that can be powered by the grid or a backup generator, switchgear capable of selecting between two utility feeds, configurations that allow photovoltaic arrays to supplement power from the grid, or configurations that allow photovoltaic arrays to send excess power to the grid. Specializations of the EnergyRouter are provided to handle each of these scenarios by providing the Energy Manager Component with the knowledge that it needs in order to determine the resulting power flows.

UML Package Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter*

5.6.6.1. EnergyRouter Diagram

This diagram illustrates the classes used to model the electrical or other energy connections between the component models that are used to represent physical devices. These connections are captured in the ConnectionPoint.connectedTo attribute. Object instances that have the same value for the connectedTo attribute are connected to each other for the purpose of modeling energy transfers.

In the case of electrical connections, a ConnectionPoint represents more than one physical terminal on a device. For example, the hot and neutral terminals on a single phase device would be a single ConnectionPoint. The three or four terminals on a three-phase device would also be a single ConnectionPoint.

The RouterConnectionPoint adds additional attributes to the ConnectionPoint Class to represent the capabilities of the routable ConnectionPoints, e.g., whether or not they *can be* disconnected, and the present state of the ConnectionPoints, e.g., whether they currently *are* disconnected.

Specializations of the EnergyRouter Class are provided for handling common energy routing scenarios. These specializations incorporate constraints on the RouterConnectionPoints associated with the Energy Router so that the constraints do not need to be explicitly added each time one of these scenarios need to be addressed using the model. Applying these constraints to the RouterConnectionPoints and then making use of Kirchhoff's Circuit Laws allows the Energy Manager to determine the resulting power flows. These specializations also explain how Aggregations are affected for this particular type of EnergyRouter.

The EnergyRouter Class only has three RouterConnectionPoints attached to it. Scenarios that require more than three RouterConnectionPoints can be modeled by chaining more than one EnergyRouter together.

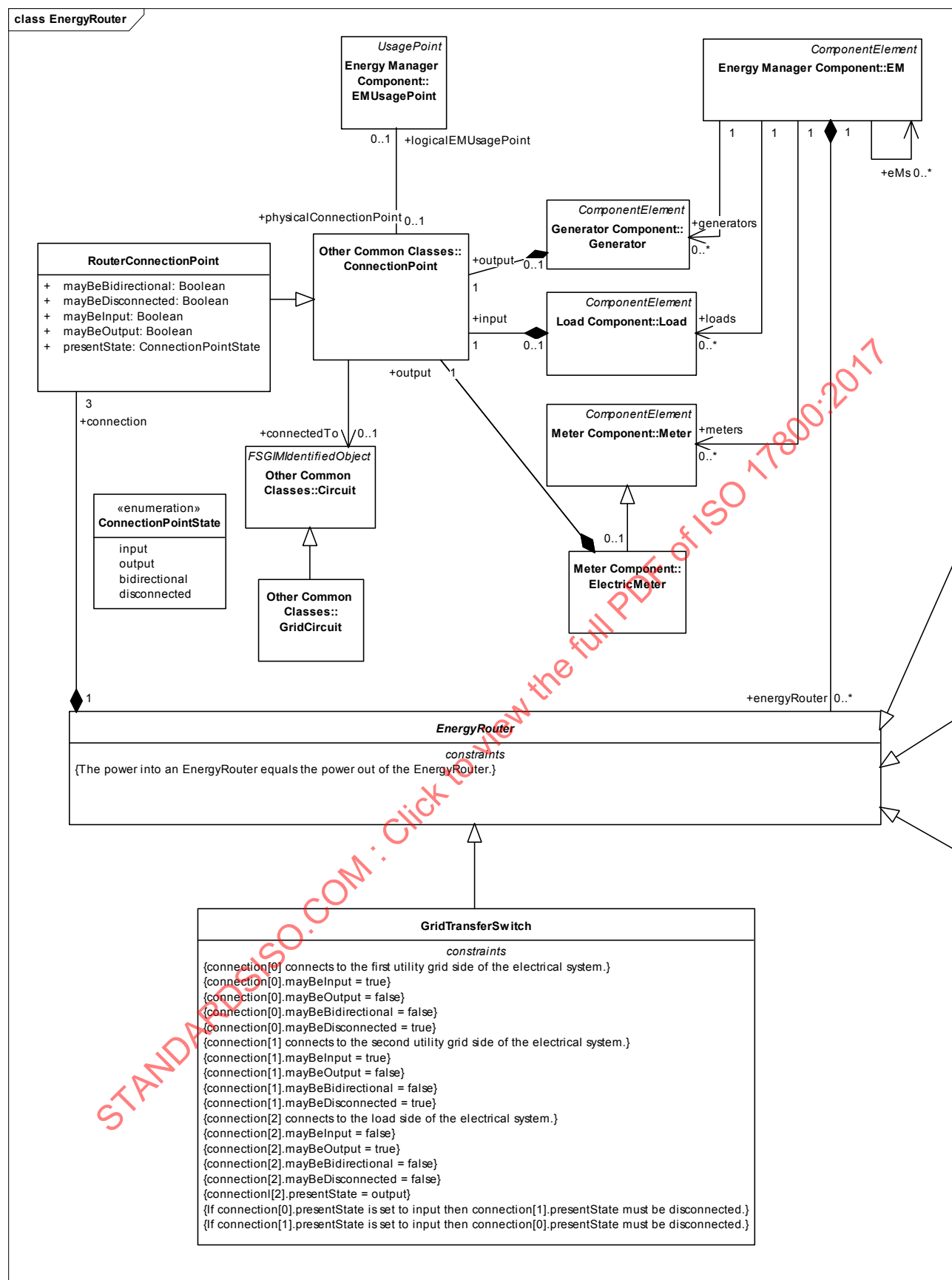




Figure 5.36 - EnergyRouter

The EnergyRouter Diagram makes use of the following elements:

BidirectionalCombiner (Class)

This class is used to model bidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that allows energy to flow from the onsite generation source to the utility grid but not from the utility grid to the onsite generation source. The effect of the constraints that are placed on this class results in the following behavior for the bidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.
- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power sent back to the grid is the amount of power available from the onsite generation side of the combiner box minus the amount of power drawn from load side of the combiner box.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\BidirectionalCombiner*. See Clause 5.6.7.14.1 for the complete definition of this element.

Circuit (Class)

This class is used to model electrical circuits. All ConnectionPoints that are electrically tied together will point to the same instance of this class.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\Circuit*. See Clause 6.6.5.1 for the complete definition of this element.

ConnectionPoint (Class)

This class is used to model the physical connections from Generators, Loads, Meters, and Energy Routers to sources or sinks of energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\ConnectionPoint*. See Clause 6.6.5.2 for the complete definition of this element.

ConnectionPointState (Enumeration)

This enumeration is used to specify the present configuration of a ConnectionPoint.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\ConnectionPointState*. See Clause 5.6.7.14.11 for the complete definition of this element.

ElectricMeter (Class)

The ElectricMeter Class is a subclass of the Meter Class. It is an abstract representation of devices that measure real and reactive power or energy. These devices optionally measure other electrical parameters such as apparent power, apparent energy, present subinterval demand or power quality.

UML Element Location: *Model\Device and Model Components\Meter Component\ElectricMeter*. See Clause 5.3.2.2 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

EMUsagePoint (Class)

Logical point on a network at which consumption or production is either measured (e.g., metered) or estimated (e.g., unmetered street lights).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMUsagePoint*. See Clause 5.6.7.8 for the complete definition of this element.

EnergyRouter (Abstract Class)

This class is used to model the various devices that are capable of connecting a load to more than one source of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\EnergyRouter*. See Clause 5.6.7.14.3 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

GridCircuit (Class)

This class is used to model electrical circuits that are part of the utility grid.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\GridCircuit*. See Clause 6.6.5.3 for the complete definition of this element.

GridTransferSwitch (Class)

This class is used to model switchgear that are used to allow a load to be switched between two utility feeds without allowing the two utility feeds to be connected together through the switchgear.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\GridTransferSwitch*. See Clause 5.6.7.14.4 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

RouterConnectionPoint (Class)

This class is used to model the input/output characteristics of connections from an EnergyRouter to a source or sink of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\RouterConnectionPoint*. See Clause 5.6.7.14.6 for the complete definition of this element.

TransferSwitch (Class)

This class is used to model transfer switches, which are devices that allow a load or set of loads to be switched between two power sources without allowing the two power sources to be connected together.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\TransferSwitch*. See Clause 5.6.7.14.7 for the complete definition of this element.

UnidirectionalCombiner (Class)

This class is used to model unidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that does not allow energy to flow from one source to another. The effect of the constraints that are placed on this class results in the following behavior for the unidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn

from utility grid side of the combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.

- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of the combiner box is zero and the amount of power actually injected into the onsite generation side of the combiner box is equal to the amount of power drawn from load side of the combiner box.

UML Element Location: *Model\Device and Model Components\Energy Manager*

Component\EnergyRouter\UnidirectionalCombiner. See Clause 5.6.7.14.9 for the complete definition of this element.

5.6.6.2. EnergyRouter Aggregations Diagram

This diagram explains how Aggregations are affected by the presence of EnergyRouters in Circuits. The EnergyRouter Rulesets apply to all Aggregations, both standard and custom, unless explicitly stated otherwise in the Rulesets for those Aggregations.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

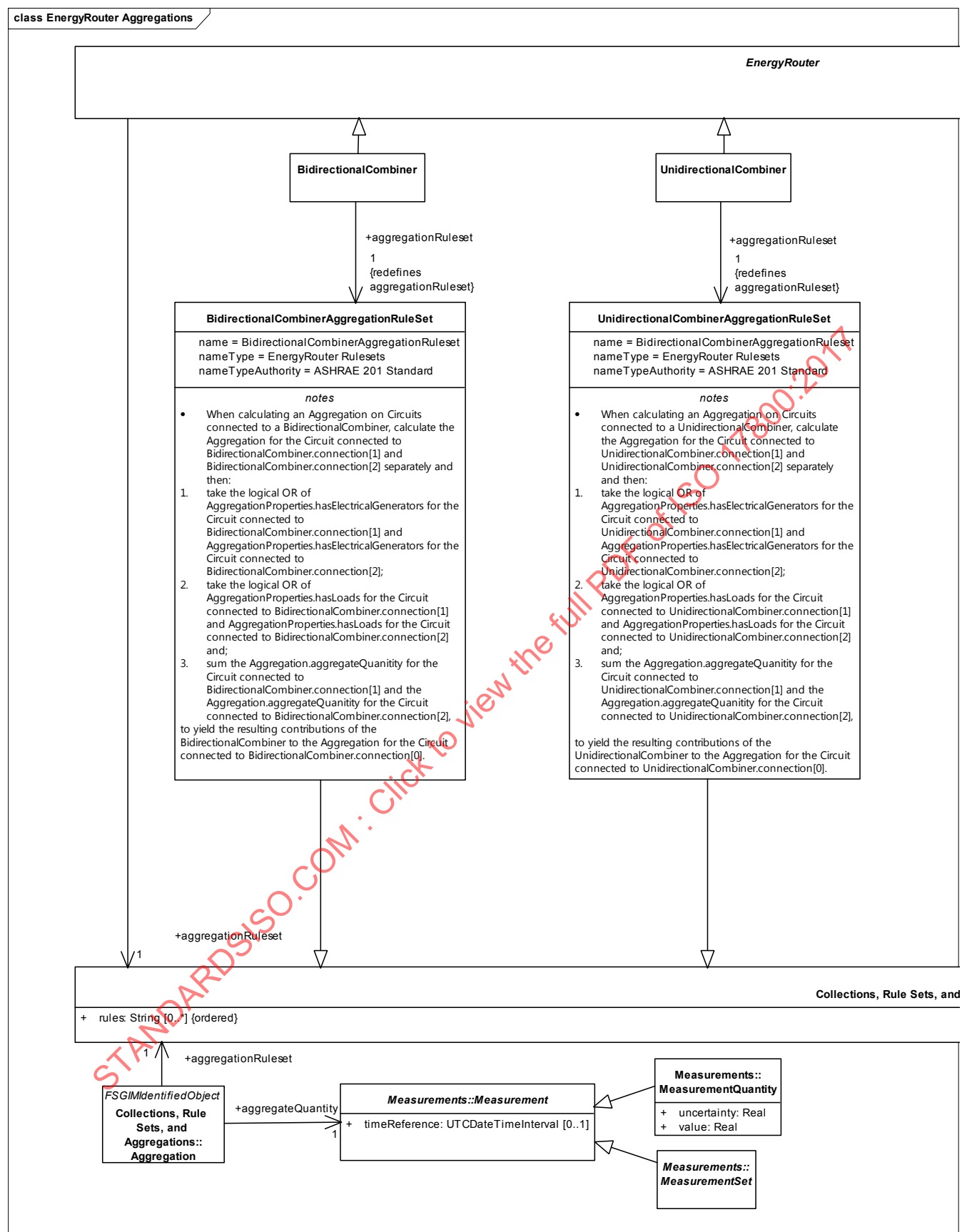




Figure 5.37 - EnergyRouter Aggregations

The EnergyRouter Aggregations Diagram makes use of the following elements:

Aggregation (Class)

An aggregation allows the assembly of statistics of interest arrived at through defined algorithms applied to other parts of the model. The description of an aggregation includes all of the inputs that comprise the algorithms as well as the outputs of the algorithm. An aggregation instance represents one or more aggregateQuantities that are computed by applying aggregation rules for each quantity against a Collection or FilteredCollection of Components.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Aggregation*. See Clause 5.6.7.11.1 for the complete definition of this element.

BidirectionalCombiner (Class)

This class is used to model bidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that allows energy to flow from the onsite generation source to the utility grid but not from the utility grid to the onsite generation source. The effect of the constraints that are placed on this class results in the following behavior for the bidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.
- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power sent back to the grid is the amount of power available from the onsite generation side of the combiner box minus the amount of power drawn from load side of the combiner box.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\BidirectionalCombiner*. See Clause 5.6.7.14.1 for the complete definition of this element.

BidirectionalCombinerAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a BidirectionalCombiner, calculate the Aggregation for the Circuit connected to BidirectionalCombiner.connection[1] and BidirectionalCombiner.connection[2] separately and then:
 1. take the logical OR of AggregationProperties.hasElectricalGenerators for the Circuit connected to BidirectionalCombiner.connection[1] and AggregationProperties.hasElectricalGenerators for the Circuit connected to BidirectionalCombiner.connection[2];
 2. take the logical OR of AggregationProperties.hasLoads for the Circuit connected to BidirectionalCombiner.connection[1] and AggregationProperties.hasLoads for the Circuit connected to BidirectionalCombiner.connection[2] and;
 3. sum the Aggregation.aggregateQuantity for the Circuit connected to BidirectionalCombiner.connection[1] and the Aggregation.aggregateQuantity for the Circuit connected to BidirectionalCombiner.connection[2], to yield the resulting contributions of the BidirectionalCombiner to the Aggregation for the Circuit connected to BidirectionalCombiner.connection[0].

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\BidirectionalCombinerAggregationRuleSet*. See Clause 5.6.7.14.2 for the complete definition of this element.

ConnectionPointState (Enumeration)

This enumeration is used to specify the present configuration of a ConnectionPoint.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\ConnectionPointState*. See Clause 5.6.7.14.11 for the complete definition of this element.

EnergyRouter (Abstract Class)

This class is used to model the various devices that are capable of connecting a load to more than one source of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\EnergyRouter*. See Clause 5.6.7.14.3 for the complete definition of this element.

GridTransferSwitch (Class)

This class is used to model switchgear that are used to allow a load to be switched between two utility feeds without allowing the two utility feeds to be connected together through the switchgear.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\GridTransferSwitch*. See Clause 5.6.7.14.4 for the complete definition of this element.

GridTransferSwitchAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a GridTransferSwitch, calculate the Aggregation for the Circuit connected to GridTransferSwitch.connection[2].
- If GridTransferSwitch.connection[0].presentState has a value of "input," report:
 1. the Aggregation.aggregateQuantity for the Circuit connected to GridTransferSwitch.connection[2] as the contribution of the GridTransferSwitch to the Aggregation for the Circuit connected to GridTransferSwitch.connection[0], and with
 2. the AggregationProperties.hasElectricalGenerators, and the AggregationProperties.hasLoads for the Circuit connected to GridTransferSwitch.connection[2] as the contribution of the GridTransferSwitch to the AggregationProperties for the Circuit connected to GridTransferSwitch.connection[0].
- If GridTransferSwitch.connection[0].presentState has a value of "disconnected," report:
 1. the results of the Aggregation for the Circuit connected to GridTransferSwitch.connection[2] with all associated MeasurementQuantity.values, including those indirectly associated as part of a MeasurementSet, set to 0 and all associated MeasurementQuantity.uncertainty values, including those indirectly associated as part of a MeasurementSet, set to 0, as the contribution of the GridTransferSwitch to the Aggregation for the Circuit connected to GridTransferSwitch.connection[0], and with
 2. AggregationProperties.hasElectricalGenerators set to False and AggregationProperties.hasLoads set to False as the contribution of the GridTransferSwitch to the AggregationProperties for the Circuit connected to GridTransferSwitch.connection[0].
- If GridTransferSwitch.connection[1].presentState has a value of "input," report:
 1. the Aggregation.aggregateQuantity for the Circuit connected to GridTransferSwitch.connection[2] as the contribution of the GridTransferSwitch to the Aggregation for the Circuit connected to GridTransferSwitch.connection[1], and with
 2. the AggregationProperties.hasElectricalGenerators, and the AggregationProperties.hasLoads for the Circuit connected to GridTransferSwitch.connection[2] as the contribution of the GridTransferSwitch to the AggregationProperties for the Circuit connected to GridTransferSwitch.connection[1].
- If GridTransferSwitch.connection[1].presentState has a value of "disconnected," report:
 1. the results of the Aggregation for the Circuit connected to GridTransferSwitch.connection[2] with all associated MeasurementQuantity.values, including those indirectly associated as part of a MeasurementSet, set to 0 and all associated MeasurementQuantity.uncertainty values, including those indirectly associated as part of a MeasurementSet, set to 0 as the contribution of the GridTransferSwitch to the Aggregation for the Circuit connected to GridTransferSwitch.connection[1], and with
 2. AggregationProperties.hasElectricalGenerators set to False, and AggregationProperties.hasLoads set to False as the contribution of the GridTransferSwitch to the AggregationProperties for the Circuit connected to GridTransferSwitch.connection[1].

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\GridTransferSwitchAggregationRuleSet*. See Clause 5.6.7.14.5 for the complete definition of this element.

Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*. See Clause 6.6.4.1 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

RouterConnectionPoint (Class)

This class is used to model the input/output characteristics of connections from an EnergyRouter to a source or sink of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\RouterConnectionPoint*. See Clause 5.6.7.14.6 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

TransferSwitch (Class)

This class is used to model transfer switches, which are devices that allow a load or set of loads to be switched between two power sources without allowing the two power sources to be connected together.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\TransferSwitch*. See Clause 5.6.7.14.7 for the complete definition of this element.

TransferSwitchAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a TransferSwitch, calculate the Aggregation for the Circuit connected to TransferSwitch.connection[1] and TransferSwitch.connection[2] separately and then execute the following steps:
 1. Set AggregationProperties.hasLoads for the Circuit connected to TransferSwitch.connection[0] = True;
 2. Set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] = False;
 3. If TransferSwitch.connection[0].presentState is "input" and AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[2] is True, set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] to True;
 4. If TransferSwitch.connection[1].presentState is "input" and AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[1] is True, set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] to True, and;
 5. If AggregationProperties.hasLoads for the Circuit connected to TransferSwitch.connection[1] is True and AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[1] is True, set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] to True.
- Use the resulting aggregateQuantity, hasLoads flag, and hasElectricalGenerators flag for the resulting contributions of the TransferSwitch to the Aggregation and AggregationProperties for the Circuit connected to TransferSwitch.connection[0].

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\TransferSwitchAggregationRuleSet*. See Clause 5.6.7.14.8 for the complete definition of this element.

UnidirectionalCombiner (Class)

This class is used to model unidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that does not allow energy to flow from one source to another. The effect of the constraints that are placed on this class results in the following behavior for the unidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of the combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.
- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of the combiner box is zero and the amount of power actually injected into the onsite generation side of the combiner box is equal to the amount of power drawn from load side of the combiner box.

UML Element Location: *Model\Device and Model Components\Energy Manager*

Component\EnergyRouter\UnidirectionalCombiner. See Clause 5.6.7.14.9 for the complete definition of this element.

UnidirectionalCombinerAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a UnidirectionalCombiner, calculate the Aggregation for the Circuit connected to UnidirectionalCombiner.connection[1] and UnidirectionalCombiner.connection[2] separately and then:
 1. take the logical OR of AggregationProperties.hasElectricalGenerators for the Circuit connected to UnidirectionalCombiner.connection[1] and AggregationProperties.hasElectricalGenerators for the Circuit connected to UnidirectionalCombiner.connection[2];
 2. take the logical OR of AggregationProperties.hasLoads for the Circuit connected to UnidirectionalCombiner.connection[1] and AggregationProperties.hasLoads for the Circuit connected to UnidirectionalCombiner.connection[2] and;
 3. sum the Aggregation.aggregateQuantity for the Circuit connected to UnidirectionalCombiner.connection[1] and the Aggregation.aggregateQuantity for the Circuit connected to UnidirectionalCombiner.connection[2], to yield the resulting contributions of the UnidirectionalCombiner to the Aggregation for the Circuit connected to UnidirectionalCombiner.connection[0].

UML Element Location: *Model\Device and Model Components\Energy Manager*

Component\EnergyRouter\UnidirectionalCombinerAggregationRuleSet. See Clause 5.6.7.14.10 for the complete definition of this element.

5.6.7. Elements Defined in the Energy Manager Component Model

The following elements are defined in this model.

5.6.7.1. DemandTarget (Class)

This class defines a demand target during a time interval.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\DemandTarget*.

Table 5.29 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
applicableTimePeriod	The time period during which a demand target applies.	AvailabilityType See Clause 5.7.3.2.5.	[0..1]
peakDemandTarget	The targeted peak demand not to be exceeded during the applicableTimePeriod.	PowerRealType See Clause 6.6.4.6.16.	[1]

5.6.7.2. EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

Parent Class: ComponentElement (See Clause 5.2.3.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*.

This element has the following connections to other elements:

Table 5.30 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	billingCycle	Sequences of actual archived historical interval data that are synchronized to the facility's billing cycle.	Sequence See Clause 5.6.7.10.4.	[0..*]
Association	eMs	Energy Managers managed by this Energy Manager.	EM See Clause 5.6.7.2.	[0..*]
Composition	energyRouter	The energy router(s) managed by this Energy Manager.	EnergyRouter See Clause 5.6.7.14.3.	[0..*]
Association	forecasts	Sequences of projections of interval data.	ForecastSequence See Clause 5.6.7.10.2.	[0..*]
Association	generators	Generators managed by this Energy Manager.	Generator See Clause 5.5.6.2.	[0..*]
Association	loads	Loads managed by this Energy Manager.	Load See Clause 5.4.2.5.	[0..*]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	meters	Meters monitored by this Energy Manager.	Meter See Clause 5.3.2.1.	[0..*]
Association	presentAggregationData	The set of data associated with the present (filling) interval. Note: All direct subordinate Meters, Loads, Generators, and Energy Managers have one or more ConnectionPoints that link them to a Circuit. There is one set of EMPresentData for each of these Circuits.	EMPresentData See Clause 5.6.7.7.	[1..*]
Association	userSequence	Used defined Sequences of Intervals used by this Energy Manager.	Sequence See Clause 5.6.7.10.4.	[0..*]

5.6.7.3. EMGenerationType (Class)

This class represents net demand reductions resulting from supply increases. The class supports power ramps that describe the power profile for a generation resource when it is energized (stagingRamps) or de-energized (recoveryRamps).

For each array element of a generation stagingRamp, the stagingRamp[n].rate defines the amount of power increase and the stagingRamp[n].duration defines the length of time in seconds upon which this increase is in effect. If the stagingRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the generation stagingRamp attribute name implies that the demand is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

For each array element of a generator recoveryRamp, the recoveryRamp[n].rate defines the amount of power decrease and the recoveryRamp[n].duration defines the length of time in seconds upon which this decrease is in effect. If the recoveryRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the generator recoveryRamp attribute name implies that the demand reduction is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

Parent Class: GenerationType (See Clause 5.7.2.13.6.2.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMGenerationType.*

This element has the following connections to other elements:

Table 5.31 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	actualIntervalSupplyChangeBasis	This value identifies the affected intervals upon which the grid commitment was achieved.	EMIntervalData See Clause 5.6.7.4.	[0..*]
Association	actualPresentSupplyChangeBasis	This value provides the technical data provenance upon which the grid commitment was achieved.	EMPresentData See Clause 5.6.7.7.	[1]
Association	committedIntervalSupplyChangeBasis	This value identifies the affected intervals upon which the grid commitment is made.	EMIntervalData See Clause 5.6.7.4.	[0..*]
Association	committedPresentSupplyChangeBasis	This value provides the technical data provenance upon which the grid commitment was made.	EMPresentData See Clause 5.6.7.7.	[1]
Association	generatorEmissions	Emissions associated with operating a generator.	EmissionsMeasurement sSet See Clause 6.6.4.8.12.	[0..1]
Association	measuredAt	Points in the facility power network at which measurements of generation are made.	EMUsagePoint See Clause 5.6.7.8.	[1..*]

5.6.7.4. EMIntervalData (Class)

This class represents data in the domain of concern of an Energy Manager over some complete interval of time. This time interval may be in the past or future.

Parent Class: AttachType (See Clause 5.7.3.2.4.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMIntervalData.*

Table 5.32 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
aggregateAdjustedFullDRDemand	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate demand calculated by applying the AdjustedFullDRDemandRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads, with the largest acceptable demand response, directly or indirectly managed by the associated EM, averaged over the identified time interval.	AdjustedFullDRDemandAggregation See Clause 5.6.7.11.6.1.	[0..1]
aggregateAdjustedFullDRSupply	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate supply calculated by applying the AdjustedFullDRSupplyRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate	AdjustedFullDRSupplyAggregation See Clause 5.6.7.11.6.3.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
aggregateAdjustedNoDRDemand	supply of generators and electrical storage devices, with the largest acceptable demand response, directly or indirectly managed by the associated EM, averaged over the identified time interval. For an interval that is not part of a ForecastSequence, this attribute represents the aggregate demand calculated by applying the AdjustedNoDRDemandRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads, as if no demand response was in effect, directly or indirectly managed by the associated EM, averaged over the identified time interval.	AdjustedNoDRDemand Aggregation See Clause 5.6.7.11.6.5.	[0..1]
aggregateAdjustedNoDRSupply	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate supply calculated by applying the AdjustedNoDRSupplyRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of generators and electrical storage devices, as if no demand response was in effect, directly or indirectly managed by the associated EM, averaged over the identified time interval.	AdjustedNoDRSupply Aggregation See Clause 5.6.7.11.6.7.	[0..1]
aggregateDemand	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate demand calculated by applying the DemandRuleset algorithm averaged over the interval referenced by the IntervalDataContainer. Alternatively, this could be determined by calculating EnergyConsumedAggregation divided by the interval duration. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads directly or indirectly managed by the associated EM, averaged over the identified time interval.	DemandAggregation See Clause 5.6.7.11.6.9.	[0..1]
aggregateElectricalEnergyStored	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate stored electrical energy calculated by applying the ElectricalEnergyStoredRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate electrical energy stored by electrical storage devices directly or indirectly managed by the associated EM, during the identified time interval.	ElectricalEnergyStored Aggregation See Clause 5.6.7.11.6.11.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
aggregateEmissionsGenerated	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate generated emissions calculated by applying the EmissionsGeneratedRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate emissions produced by generators and electrical storage devices directly or indirectly managed by the associated EM, during the identified time interval.	EmissionsGeneratedAggregation See Clause 5.6.7.11.6.13.	[0..1]
aggregateEmissionsGenerationRate	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate emissions generation rate calculated by applying the EmissionsGenerationRateRuleset algorithm averaged over the interval referenced by the IntervalDataContainer. Alternatively, this could be determined by calculating EmissionsGeneratedAggregation/ divided by the interval duration. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate emissions generation rate of generators and electrical storage devices directly or indirectly managed by the associated EM, averaged over the identified time interval.	EmissionsGenerationRateAggregation See Clause 5.6.7.11.6.14.	[0..1]
aggregateEnergyConsumed	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate consumed energy calculated by applying the EnergyConsumedRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate energy consumed by loads directly or indirectly managed by the associated EM, during the identified time interval.	EnergyConsumedAggregation See Clause 5.6.7.11.6.17.	[0..1]
aggregateEnergySupplied	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate supplied energy calculated by applying the EnergySuppliedRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate energy supplied by generators and electrical storage devices directly or indirectly managed by the associated EM, during the identified time interval.	EnergySuppliedAggregation See Clause 5.6.7.11.6.19.	[0..1]
aggregateNetDemand	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate net demand calculated by applying the NetDemandRuleset algorithm averaged over the interval referenced by the IntervalDataContainer. Alternatively, this could be determined by calculating NetEnergyConsumedAggregation divided by the interval duration.	NetDemandAggregation See Clause 5.6.7.11.6.21.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
aggregateNetEnergyConsumed	For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate net demand of loads, generators, and electrical storage devices directly or indirectly managed by the associated EM, averaged over the identified time interval. For an interval that is not part of a ForecastSequence, this attribute represents the aggregate net consumed energy calculated by applying the NetEnergyConsumedRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate net energy consumption of loads, generators, and electrical storage devices directly or indirectly managed by the associated EM, during the identified time interval.	NetEnergyConsumedA ggregation See Clause 5.6.7.11.6.23.	[0..1]
aggregateSupply	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate supply calculated by applying the SupplyRuleset algorithm averaged over the interval referenced by the IntervalDataContainer. Alternatively, this could be determined by calculating EnergySuppliedAggregation divided by the interval duration. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate supply of generators and electrical storage devices directly or indirectly managed by the associated EM, averaged over the identified time interval.	Supply/Aggregation See Clause 5.6.7.11.6.25.	[0..1]
aggregateThermalEnergyStored	For an interval that is not part of a ForecastSequence, this attribute represents the aggregate stored thermal energy calculated by applying the ThermalEnergyStoredRuleset algorithm at the end of the interval referenced by the IntervalDataContainer. For an interval that is part of a ForecastSequence, the exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate thermal energy stored by thermal storage devices directly or indirectly managed by the associated EM, during the identified time interval.	ThermalEnergyStoredA ggregation See Clause 5.6.7.11.6.27.	[0..1]
resources	The set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager during the interval referenced by the IntervalDataContainer through the attach relation.	AllResourcesInEMDo main See Clause 5.6.7.11.2.	[0..1]

This element has the following connections to other elements:

Table 5.33 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	actualIntervalDemandChange	This attribute represents the demand profile change in response to a request from the grid. The process for calculating this is a local matter.	EMLoadReductionType See Clause 5.6.7.5.	[0..*]
Association	actualIntervalNetDemandChange	This attribute represents the net demand profile change in response to a request from the grid. The process for calculating this is a local matter.	EMPowerResponseType See Clause 5.6.7.6.	[0..*]
Association	actualIntervalSupplyChange	This attribute represents the actual supply profile change in response to a request from the grid. The process for calculating this is a local matter.	EMGenerationType See Clause 5.6.7.3.	[0..*]
Association	aggregationMetadata	The metadata that applies to a Circuit for which an aggregation is being computed.	AggregationProperties See Clause 5.6.7.13.1.	[1]
Association	committedIntervalDemandChange	This represents the demand profile change commitment made to the energy provider in response to a curtailment request. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMLoadReductionType See Clause 5.6.7.5.	[0..*]
Association	committedIntervalNetDemandChange	This represents the net demand profile change commitment made to the energy provider in response to a curtailment request. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMPowerResponseType See Clause 5.6.7.6.	[0..*]
Association	committedIntervalSupplyChange	This represents the supply profile change commitment made to the energy provider in response to a dispatch request. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMGenerationType See Clause 5.6.7.3.	[0..*]
Association	eiEvents	Energy events of interest to this energy manager for this interval.	EiEventType See Clause 5.7.1.2.1.8.	[0..*]
Association	energySuppliers	Organizations available to the EM for supplying energy during the interval. An EMIntervalData instance for an ESI energy manager will have at least one energySupplier.	ServiceSupplierData See Clause 5.6.7.9.	[0..*]
Association	intervalMeasurementQuantity	User defined MeasurementQuantity relevant to the EM for the interval.	MeasurementQuantity See Clause 6.6.4.5.	[0..*]
Association	intervalMeasurementSet	User defined MeasurementSet relevant to the EM for the interval.	MeasurementSet See Clause 6.6.4.2.	[0..*]
Association	intervalPowerQualitySummary	Power quality summary associated with the Interval.	ElectricPowerQualitySummary See Clause 5.7.5.2.1.5.	[0..*]
Association	intervalWeatherReportMeasured	The weather associated with the interval. This weather report is provided by a source internal to the facility	FSGIMWeatherReport See Clause 5.6.7.12.4.	[0..*]
Association	intervalWeatherReportProvided	The weather associated with the interval. This weather report is provided	FSGIMWeatherReport See Clause 5.6.7.12.4.	[0..*]

Connection Type	Role Name	Description	Role Type	Multiplicity
	ded	by an external source.	See Clause 5.6.7.12.4.	
Association	intervalWeatherTrendMeasured	The weather trend associated with the interval. This weather trend is provided by a source internal to the facility.	FSGIMWeatherTrend See Clause 5.6.7.12.5.	[0..1]
Association	intervalWeatherTrendProvided	The weather trend associated with the interval. This weather trend is provided by an external source.	FSGIMWeatherTrend See Clause 5.6.7.12.5.	[0..1]
Association	prices	Price information for energy consumed during the interval.	EmixBaseType See Clause 5.7.2.13.1.2.	[0..*]
Association	usageSummary	Summary data related to energy for the interval(s). Note that this UsageSummary is distinct from any UsageSummary generated by the supplier.	UsageSummary See Clause 5.7.5.2.1.27.	[0..1]

5.6.7.5. EMLoadReductionType (Class)

This class represents demand reductions from Loads in response to a demand response event. The class supports power ramps that describe the power profile for a resource when it is de-energized (stagingRamps) or energized (recoveryRamps).

For each array element of a stagingRamp, the stagingRamp[n].rate defines the amount of power decrease and the stagingRamp[n].duration defines the length of time in seconds upon which this decrease is in effect. If the stagingRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the stagingRamp attribute name implies that the demand reduction is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be decreasing.

For each array element of a recoveryRamp, the recoveryRamp[n].rate defines the amount of power increase and the recoveryRamp[n].duration defines the length of time in seconds upon which this increase is in effect. If the recoveryRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the recoveryRamp attribute name implies that the demand is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

Parent Class: LoadReductionType (See Clause 5.7.2.13.6.3.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMLoadReductionType*.

This element has the following connections to other elements:

Table 5.34 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	actualIntervalDemandChangeBasis	This value identifies the affected intervals upon which the grid commitment was achieved.	EMIntervalData See Clause 5.6.7.4.	[0..*]
Association	actualIPresentDemandChangeBasis	This value provides the technical data provenance upon which the grid commitment was achieved.	EMPresentData See Clause 5.6.7.7.	[1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	committedIntervalDemandChangeBasis	This value identifies the affected intervals upon which the grid commitment is made.	EMIntervalData See Clause 5.6.7.4.	[0..*]
Association	committedPresentDemandChangeBasis	This value provides the technical data provenance upon which the grid commitment was made.	EMPresentData See Clause 5.6.7.7.	[1]
Association	measuredAt	Points in the facility power network at which measurements of usage are made.	EMUsagePoint See Clause 5.6.7.8.	[1..*]

5.6.7.6. EMPowerResponseType (Class)

This class represents net demand reductions resulting from a combination of reductions in demand (EMLoadReductionType) and increases in supply (EMGenerationType).

Parent Class: PowerResponseType (See Clause 5.7.2.13.6.7.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPowerResponseType*.

This element has the following connections to other elements:

Table 5.35 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	actualIntervalNetDemandChangeBasis	This value identifies the affected intervals upon which the grid commitment was achieved.	EMIntervalData See Clause 5.6.7.4.	[0..*]
Association	actualPresentNetDemandChangeBasis	This value provides the technical data provenance upon which the grid commitment was achieved.	EMPresentData See Clause 5.6.7.7.	[1]
Association	committedIntervalNetDemandChangeBasis	This value identifies the affected intervals upon which the grid commitment is made.	EMIntervalData See Clause 5.6.7.4.	[0..*]
Association	committedPresentNetDemandChangeBasis	This value provides the technical data provenance upon which the grid commitment was made.	EMPresentData See Clause 5.6.7.7.	[1]

5.6.7.7. EMPresentData (Class)

The container that directly holds all of the aggregated present data and indirectly holds the associated present context.

Constraint: If the attributes required by a standard aggregation's Ruleset are present, then that aggregation must be present.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPresentData*.

Table 5.36 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
partialIntervalAggregateElectricalEnergyStored	The aggregate stored electrical energy calculated by applying the ElectricalEnergyStoredAggregation algorithm in a partially completed (filling) demand interval.	ElectricalEnergyStoredAggregation See Clause 5.6.7.11.6.11.	[0..1]
partialIntervalAggregateEmissionsGenerated	The aggregate emissions calculated by applying the EmissionsGeneratedAggregation algorithm in a partially completed (filling) demand interval.	EmissionsGeneratedAggregation See Clause 5.6.7.11.6.13.	[0..1]
partialIntervalAggregateEnergyConsumed	The aggregate consumed energy calculated by applying the EnergyConsumedAggregation algorithm in a partially completed (filling) demand interval.	EnergyConsumedAggregation See Clause 5.6.7.11.6.17.	[0..1]
partialIntervalAggregateEnergySupplied	The aggregate generated energy calculated by applying the EnergySuppliedAggregation algorithm in a partially completed (filling) demand interval.	EnergySuppliedAggregation See Clause 5.6.7.11.6.19.	[0..1]
partialIntervalAggregateNetEnergyConsumed	The net consumed energy calculated by applying the NetEnergyConsumedAggregation algorithm in a partially completed (filling) demand interval.	NetEnergyConsumedAggregation See Clause 5.6.7.11.6.23.	[0..1]
partialIntervalAggregateThermalEnergyStored	The aggregate stored thermal energy calculated by applying the ThermalEnergyStoredAggregation algorithm in a partially completed (filling) demand interval.	ThermalEnergyStoredAggregation See Clause 5.6.7.11.6.27.	[0..1]
presentAggregateAdjustedFullDRDemand	The aggregate demand calculated by applying the AdjustedFullDRDemandAggregation algorithm averaged over the time since the beginning of the present demand interval.	AdjustedFullDRDemandAggregation See Clause 5.6.7.11.6.1.	[0..1]
presentAggregateAdjustedFullDRSupply	The aggregate supply calculated by applying the AdjustedFullDRSupplyAggregation algorithm averaged over the time since the beginning of the present demand interval.	AdjustedFullDRSupplyAggregation See Clause 5.6.7.11.6.3.	[0..1]
presentAggregateAdjustedNoDRDemand	The aggregate demand calculated by applying the AdjustedNoDRDemandAggregation algorithm averaged over the time since the beginning of the present demand interval.	AdjustedNoDRDemandAggregation See Clause 5.6.7.11.6.5.	[0..1]
presentAggregateAdjustedNoDRSupply	The aggregate supply calculated by applying the AdjustedNoDRSupplyAggregation algorithm averaged over the time since the beginning of the present demand interval.	AdjustedNoDRSupplyAggregation See Clause 5.6.7.11.6.7.	[0..1]
presentAggregateDemand	The aggregate demand calculated by applying the DemandAggregation algorithm at the present time.	DemandAggregation See Clause 5.6.7.11.6.9.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
presentAggregateEmissionsGenerationRate	The aggregate emissions generation rate calculated by applying the EmissionsGenerationRateAggregation algorithm at the present time.	EmissionsGenerationRateAggregation See Clause 5.6.7.11.6.14.	[0..1]
presentAggregateNetDemand	The net demand calculated by applying the NetDemandAggregation algorithm at the present time.	NetDemandAggregation See Clause 5.6.7.11.6.21.	[0..1]
presentAggregateSupply	The aggregate supply calculated by applying the SupplyAggregation algorithm at the present time.	SupplyAggregation See Clause 5.6.7.11.6.25.	[0..1]
presentResources (Derived attribute)	The set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager at this instant in time.	AllResourcesInEMDomain See Clause 5.6.7.11.2.	[1]

This element has the following connections to other elements:

Table 5.37 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	actualPresentDemandChange	This represents the actual demand profile change in response to a signal from the grid. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMLoadReductionType See Clause 5.6.7.5.	[0..*]
Association	actualPresentNetDemandChange	This represents the actual net demand profile change in response to a signal from the grid. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMPowerResponseType See Clause 5.6.7.6.	[0..*]
Association	actualPresentSupplyChange	This represents the actual supply profile change in response to a signal from the grid. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMGenerationType See Clause 5.6.7.3.	[0..*]
Association	aggregationMetadata	The metadata that applies to a Circuit for which an aggregation is being computed.	AggregationProperties See Clause 5.6.7.13.1.	[1]
Association	committedPresentDemandChange	This represents the demand profile change commitment made to the energy provider in response to a signal from the grid. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMLoadReductionType See Clause 5.6.7.5.	[0..*]
Association	committedPresentNetDemandChange	This represents the net demand profile change commitment made to the energy provider in response to a signal from the grid. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMPowerResponseType See Clause 5.6.7.6.	[0..*]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	committedPresentSupplyChange	This represents the supply profile change commitment made to the energy provider in response to a signal from the grid. The value is based upon the applicable technical aggregations and enterprise policy decisions.	EMGenerationType See Clause 5.6.7.3.	[0..*]
Association	currentEnergySupplier	Organization currently available to the EM for supplying energy.	ServiceSupplierData See Clause 5.6.7.9.	[0..1]
Association	currentPrices	Current energy price information known by the EM at the present time.	EmixBaseType See Clause 5.7.2.13.1.2.	[0..*]
Association	partialIntervalPowerQualitySummary	Power quality summary in a partially completed interval.	ElectricPowerQualitySummary See Clause 5.7.5.2.1.5.	[0..1]
Association	presentEIEvents	Energy events underway that are relevant to the EM at the present time.	EiEventType See Clause 5.7.1.2.1.8.	[0..*]
Association	presentMeasurementQuantity	User defined MeasurementQuantity relevant to the EM at the present time.	MeasurementQuantity See Clause 6.6.4.5.	[0..*]
Association	presentMeasurementSet	User defined MeasurementSet relevant to the EM at the present time.	MeasurementSet See Clause 6.6.4.2.	[0..*]
Association	presentWeatherReportMeasured	The weather at the present time and place. This weather report is provided by a source internal to the facility.	FSGIMWeatherReport See Clause 5.6.7.12.4.	[0..*]
Association	presentWeatherReportProvided	The weather at the present time and place. This weather report is provided by an external source.	FSGIMWeatherReport See Clause 5.6.7.12.4.	[0..*]
Association	presentWeatherTrendMeasured	The weather trend at the present time and place. This weather trend is provided by a source internal to the facility.	FSGIMWeatherTrend See Clause 5.6.7.12.5.	[0..1]
Association	presentWeatherTrendProvided	The weather trend at the present time and place. This weather trend is provided by an external source.	FSGIMWeatherTrend See Clause 5.6.7.12.5.	[0..1]
Association	usageSummary	The facility's Summary data related to energy for the present time. Note that this UsageSummary is distinct from any UsageSummary generated by the supplier.	UsageSummary See Clause 5.7.5.2.1.27.	[0..1]
Association	userAggregations	User defined aggregations used by this Energy Manager.	Aggregation See Clause 5.6.7.11.1.	[0..*]

5.6.7.8. EMUsagePoint (Class)

Logical point on a network at which consumption or production is either measured (e.g., metered) or estimated (e.g., unmetered street lights).

Constraint: For the FSGIM, the associations from the parent UsagePoint Class to the other NAESB classes all have a multiplicity of 0 with the exception of ElectricPowerQualitySummary because this information is captured in other ways in the FSGIM model.

Parent Class: UsagePoint (See Clause 5.7.5.2.1.26.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMUsagePoint*.

This element has the following connections to other elements:

Table 5.38 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	demandSources	Sources of demand contributing to values measured at an EMUsagePoint.	EMLoadReductionType See Clause 5.6.7.5.	[0..*]
Association	emixInterfaceType	Identifies the corresponding EmixInterfaceType for external EMUsagePoints.	EmixInterfaceType See Clause 5.7.2.13.1.3.	[0..1]
Association	localTimeParameters	Defines the local time conversions from a UTC timestamp.	LocalTimeParameters See Clause 6.6.1.5.	[0..1]
Association	measurements	Measurements at a specific EMUsagePoint.	Measurement See Clause 6.6.4.1.	[0..*]
Association	physicalConnectionPoint	The representation of the point that connects to a Circuit in the facility's physical electrical topology.	ConnectionPoint See Clause 6.6.5.2.	[0..1]
Association	supplySources	Sources of supply contributing to values measured at an EMUsagePoint.	EMGenerationType See Clause 5.6.7.3.	[0..*]

5.6.7.9. ServiceSupplierData (Class)

This class contains information associated with the generated emissions, quality of power, and energy-price units associated with an identified supplier of energy.

Parent Class: FSGIMIdentifiedObject (See Clause 5.2.3.4.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\ServiceSupplierData*.

Table 5.39 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
cpp	The critical peak period bucket to which the reading is attributed. A value of 0 means that there is no applicable critical peak period. Even though critical peak period is usually considered a specialized form of time-of-use pricing, this attribute is defined explicitly to provide flexibility.	Integer See Clause 6.6.3.5.	[1]
emissions	The emissions generated by this supplier for the energy used.	EmissionsMeasurementsSet See Clause 6.6.4.8.12.	[0..1]
emissionsRate	The rate of emissions generated by this supplier for the energy used.	EmissionsRateMeasurementsSet	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
energyPrice	The unit energy price associated with this supplier for the energy used.	See Clause 6.6.4.8.15. UnitEnergyPriceType See Clause 5.7.2.13.4.8.	[0..1]
marketContext	The overall market or program rules that govern relations between the energy supplier and the facility.	AnyURI See Clause 6.6.3.2.	[0..*]
tou	The time-of-use bucket to which the reading value is attributed. A value of 0 means not applicable.	Integer See Clause 6.6.3.5.	[1]

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

This element has the following connections to other elements:

Table 5.40 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	energySupplier	The supplier of the energy referenced by this data.	ServiceSupplier See Clause 5.7.5.2.1.23.	[1]
Association	supplierUsageSummary	The supplier's summary for this energySupplierData. Note that this UsageSummary is distinct from any UsageSummary generated by the facility.	UsageSummary See Clause 5.7.5.2.1.27.	[0..1]
Association	target	The internal demand goal or target for this energy supplier.	DemandTarget See Clause 5.6.7.1.	[0..*]

5.6.7.10. Elements Defined in the Sequences and Intervals Model

The following elements are defined in this model.

5.6.7.10.1. PeakPowerData (Class)

This class contains information associated with peak power intervals within a Sequence.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\PeakPowerData*.

Table 5.41 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
demandPrice	This attribute represents the demand price associated with a billingCycle.	MonetaryQuantity See Clause 6.6.5.4.	[0..1]
supplyPrice	This attribute represents the supply price associated with a billingCycle. This is analogous to the concept of a demand price, for facilities that have local generation.	MonetaryQuantity See Clause 6.6.5.4.	[0..1]
ratchetPrice	This attribute represents the ratchet price associated with a Sequence applicable to the ratchet period.	MonetaryQuantity See Clause 6.6.5.4.	[0..1]

This element has the following connections to other elements:

Table 5.42 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	peakNetDemand	This value is only applicable to the EM.billingCycle sequence of an energy manager that is serving in an ESI EM role.	EMIntervalData See Clause 5.6.7.4.	[0..1]
		The value identifies the EMIntervalData instance containing the most positive EMIntervalData.aggregateNetDemand within the sequence of		

Connection Type	Role Name	Description	Role Type	Multiplicity
		intervals. The applicable tariff rules are used to determine which instance of EMIntervalData represents this peak net demand. If the information is available and there is at least one instance that meets the tariff rules and where the EMIntervalData.aggregateNetDemand is positive, then this information must be provided.		
Association	peakNetSupply	This value is only applicable to the EM.billingCycle sequence of an energy manager that is serving in an ESI EM role. The value identifies the EMIntervalData instance containing the most negative EMIntervalData.aggregateNetDemand within the sequence of intervals. The applicable tariff rules are used to determine which instance of EMIntervalData represents this peak net supply. If the information is available and there is at least one instance that meets the tariff rules and where the EMIntervalData.aggregateNetDemand is negative, then this information must be provided.	EMIntervalData See Clause 5.6.7.4.	[0..1]

5.6.7.10.2. ForecastSequence (Class)

The Sequence that is used for storing all forecasts, including forecasts for time periods where the originally forecasted time period has now passed.

Parent Class: Sequence (See Clause 5.6.7.10.4.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\ForecastSequence.*

Table 5.43 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
timeOffForecast	Time at which the forecast was made.	UTCDateTime See Clause 6.6.1.11.	[0..1]

5.6.7.10.3. IntervalDataContainer (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval. This is a specialization of the WSCalendar PIM IntervalType.

Constraint: At any time, at most two of the attributes (dtStart, dtEnd, duration) may have a value.

Constraint: Only dtStart, dtEnd, duration may be specified.

Constraint: When the 'duration' attribute is defined, the duration must be greater than zero.

Constraint: When the dtStart and dtEnd attributes are defined, the dtStart attribute must precede the dtEnd attribute in time.

Parent Class: IntervalType (See Clause 5.7.3.2.7.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\IntervalDataContainer*.
This element has the following connections to other elements:

Table 5.44 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregations	Aggregations for the Interval.	Aggregation See Clause 5.6.7.11.1.	[0..*]
Association	attach	Associates a set of EMIntervalData with this IntervalDataContainer. Note: All direct subordinate Meters, Loads, Generators, and Energy Managers have one of more ConnectionPoints that link them to a Circuit. For each instance of the Sequence Class or ForecastSequence Class, there is one instance of the EMIntervalData Class for each of these Circuits. (This association redefines the attach association between the IntervalType Class (see Clause 5.7.3.2.7) and the AttachType Class (see Clause 5.7.3.2.4).)	EMIntervalData See Clause 5.6.7.4.	[0..*]
Association	localTimeParameters	Defines the local time conversions from a UTC timestamp.	LocalTimeParameters See Clause 6.6.1.5.	[0..1]

5.6.7.10.4. Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be summarised by monitoring the time references.

Parent Class: GluonType (See Clause 5.7.3.2.6.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*.

Table 5.45 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
intervalDuration (Derived attribute)	This attribute specifies the duration of individual intervals in the common case of a sequence of contiguous intervals, all having the same duration. This attribute is derived from the path <i>vailability.availability.interval.duration</i> . Constraint: If the duration of the individual intervals is not the common case of a sequence of contiguous intervals, all having the same duration, then the multiplicity of	Duration See Clause 6.6.1.9.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
name	this attribute shall be 0. Name used to identify the sequence.	String See Clause 6.6.3.7.	[1]
timeOfLastSync	This attribute represents the most recent time that IntervalDataContainer.timeReference was synchronized with the energy provider. Synchronization is sometimes implemented using a demand end-of-interval (EOI) pulse obtained from the meter. To determine the next expected EOI this attribute is summed with the Sequence.intervalDuration attribute.	UTCDateTime See Clause 6.6.1.11.	[0..1]

This element has the following connections to other elements:

Table 5.46 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	intervals	Time related Intervals that are part of this Sequence.	IntervalDataContainer See Clause 5.6.7.10.3.	[1..*]
Association	localTimeParameters	Defines the local time conversions from a UTC timestamp.	LocalTimeParameters See Clause 6.6.1.5.	[0..1]
Association	peakPower	This association is only applicable to an energy manager that is serving in an ESI EM role. It identifies information associated with any peaks during the given Sequence.	PeakPowerData See Clause 5.6.7.10.1.	[0..1]

5.6.7.11. Elements Defined in the Collections, Rule Sets, and Aggregations Model

The following elements are defined in this model.

5.6.7.11.1. Aggregation (Class)

An aggregation allows the assembly of statistics of interest arrived at through defined algorithms applied to other parts of the model. The description of an aggregation includes all of the inputs that comprise the algorithms as well as the outputs of the algorithm. An aggregation instance represents one or more aggregateQuantities that are computed by applying aggregation rules for each quantity against a Collection or FilteredCollection of Components.

Parent Class: FSGIMIdentifiedObject (See Clause 5.2.3.4.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Aggregation.*

This element has the following connections to other elements:

Table 5.47 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The resulting measurement quantity values for an Aggregation.	Measurement See Clause 6.6.4.1.	[1]
Association	aggregationRuleset	The ruleset that defines how an AggregateQuantity is derived from a collection.	Ruleset See Clause 5.6.7.11.5.	[1]
Association	collection	The collection used as a source for deriving aggregate values.	Collection See Clause 5.6.7.11.3.	[0..*]

5.6.7.11.2. AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

Parent Class: Collection (See Clause 5.6.7.11.3.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain.*

5.6.7.11.3. Collection (Class)

This class defines a set of entities that is identified as a group because the members share some trait. A Collection can contain other Collections.

Parent Class: ComponentElement (See Clause 5.2.3.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Collection.*

This element has the following connections to other elements:

Table 5.48 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Aggregation	part	The component elements that make up a collection.	ComponentElement See Clause 5.2.3.1.	[0..*]

5.6.7.11.4. FilteredCollection (Class)

A FilteredCollection "filters" one or more inputCollections according to a set of filteringRules to produce a new instance of a Collection with the subset defined by the Ruleset.

Parent Class: Collection (See Clause 5.6.7.11.3.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\FilteredCollection*.
This element has the following connections to other elements:

Table 5.49 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	filteringRuleSet	The ruleset that defines how to derive the members of a filtered collection from the members of the input collections.	RuleSet See Clause 5.6.7.11.5.	[1]
Association	inputCollection	The collections from which a filtered collection's members are derived.	Collection See Clause 5.6.7.11.3.	[1..*]
Derived Aggregation	ruleBasedPart	The parts of a FilteredCollection that are derived from its inputCollections according to the filteringRuleSet.	ComponentElement See Clause 5.2.3.1.	[0..*]

5.6.7.11.5. RuleSet (Class)

A RuleSet contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in RuleSet.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

Parent Class: FSGIMIdentifiedObject (See Clause 5.2.3.4.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*.

Table 5.50 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
rules	The rules attribute contains a set of strings that specifies the rules contained in the ruleset. These rules should be interpreted in the order listed. In the standards document, the rules are shown in the Ruleset subclass notes.	String See Clause 6.6.3.7.	[0..*]

5.6.7.11.6. Elements Defined in the Standard Aggregations Model

The following elements are defined in this model.

5.6.7.11.6.1. AdjustedFullDRDemandAggregation (Class)

For historical aggregations, this is the aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval assuming the largest acceptable response to demand response events occurred.

For a partially completed (filling) demand interval, this is the expected demand of loads directly or indirectly managed by the associated EM as if the largest acceptable response to a demand response event was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads, with the largest acceptable demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

For example, in a production line this could be the demand of the production line when it was providing the largest acceptable demand reduction in response to a DR request. For an HV AC system, this could be the demand that still meets code requirements and acceptable comfort conditions assuming a certain set of weather conditions and occupancy levels.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRDemandAggregation.*

This element has the following connections to other elements:

Table 5.51 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The PowerMeasurementsSet resulting from an AdjustedFullDRDemandAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
Association	aggregationRuleset	The rules that define how an AdjustedFullDRDemandAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	AdjustedFullDRDemandRuleSet See Clause 5.6.7.11.6.2.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the AdjustedFullDRDemandAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.2. AdjustedFullDRDemandRuleSet (Class)

Calculate "full DR adjusted demand from loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) where Load.demandLimits.adjustedFullDRPower is available, sum Load.demandLimits.adjustedFullDRPower.

Calculate "full DR adjusted demand from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedFullDRDemand is available, sum the subordinate EMPresentData.presentAggregateAdjustedFullDRDemand.aggregateQuantity.

Calculate final result:

- Sum "full DR adjusted demand from loads" and "full DR adjusted demand from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRDemandRuleSet.*

5.6.7.11.6.3. AdjustedFullDRSupplyAggregation (Class)

For historical aggregations, this is the aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval assuming the largest acceptable response to demand response events occurred.

For a partially completed (filling) demand interval, this is the expected supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM as if the largest acceptable response to a demand response event was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of electrical generators and electrical storage devices, with the largest acceptable demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRSupplyAggregation.*

This element has the following connections to other elements:

Table 5.52 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The PowerMeasurementsSet resulting from an AdjustedFullDRSupplyAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
Association	aggregationRuleset	The rules that define how an AdjustedFullDRSupplyAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	AdjustedFullDRSupplyRuleSet See Clause 5.6.7.11.6.4.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the AdjustedFullDRSupplyAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the	AllResourcesInEMDomain main. See Clause 5.6.7.11.2.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
		Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3.)		

5.6.7.11.6.4. AdjustedFullDRSupplyRuleSet (Class)

Calculate "full DR adjusted supply from generators":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.supplyLimits.adjustedFullDRPower is available, sum Generator.supplyLimits.adjustedFullDRPower.

Calculate "full DR adjusted supply from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedFullDRSupply is available, sum the subordinate EMPresentData.presentAggregateAdjustedFullDRSupply.aggregateQuantity.

Calculate final result:

- Sum "full DR adjusted supply from generators" and "full DR adjusted supply from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Components\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedFullDRSupplyRuleSet.*

5.6.7.11.6.5. AdjustedNoDRDemandAggregation (Class)

For historical aggregations, this is the aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval assuming no demand response events occurred.

For a partially completed (filling) demand interval, this is the expected demand of loads directly or indirectly managed by the associated EM as if no demand response was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads, without demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

For example, in a production line this could be the expected average demand of the production line assuming that the production line was operating at a certain capacity. For an HVAC system, this could be the expected average demand assuming a certain set of weather conditions and occupancy levels.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRDemandAggregation.*

This element has the following connections to other elements:

Table 5.53 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The PowerMeasurementsSet resulting from an AdjustedNoDRDemandAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
Association	aggregationRuleset	The rules that define how an AdjustedNoDRDemandAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	AdjustedNoDRDemand RuleSet See Clause 5.6.7.11.6.6.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the AdjustedNoDRDemandAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDo main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.6. AdjustedNoDRDemandRuleSet (Class)

Calculate "no DR adjusted demand from loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) where Load.demandLimits.adjustedNoDRPower is available, sum Load.demandLimits.adjustedNoDRPower.

Calculate "no DR adjusted demand from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedNoDRDemand is available, sum the subordinate EMPresentData.presentAggregateAdjustedNoDRDemand.aggregateQuantity.

Calculate final result:

- Sum "no DR adjusted demand from loads" and "no DR adjusted demand from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRDemandRuleSet.*

5.6.7.11.6.7. AdjustedNoDRSupplyAggregation (Class)

For historical aggregations, this is the aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval assuming no demand response events occurred.

For a partially completed (filling) demand interval, this is the expected supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM as if no demand response was in effect, averaged over the time since the beginning of the present demand interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate supply of electrical generators and electrical storage devices, without demand response, directly or indirectly managed by the associated EM averaged over the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRSupplyAggregation.*

This element has the following connections to other elements:

Table 5.54 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The PowerMeasurementsSet resulting from an AdjustedNoDRSupplyAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
Association	aggregationRuleset	The rules that define how an AdjustedNoDRSupplyAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	AdjustedNoDRSupply RuleSet See Clause 5.6.7.11.6.8.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the AdjustedNoDRSupplyAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDo main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.8. AdjustedNoDRSupplyRuleSet (Class)

Calculate "no DR adjusted supply from generators":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.demandLimits.adjustedNoDRPower is available, sum Generator.demandLimits.adjustedNoDRPower.

Calculate "no DR adjusted supply from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.presentAggregateAdjustedNoDRSupply is available, sum the subordinate EMPresentData.presentAggregateAdjustedNoDRSupply.aggregateQuantity.

Calculate final result:

- Sum "no DR adjusted supply from generators" and "no DR adjusted supply from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\AdjustedNoDRSupplyRuleSet.*

5.6.7.11.6.9. DemandAggregation (Class)

For historical aggregations, this is the aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate present subinterval demand of loads directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate demand of loads directly or indirectly managed by the associated EM averaged over the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\DemandAggregation.*
This element has the following connections to other elements:

Table 5.55 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The PowerMeasurementsSet resulting from a DemandAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregationRuleSet	The rules that define how a DemandAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleSet association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	DemandRuleSet See Clause 5.6.7.11.6.10.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the DemandAggregation aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.10. DemandRuleSet (Class)

Calculate "metered demand":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Load Class (including instances of the CurtailableLoad Class) and/or
 2. direct subordinate instances of the EM Class where EM.presentAggregationData.aggregationMetadata.hasElectricalGenerators is False
are connected to its output ConnectionPoint and where ElectricMeter.powerReading is available, sum ElectricMeter.powerReading.

Calculate "demand from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered demand" and where Load.actualDemand is available, sum Load.actualDemand.

Calculate "demand from energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered demand" and where EMPresentData.presentAggregateDemand is available, sum the subordinate EMPresentData.presentAggregateDemand.aggregateQuantity.

Calculate final result:

- Sum "metered demand", "demand from unmetered loads", and "demand from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\DemandRuleSet.

5.6.7.11.6.11. ElectricalEnergyStoredAggregation (Class)

For historical or forecast aggregations, this is the energy stored in electrical storage directly or indirectly managed by the associated EM during the identified time interval. For a partially completed (filling) demand interval, this is the energy stored in electrical storage directly or indirectly managed by the associated EM since the beginning of the interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ElectricalEnergyStoredAggregation.*

This element has the following connections to other elements:

Table 5.56 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The EnergyMeasurementsSet resulting from an ElectricalEnergyStoredAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	EnergyMeasurementsSet See Clause 6.6.4.7.9.	[1]
Association	aggregationRuleset	The rules that define how an EnergyStoredAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	ElectricalEnergyStored RuleSet See Clause 5.6.7.11.6.12.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the ElectricalEnergyStoredAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.12. ElectricalEnergyStoredRuleSet (Class)

Calculate "stored electrical energy from storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.storage has the value "electricalStorage" and where Generator.actualStoredEnergy is available, sum Generator.actualStoredEnergy.

Calculate "stored electrical energy from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.partialIntervalAggregateElectricalEnergyStored is available, sum the subordinate EMPresentData.partialIntervalAggregateElectricalEnergyStored.aggregateQuantity.

Calculate final result:

- Sum "stored electrical energy from storage devices" and "stored electrical energy from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ElectricalEnergyStoredRuleSet.*

5.6.7.11.6.13. EmissionsGeneratedAggregation (Class)

For historical aggregations, this is the aggregate emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGeneratedAggregation.*

This element has the following connections to other elements:

Table 5.57 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The EmissionsMeasurementsSet resulting from an EmissionsGeneratedAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	EmissionsMeasurementsSet See Clause 6.6.4.8.12.	[1]
Association	aggregationRuleset	The rules that define how an EmissionsGeneratedAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between	EmissionsGeneratedRuleset See Clause	[1]

Connection Type	Role Name	Description	Role Type	Multiplicity
		the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	5.6.7.11.6.16.	
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the EmissionsGeneratedAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.14. EmissionsGenerationRateAggregation (Class)

For historical aggregations, this is the aggregate rate of emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) emissions interval, this is the aggregate present subinterval rate of emissions generated by electrical generators and electrical storage devices directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate rate of emissions generation of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGenerationRateAggregation.*

This element has the following connections to other elements:

Table 5.58 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The EmissionsRateMeasurementsSet resulting from an EmissionsGenerationRateAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	EmissionsRateMeasurementsSet See Clause 6.6.4.8.15.	[1]
Association	aggregationRuleset	The rules that define how an EmissionsGenerationRateAggregation aggregateQuantity is derived from the resources in an EM's domain.	EmissionsGenerationRateRuleSet	[1]

Connection Type	Role Name	Description	Role Type	Multiplicity
		(This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	See Clause 5.6.7.11.6.15.	
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the EmissionsGenerationRateAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.15. EmissionsGenerationRateRuleSet (Class)

Calculate "metered emissions rate":

- For each direct subordinate that is an instance of the EmissionsMeter Class where EmissionsMeter.emissionsRateReading is available, sum EmissionsMeter.emissionsRateReading.

Calculate "emissions rate from energy managers":

- For each direct subordinate that is an instance of the EM Class and where presentAggregateEmissionsGenerationRate is available, sum the subordinate presentAggregateEmissionsGenerationRate.aggregateQuantity.

Calculate final result:

- Sum "metered emissions rate" and "emissions rate from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGenerationRateRuleSet.*

5.6.7.11.6.16. EmissionsGeneratedRuleSet (Class)

Calculate "metered emissions":

- For each direct subordinate that is an instance of the EmissionsMeter Class where EmissionsMeter.emissionsReading is available, sum EmissionsMeter.emissionsReading.

Calculate "emissions from energy managers":

- For each direct subordinate that is an instance of the EM Class and where partialIntervalAggregateEmissionsGenerated is available, sum the subordinate partialIntervalAggregateEmissionsGenerated.aggregateQuantity.

Calculate final result:

- Sum "metered emissions" and "emissions from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EmissionsGeneratedRuleSet.*

5.6.7.11.6.17. EnergyConsumedAggregation (Class)

For historical aggregations, this is the aggregate energy consumed by loads directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate energy consumed by loads directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate energy consumed by loads directly or indirectly managed by the associated EM during the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergyConsumedAggregation.*

This element has the following connections to other elements:

Table 5.59 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The EnergyMeasurementsSet resulting from an EnergyConsumedAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	EnergyMeasurementsSet See Clause 6.6.4.7.9.	[1]
Association	aggregationRuleset	The rules that define how an EnergyConsumedAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	EnergyConsumedRuleSet See Clause 5.6.7.11.6.18.	[1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the EnergyConsumedAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.18. EnergyConsumedRuleSet (Class)

Calculate "metered consumption":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Load Class (including instances of the CurtailableLoad Class) and/or
 2. direct subordinate instances of the EM Class where EM.presentAggregationMetadata.hasElectricalGenerators is False

are connected to its output ConnectionPoint and where either

1. ElectricMeter.intervalEnergyReading or
2. the present ElectricMeter.accumulatedEnergyReading and the ElectricMeter.accumulatedEnergyReading at the beginning of the interval are available, either sum
 1. ElectricMeter.intervalEnergyReading or
 2. (the present ElectricMeter.accumulatedEnergyReading minus the ElectricMeter.accumulatedEnergyReading at the beginning of the interval) respectively.

Calculate "consumption from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered consumption" and where Load.actualDemand is available, sum (integral of the Load.actualDemand over the time since the beginning of the interval).

Calculate "consumption from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered consumption" and where EMPresentData.partialIntervalAggregateEnergyConsumed is available, sum the subordinate EMPresentData.partialIntervalAggregateEnergyConsumed.aggregateQuantity.

Calculate final result:

- Sum "metered consumption", "consumption from unmetered loads", and "consumption from unmetered energy managers".

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergyConsumedRuleSet.*

5.6.7.11.6.19. EnergySuppliedAggregation (Class)

For historical aggregations, this is the aggregate energy supplied by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate energy supplied by electrical generators and electrical storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate energy supplied by electrical generators and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergySuppliedAggregation.*

This element has the following connections to other elements:

Table 5.60 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The EnergyMeasurementsSet resulting from an EnergySuppliedAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	EnergyMeasurementsSet See Clause 6.6.4.7.9.	[1]
Association	aggregationRuleSet	The rules that define how an EnergySuppliedAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleSet association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	EnergySuppliedRuleSet See Clause 5.6.7.11.6.20.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the EnergySuppliedAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.20. EnergySuppliedRuleSet (Class)

Calculate "metered supplied energy":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Generator Class (including instances of the DispatchableGenerator Class) and/or
 2. direct subordinate instances of the EM Class where EM.presentAggregationData.aggregationMetadata.hasLoads is False are connected to its output ConnectionPoint and where either
 1. ElectricMeter.intervalEnergyReading or
 2. the present ElectricMeter.accumulatedEnergyReading and the ElectricMeter.accumulatedEnergyReading at the beginning of the interval are available, either sum
 1. $-1 * \text{ElectricMeter.intervalEnergyReading}$ or
 2. $-1 * (\text{the present ElectricMeter.accumulatedEnergyReading} - \text{ElectricMeter.accumulatedEnergyReading at the beginning of the interval})$ respectively.

Calculate "supplied energy from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered supplied energy" and where Generator.actualSupply is available, sum (integral of the Generator.actualSupply over the time since the beginning of the interval).

Calculate "supplied energy from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered supplied energy" and where EMPresentData.partialIntervalAggregateEnergySupplied is available, sum the subordinate EMPresentData.partialIntervalAggregateEnergySupplied.aggregateQuantity.

Calculate final result:

- Sum "metered supplied energy", "supplied energy from unmetered generators and electrical storage devices", and "supplied energy from unmetered energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\EnergySuppliedRuleSet.*

5.6.7.11.6.21. NetDemandAggregation (Class)

For historical aggregations, this is the aggregate net demand of loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate present subinterval net demand of loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate net demand of loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetDemandAggregation.*

This element has the following connections to other elements:

Table 5.61 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The PowerMeasurementsSet resulting from a NetDemandAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
Association	aggregationRuleset	The rules that define how a NetDemandAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	NetDemandRuleSet See Clause 5.6.7.11.6.22.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the NetDemandAggregation aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDo main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.22. NetDemandRuleSet (Class)

Calculate "metered net demand":

- For each direct subordinate that is an instance of the ElectricMeter Class, sum ElectricMeter.powerReadings.

Calculate "net demand from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtailableLoad Class) that is not included in "metered net demand" and where actualDemand is available, sum actualDemand.

Calculate "net demand from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered net demand" and where actualGeneration is available, sum $(-1 * \text{actualGeneration})$.

Calculate "net demand from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered net demand" and where presentAggregateNetDemand is available, sum the subordinate presentAggregateNetDemand.aggregateQuantity.

Calculate final result:

- Sum "metered net demand", "net demand from unmetered loads", "net demand from unmetered generators and electrical storage devices," and "net demand from unmetered energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetDemandRuleSet.*

5.6.7.11.6.23. NetEnergyConsumedAggregation (Class)

For historical aggregations, this is the aggregate net energy consumed by loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate net energy consumed by loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate net energy consumed by loads, electrical generators, and electrical storage devices directly or indirectly managed by the associated EM during the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetEnergyConsumedAggregation.*

This element has the following connections to other elements:

Table 5.62 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The EnergyMeasurementsSet resulting from a NetEnergyConsumedAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	EnergyMeasurementsSet See Clause 6.6.4.7.9.	[1]
Association	aggregationRuleset	The rules that define how a NetEnergyConsumedAggregation aggregateQuantity is derived from the resources in an EM's domain.	NetEnergyConsumedRuleSet	[1]

Connection Type	Role Name	Description	Role Type	Multiplicity
		(This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	See Clause 5.6.7.11.6.24.	
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the NetEnergyConsumedAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.24. NetEnergyConsumedRuleSet (Class)

Calculate "metered net energy consumed":

- For each direct subordinate that is an instance of the ElectricMeter Class, either sum ElectricMeter.intervalEnergyReading or sum (the present ElectricMeter.accumulatedEnergyReading - ElectricMeter.accumulatedEnergyReading at the beginning of the interval).

Calculate "net energy consumed from unmetered loads":

- For each direct subordinate that is an instance of the Load Class (including instances of the CurtableLoad Class) that is not included in "metered net energy consumed" and where Load.actualDemand is available, sum (integral of the Load.actualDemand over the time since the beginning of the interval).

Calculate "net energy consumed from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered net energy consumed" and where Generator.actualSupply is available, sum $(-1 * (\text{integral of the Generator.actualSupply over the time since the beginning of the interval}))$.

Calculate "net energy consumed from unmetered energy managers":

- For each direct subordinate that is an instance of the EM Class that is not included in "metered net energy consumed" and where EMPresentData.partialIntervalAggregatedNetEnergyConsumed is available, sum the subordinate EMPresentData.partialIntervalAggregatedNetEnergyConsumed.aggregateQuantity.

Calculate final result:

- Sum "metered net energy consumed", "net energy consumed from unmetered loads", "net energy consumed from unmetered generators and electrical storage devices," and "net energy consumed from unmetered energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\NetEnergyConsumedRuleSet.*

5.6.7.11.6.25. Supply Aggregation (Class)

For historical aggregations, this is the aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate present subinterval supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate supply of electrical generators and electrical storage devices directly or indirectly managed by the associated EM averaged over the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\SupplyAggregation.*
This element has the following connections to other elements:

Table 5.63 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The PowerMeasurementsSet resulting from a SupplyAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
Association	aggregationRuleset	The rules that define how a SupplyAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleset association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	SupplyRuleSet See Clause 5.6.7.11.6.26.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the SupplyAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDo main See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.26. SupplyRuleSet (Class)

Calculate "metered supply":

- For each direct subordinate that is an instance of the ElectricMeter Class where *only*
 1. direct subordinate instances of the Generator Class (including instances of the DispatchableGenerator Class) and/or
 2. direct subordinate instances of the EM Class where EM.presentAggregationMetadata.hasLoads is False
- are connected to its outputConnectionPoint and where ElectricMeter.powerReading is available, sum ElectricMeter.powerReading.

Calculate "supply from unmetered generators and electrical storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) that is not included in "metered supply" and where Generator.actualSupply is available, sum Generator.actualSupply.

Calculate "supply from energy managers":

- For each direct subordinate that is an instance of the EMClass that is not included in "metered supply" and where EMPresentData.presentAggregateSupply is available, sum the subordinate EMPresentData.presentAggregateSupply.aggregateQuantity.

Calculate final result:

- Sum "metered supply", "supply from unmetered generators and electrical storage devices", and "supply from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\SupplyRuleSet.*

5.6.7.11.6.27. ThermalEnergyStoredAggregation (Class)

For historical aggregations, this is the aggregate thermal energy stored by thermal storage devices directly or indirectly managed by the associated EM during the identified time interval.

For a partially completed (filling) demand interval, this is the aggregate thermal energy stored by thermal storage devices directly or indirectly managed by the associated EM since the beginning of the interval.

The exact manner in which a forecast aggregation is determined is a local matter. It shall represent the expected aggregate thermal energy stored by thermal storage devices directly or indirectly managed by the associated EM during the identified time interval.

Parent Class: Aggregation (See Clause 5.6.7.11.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ThermalEnergyStoredAggregation.*

This element has the following connections to other elements:

Table 5.64 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregateQuantity	The EnergyThermalQuantityt resulting from a ThermalEnergyStoredAggregation. (This association redefines the aggregateQuantity association between the Aggregation Class (see Clause 5.6.7.11.1) and the Measurement Class (see Clause 6.6.4.1).)	EnergyThermalQuantity See Clause 6.6.4.7.15.	[1]
Association	aggregationRuleSet	The rules that define how a ThermalEnergyStoredAggregation aggregateQuantity is derived from the resources in an EM's domain. (This association redefines the aggregationRuleSet association between the Aggregation Class (see Clause 5.6.7.11.1) and the RuleSet Class (see Clause 5.6.7.11.5).)	ThermalEnergyStoredRuleSet See Clause 5.6.7.11.6.28.	[1]
Association	collection	The instance of AllResourcesInEMDomain used as a source for deriving the ThermalEnergyStoredAggregation.aggregateQuantity. The multiplicity 0..1 reflects the fact that the associated Collection is required in order to compute the aggregateQuantity, but it is optional as to whether this Collection is retained. (This association redefines the collection association between the Aggregation Class (see Clause 5.6.7.11.1) and the Collection Class (see Clause 5.6.7.11.3).)	AllResourcesInEMDomain See Clause 5.6.7.11.2.	[0..1]

5.6.7.11.6.28. ThermalEnergyStoredRuleSet (Class)

Calculate "stored thermal energy from storage devices":

- For each direct subordinate that is an instance of the Generator Class (including instances of the DispatchableGenerator Class) where Generator.storage has the value "thermalStorage" and where Generator.actualStoredEnergy is available, sum Generator.actualStoredEnergy.

Calculate "stored thermal energy from energy managers":

- For each direct subordinate that is an instance of the EM Class and where EMPresentData.partialIntervalAggregateThermalEnergyStored is available, sum the subordinate EMPresentData.partialIntervalAggregateThermalEnergyStored.aggregateQuantity.

Calculate final result:

- Sum "stored thermal energy from storage devices" and "stored thermal energy from energy managers."

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Standard Aggregations\ThermalEnergyStoredRuleSet.*

5.6.7.12. Elements Defined in the FSGIMWeather Model

The following elements are defined in this model:

5.6.7.12.1. FSGIMWeatherArea (Class)

Defines the area of interest for a weather observation or forecast.

Parent Class: Aerodrome (See Clause 5.7.6.6.1.3.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherArea.*

5.6.7.12.2. FSGIMWeatherForecast (Class)

The weather forecast of observations at a future time and with a defined confidence and probability.

Parent Class: Forecast (See Clause 5.7.6.6.2.7.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherForecast.*

This element has the following connections to other elements:

Table 5.65 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	result	Contents of the Forecast are FSGIMWeatherBase objects. (This association redefines the result association between the Observation Class (see Clause 5.7.6.6.5.1.2) and the AbstractObject Class (see Clause 5.7.6.6.3.1.3).)	FSGIMWeatherBase See Clause 5.6.7.12.8.	[0..*]

5.6.7.12.3. FSGIMWeatherObservation (Class)

Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

Parent Class: Observation (See Clause 5.7.6.6.2.10.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherObservation.*

This element has the following connections to other elements:

Table 5.66 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	featureOfInterest	Area of interest associated with this Observation. (This association redefines the featureOfInterest association between the Observation Class (see Clause 5.7.6.6.5.1.2) and the AbstractFeature Class (see Clause 5.7.6.6.3.1.1).)	FSGIMWeatherArea See Clause 5.6.7.12.1.	[1]
Association	result	Contents of the Observation are FSGIMWeatherBase objects. (This association redefines the result association between the Observation Class (see Clause 5.7.6.6.5.1.2) and the AbstractObject Class (see Clause 5.7.6.6.3.1.3).)	FSGIMWeatherBase See Clause 5.6.7.12.8.	[0..*]

5.6.7.12.4. FSGIMWeatherReport (Class)

A container class for weather observations, forecasts, and trends.

Parent Class: AbstractReport (See Clause 5.7.6.6.1.9.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherReport.*

This element has the following connections to other elements:

Table 5.67 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	appliesTo	The area of interest to which this report applies. (This association redefines the appliesTo association between the AbstractReport Class (see Clause 5.7.6.6.1.9.1) and the Aerodrome Class (see Clause 5.7.6.6.1.3.1).)	FSGIMWeatherArea See Clause 5.6.7.12.1.	[0..1]
Association	FSGIMObservation	Observations that are associated with this report. (This association redefines the wxObservation association between the AbstractReport Class (see Clause 5.7.6.6.1.9.1) and the Observation Class (see Clause 5.7.6.6.2.10.1).)	FSGIMWeatherObservation See Clause 5.6.7.12.3.	[1..*]
Association	wxForecast	Forecasts associated with this report. (This association redefines the wxForecast association between the AbstractReport Class (see Clause 5.7.6.6.1.9.1) and the Forecast Class (see Clause 5.7.6.6.2.7.1).)	FSGIMWeatherForecast See Clause 5.6.7.12.2.	[0..*]

5.6.7.12.5. FSGIMWeatherTrend (Class)

A sequence of forecasts of weather in increments of a defined duration such as hourly.

Parent Class: TREND (See Clause 5.7.6.6.1.9.2.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherTrend*.

This element has the following connections to other elements:

Table 5.68 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	appliesTo	The area of interest to which this trend applies.	FSGIMWeatherArea See Clause 5.6.7.12.1.	[0..1]
Association	forecast	Sequence of forecasts for this Trend. (This association redefines the forecast association between the TREND Class (see Clause 5.7.6.6.1.9.2) and the Forecast Class (see Clause 5.7.6.6.2.7.1).)	FSGIMWeatherForecast See Clause 5.6.7.12.2.	[0..*]

5.6.7.12.6. Irradiance «Type» (Class)

The power of electromagnetic radiation per unit area incident on a surface, measured in W/m².

Parent Class: AbstractMeasure (See Clause 5.7.6.6.2.1.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\Irradiance*.

This element has the following connections to other elements:

Table 5.69 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Unit of measure for irradiance. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be WPerM2	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.6.7.12.7. **RelativeHumidity «Type» (Class)**

The relative humidity, the ratio of the partial pressure of water vapor to the equilibrium vapor pressure at constant temperature.

Parent Class: AbstractMeasure (See Clause 5.7.6.6.2.1.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\RelativeHumidity*.

This element has the following connections to other elements:

Table 5.70 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Unit of measure for relative humidity. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be none (dimensionless)	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.6.7.12.8. **FSGIMWeatherBase (Class)**

This class defines the basic weather information including temperature and humidity data.

Parent Class: AbstractWxFeature (See Clause 5.7.6.6.2.2.1.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWeatherBase*.

Table 5.71 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
airPressure	The air pressure.	Pressure See Clause 5.7.6.6.2.1.7.	[0..1]
airTemperature	The air temperature.	AirTemperature See Clause 5.7.6.6.2.8.1.	[0..1]
dewpointTemperature	The dew point temperature.	AirTemperature See Clause 5.7.6.6.2.8.1.	[0..1]
isDefinedFor	Provides a means to explicitly link to the corresponding feature.	AbstractFeature See Clause 5.7.6.6.3.1.1.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
maxTemperature	Maximum air temperature.	AirTemperature See Clause 5.7.6.6.2.8.1.	[0..1]
minTemperature	Minimum air temperature.	AirTemperature See Clause 5.7.6.6.2.8.1.	[0..1]
relativeHumidity	The relative humidity as a fraction. This value may be derived from the airTemperature and dewpointTemperature attributes.	RelativeHumidity See Clause 5.6.7.12.7.	[0..1]
sunriseTime	Time of day of sunrise.	UTCDateTime See Clause 6.6.1.11.	[0..1]
sunsetTime	Time of day of sunset.	UTCDateTime See Clause 6.6.1.11.	[0..1]

This element has the following connections to other elements:

Table 5.72 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	cloudCondition	Cloud condition.	CloudCondition See Clause 5.7.6.6.2.3.1.	[0..*]
Association	precipitation	Precipitations measurements.	FSGIMPPrecipitation See Clause 5.6.7.12.9.	[0..1]
Association	solar	Solar irradiance measurements.	FSGIMSolar See Clause 5.6.7.12.10.	[0..1]
Association	wind	Wind measurements	FSGIMWind See Clause 5.6.7.12.11.	[0..1]
Association	wxCondition	Weather condition description.	WxCondition See Clause 5.7.6.6.2.11.5.	[0..*]

5.6.7.12.9. FSGIMPPrecipitation (Class)

This class extends the basic weather data with precipitation data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMPPrecipitation.*

Table 5.73 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
depth	Measurement of precipitation.	Depth See Clause 5.7.6.6.2.8.3.	[0..1]
duration	The time over which accumulation was measured. The ending time is the obsOrFestTime	Duration See Clause 6.6.1.9.	[1]
Precipitation	Describes precipitation.	Precipitation See Clause 5.7.6.6.2.3.4.	[0..1]

5.6.7.12.10. FSGIMSolar (Class)

This class extends the basic weather data with solar data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMSolar*.

Table 5.74 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
diffuseIrradiance	Solar radiation that has been scattered in passing through the earth's atmosphere. It is equal to the global irradiance less the direct normal and ground-reflected components.	Irradiance See Clause 5.6.7.12.6.	[0..1]
directNormalIrradiance	Irradiance received from the sun without significant change of direction from the apparent position of the sun.	Irradiance See Clause 5.6.7.12.6.	[0..1]
globalHorizontalIrradiance	The global (or hemispheric) solar radiant energy is the quantity of solar energy incident upon a unit surface area in unit time through a unit hemisphere above the surface, expressed in W/m ² .	Irradiance See Clause 5.6.7.12.6.	[0..1]
planeOfArrayIrradiance	The hemispheric solar radiant energy incident on a planar surface that is tilted relative to the surface of the earth.	Irradiance See Clause 5.6.7.12.6.	[0..1]

5.6.7.12.11. FSGIMWind (Class)

This class extends the basic weather data with wind data.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\FSGIMWeather\FSGIMWind*.

Table 5.75 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
verticalWindGust	Vertical wind gust.	WindGust See Clause 5.7.6.6.2.8.9.	[0..1]
windDirection	Wind direction.	WindDirection See Clause 5.7.6.6.2.8.8.	[0..1]
windGust	Wind gust.	WindGust See Clause 5.7.6.6.2.8.9.	[0..1]
windSpeed	Wind speed.	WindSpeed See Clause 5.7.6.6.2.8.10.	[0..1]

5.6.7.13. Elements Defined in the Key Energy Manager Concepts Model

The following elements are defined in this model.

5.6.7.13.1. AggregationProperties (Class)

This class defines a set of attributes that characterize a Circuit with respect to Loads and Generators that are directly connected to it or indirectly connected through other Circuits that contribute to aggregations performed on this Circuit.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Key Energy Manager Concepts\AggregationProperties*.

Table 5.76 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
circuitOfAggregation	The Circuit for which an aggregation is being computed.	Circuit See Clause 6.6.5.1.	[1]
hasElectricalGenerators	This attribute indicates whether (True) or not (False) this Circuit has associated electrical generators. A Circuit has associated electrical generators if the Circuit has: - one or more direct subordinate instances of the Generator Class (including instances of the DispatchableGenerator), or; - one or more direct subordinate instances of the EM Class with the hasElectricalGenerators attribute for this Circuit set to True, or; - one or more direct subordinate instances of the ElectricMeter Class with the hasElectricalGenerators attribute for the Circuit connected to the output ConnectionPoint set to True, or; - an EnergyRouter contained within this EM where the associated aggregation Ruleset has caused the hasElectricalGenerators attribute for this Circuit to be set to True.	Boolean See Clause 6.6.3.3.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
hasLoads	This attribute indicates whether (True) or not (False) this Circuit has associated loads. A Circuit has associated loads if the Circuit has: - one or more direct subordinate instances of the Load Class (including instances of the CurtailableLoad Class), or; - one or more direct subordinate instances of the EM Class with the hasLoads attribute for this Circuit set to True, or; - one or more direct subordinate instances of the ElectricMeter Class with the hasLoads attribute for the Circuit connected to the output ConnectionPoint set to True, or; - an EnergyRouter contained within this EM where the associated aggregation Ruleset has caused the hasLoads attribute for this Circuit to be set to True.	Boolean See Clause 6.6.3.3.	[1]

5.6.7.14. Elements Defined in the EnergyRouter Model

The following elements are defined in this model.

5.6.7.14.1. BidirectionalCombiner (Class)

This class is used to model bidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that allows energy to flow from the onsite generation source to the utility grid but not from the utility grid to the onsite generation source. The effect of the constraints that are placed on this class results in the following behavior for the bidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.
- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power sent back to the grid is the amount of power available from the onsite generation side of the combiner box minus the amount of power drawn from load side of the combiner box.

Constraint: connection[0] connects to the utility grid side of the electrical system.

Constraint: connection[0].mayBeBidirectional = true

Constraint: connection[0].mayBeDisconnected = false

Constraint: connection[0].mayBeInput = false

Constraint: connection[0].mayBeOutput = false

Constraint: connection[0].presentState = bidirectional

Constraint: connection[1] connects to the onsite generation side of the electrical system.

Constraint: connection[1].mayBeBidirectional = false

Constraint: connection[1].mayBeDisconnected = false

Constraint: connection[1].mayBeInput = true

Constraint: connection[1].mayBeOutput = false

Constraint: connection[2] connects to the load side of the electrical system.
Constraint: connection[2].mayBeBidirectional = false
Constraint: connection[2].mayBeDisconnected = false
Constraint: connection[2].mayBeInput = false
Constraint: connection[2].mayBeOutput = true
Constraint: connection[2].presentState = input
Constraint: connection[2].presentState = output

Parent Class: EnergyRouter (See Clause 5.6.7.14.3.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\BidirectionalCombiner*.
This element has the following connections to other elements:

Table 5.77 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregationRuleset	The ruleset that defines how an Aggregation.aggregateQuantity is influenced by an BidirectionalCombiner. (This association redefines the aggregationRuleset association between the EnergyRouter Class (see Clause 5.6.7.14.3) and the RuleSet Class (see Clause 5.6.7.11.5).)	BidirectionalCombiner AggregationRuleSet See Clause 5.6.7.14.2.	[1]

5.6.7.14.2. BidirectionalCombinerAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a BidirectionalCombiner, calculate the Aggregation for the Circuit connected to BidirectionalCombiner.connection[1] and BidirectionalCombiner.connection[2] separately and then:
 - take the logical OR of AggregationProperties.hasElectricalGenerators for the Circuit connected to BidirectionalCombiner.connection[1] and AggregationProperties.hasElectricalGenerators for the Circuit connected to BidirectionalCombiner.connection[2];
 - take the logical OR of AggregationProperties.hasLoads for the Circuit connected to BidirectionalCombiner.connection[1] and AggregationProperties.hasLoads for the Circuit connected to BidirectionalCombiner.connection[2] and;
 - sum the Aggregation.aggregateQuantity for the Circuit connected to BidirectionalCombiner.connection[1] and the Aggregation.aggregateQuantity for the Circuit connected to BidirectionalCombiner.connection[2],to yield the resulting contributions of the BidirectionalCombiner to the Aggregation for the Circuit connected to BidirectionalCombiner.connection[0].

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\BidirectionalCombinerAggregationRuleSet.*

5.6.7.14.3. EnergyRouter (Abstract Class)

This class is used to model the various devices that are capable of connecting a load to more than one source of energy.
Constraint: The power into an EnergyRouter equals the power out of the EnergyRouter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\EnergyRouter.*
This element has the following connections to other elements:

Table 5.78 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregationRuleSet	The ruleset that defines how an Aggregation.aggregateQuantity is influenced by an EnergyRouter	RuleSet See Clause 5.6.7.11.5.	[1]
Composition	connection	The connection points that are a part of this EnergyRouter.	RouterConnectionPoint See Clause 5.6.7.14.6.	[3]

5.6.7.14.4. GridTransferSwitch (Class)

This class is used to model switchgear that are used to allow a load to be switched between two utility feeds without allowing the two utility feeds to be connected together through the switchgear.

- Constraint: connection[0] connects to the first utility grid side of the electrical system.
Constraint: connection[0].mayBeBidirectional = false
Constraint: connection[0].mayBeDisconnected = true
Constraint: connection[0].mayBeInput = true
Constraint: connection[0].mayBeOutput = false
Constraint: connection[1] connects to the second utility grid side of the electrical system.
Constraint: connection[1].mayBeBidirectional = false
Constraint: connection[1].mayBeDisconnected = true
Constraint: connection[1].mayBeInput = true
Constraint: connection[1].mayBeOutput = false
Constraint: connection[2] connects to the load side of the electrical system.
Constraint: connection[2].mayBeBidirectional = false
Constraint: connection[2].mayBeDisconnected = false
Constraint: connection[2].mayBeInput = false
Constraint: connection[2].mayBeOutput = true
Constraint: connection[2].presentState = output

Constraint: If connection[0].presentState is set to input then connection[1].presentState must be disconnected.
Constraint: If connection[1].presentState is set to input then connection[0].presentState must be disconnected.

Parent Class: EnergyRouter (See Clause 5.6.7.14.3.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\GridTransferSwitch*.
This element has the following connections to other elements:

Table 5.79 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregationRuleset	The ruleset that defines how an Aggregation.aggregateQuantity is influenced by a GridTransferSwitch. (This association redefines the aggregationRuleset association between the EnergyRouter Class (see Clause 5.6.7.14.3) and the RuleSet Class (see Clause 5.6.7.11.5).)	GridTransferSwitchAggregationRuleSet See Clause 5.6.7.14.5.	[1]

5.6.7.14.5. GridTransferSwitchAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a GridTransferSwitch, calculate the Aggregation for the Circuit connected to GridTransferSwitch.connection[2].
- If GridTransferSwitch.connection[0].presentState has a value of "input," report:
 1. the Aggregation.aggregateQuantity for the Circuit connected to GridTransferSwitch.connection[2] as the contribution of the GridTransferSwitch to the Aggregation for the Circuit connected to GridTransferSwitch.connection[0], and with
 2. the AggregationProperties.hasElectricalGenerators, and the AggregationProperties.hasLoads for the Circuit connected to GridTransferSwitch.connection[2] as the contribution of the GridTransferSwitch to the AggregationProperties for the Circuit connected to GridTransferSwitch.connection[0].
- If GridTransferSwitch.connection[0].presentState has a value of "disconnected," report:
 1. the results of the Aggregation for the Circuit connected to GridTransferSwitch.connection[2] with all associated MeasurementQuantity.values, including those indirectly associated as part of a MeasurementSet, set to 0 and all associated MeasurementQuantity.uncertainty values, including those indirectly associated as part of a MeasurementSet, set to 0, as the contribution of the GridTransferSwitch to the Aggregation for the Circuit connected to GridTransferSwitch.connection[0], and with
 2. AggregationProperties.hasElectricalGenerators set to False and AggregationProperties.hasLoads set to False as the contribution of the Grid TransferSwitch to the AggregationProperties for the Circuit connected to GridTransferSwitch.connection[0].
- If GridTransferSwitch.connection[1].presentState has a value of "input," report:
 1. the Aggregation.aggregateQuantity for the Circuit connected to GridTransferSwitch.connection[2] as the contribution of the GridTransferSwitch to the Aggregation for the Circuit connected to GridTransferSwitch.connection[1], and with

2. the `AggregationProperties.hasElectricalGenerators`, and the `AggregationProperties.hasLoads` for the `Circuit` connected to `GridTransferSwitch.connection[2]` as the contribution of the `GridTransferSwitch` to the `AggregationProperties` for the `Circuit` connected to `GridTransferSwitch.connection[1]`.
 - If `GridTransferSwitch.connection[1].presentState` has a value of "disconnected," report:
 1. the results of the `Aggregation` for the `Circuit` connected to `GridTransferSwitch.connection[2]` with all associated `MeasurementQuantity` values, including those indirectly associated as part of a `MeasurementSet`, set to 0 and all associated `MeasurementQuantity` uncertainty values, including those indirectly associated as part of a `MeasurementSet`, set to 0 as the contribution of the `GridTransferSwitch` to the `Aggregation` for the `Circuit` connected to `GridTransferSwitch.connection[1]`, and with `AggregationProperties.hasElectricalGenerators` set to `False`, and `AggregationProperties.hasLoads` set to `False` as the contribution of the `GridTransferSwitch` to the `AggregationProperties` for the `Circuit` connected to `GridTransferSwitch.connection[1]`.
 2. `AggregationProperties.hasElectricalGenerators` set to `False`, and `AggregationProperties.hasLoads` set to `False` as the contribution of the `GridTransferSwitch` to the `AggregationProperties` for the `Circuit` connected to `GridTransferSwitch.connection[1]`.

Parent Class: `RuleSet` (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\GridTransferSwitchAggregationRuleSet.*

5.6.7.14.6. RouterConnectionPoint (Class)

This class is used to model the input/output characteristics of connections from an `EnergyRouter` to a source or sink of energy.

Parent Class: `ConnectionPoint` (See Clause 6.6.5.2.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\RouterConnectionPoint.*

Table 5.80 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
<code>mayBeInput</code>	This <code>RouterConnectionPoint</code> may be used as a connection that allows energy to flow into, but not out of, the <code>EnergyRouter</code> .	Boolean See Clause 6.6.3.3.	[1]
<code>mayBeOutput</code>	This <code>RouterConnectionPoint</code> may be used as a connection that allows energy to flow out of, but not into, the <code>EnergyRouter</code> .	Boolean See Clause 6.6.3.3.	[1]
<code>mayBeBidirectional</code>	This <code>RouterConnectionPoint</code> may be used as a connection that allows energy to flow into and out of the <code>EnergyRouter</code> .	Boolean See Clause 6.6.3.3.	[1]
<code>mayBeDisconnected</code>	This <code>RouterConnectionPoint</code> may be used as a connection that may be disconnected from the <code>EnergyRouter</code> .	Boolean See Clause 6.6.3.3.	[1]
<code>presentState</code>	The present state of this <code>RouterConnectionPoint</code> . The allowable states for this particular <code>RouterConnectionPoint</code> are determined by the boolean attributes of this class.	<code>ConnectionPointState</code> See Clause 5.6.7.14.11.	[1]

5.6.7.14.7. TransferSwitch (Class)

This class is used to model transfer switches, which are devices that allow a load or set of loads to be switched between two power sources without allowing the two power sources to be connected together.

Constraint: `connection[2].mayBeDisconnected = false`

Constraint: connection[2].mayBeInput = false
Constraint: connection[2].mayBeOutput = true
Constraint: connection[2].presentState = output
Constraint: If connection[0].presentState is set to input then connection[1].presentState must be disconnected.
Constraint: If connection[1].presentState is set to input then connection[0].presentState must be disconnected.
Constraint: connection[0] connects to the utility grid side of the electrical system.
Constraint: connection[0].mayBeBidirectional = false
Constraint: connection[0].mayBeDisconnected = true
Constraint: connection[0].mayBeInput = true
Constraint: connection[0].mayBeOutput = false
Constraint: connection[1] connects to the onsite generation side of the electrical system.
Constraint: connection[1].mayBeBidirectional = false
Constraint: connection[1].mayBeDisconnected = true
Constraint: connection[1].mayBeInput = true
Constraint: connection[1].mayBeOutput = false
Constraint: connection[2] connects to the load side of the electrical system.
Constraint: connection[2].mayBeBidirectional = false

Parent Class: EnergyRouter (See Clause 5.6.7.14.3.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\TransferSwitch.*

This element has the following connections to other elements:

Table 5.81 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregationRuleset	The ruleset that defines how an Aggregation.aggregateQuantity is influenced by a TransferSwitch. (This association redefines the aggregationRuleset association between the EnergyRouter Class (see Clause 5.6.7.14.3) and the RuleSet Class (see Clause 5.6.7.11.5).)	TransferSwitchAggregationRuleSet See Clause 5.6.7.14.8.	[1]

5.6.7.14.8. TransferSwitchAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a TransferSwitch, calculate the Aggregation for the Circuit connected to TransferSwitch.connection[1] and TransferSwitch.connection[2] separately and then execute the following steps:
 - Set AggregationProperties.hasLoads for the Circuit connected to TransferSwitch.connection[0] = True;
 - Set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] = False;
 - If TransferSwitch.connection[0].presentState is "input" and AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[2] is True, set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] to True;

4. If TransferSwitch.connection[1].presentState is "input" and AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[1] is True, set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] to True, and;
5. If AggregationProperties.hasLoads for the Circuit connected to TransferSwitch.connection[1] is True and AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[1] is True, set AggregationProperties.hasElectricalGenerators for the Circuit connected to TransferSwitch.connection[0] to True.
- Use the resulting aggregateQuantity, hasLoads flag, and hasElectricalGenerators flag for the resulting contributions of the TransferSwitch to the Aggregation and AggregationProperties for the Circuit connected to TransferSwitch.connection[0].

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\TransferSwitchAggregationRuleSet.*

5.6.7.14.9. UnidirectionalCombiner (Class)

This class is used to model unidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that does not allow energy to flow from one source to another. The effect of the constraints that are placed on this class results in the following behavior for the unidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of the combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.
- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of the combiner box is zero and the amount of power actually injected into the onsite generation side of the combiner box is equal to the amount of power drawn from load side of the combiner box.

Constraint: connection[0] connects to the utility grid side of the electrical system.

Constraint: connection[0].mayBeBidirectional = false

Constraint: connection[0].mayBeDisconnected = false

Constraint: connection[0].mayBeInput = true

Constraint: connection[0].mayBeOutput = false

Constraint: connection[0].presentState = input

Constraint: connection[1] connects to the onsite generation side of the electrical system.

Constraint: connection[1].mayBeBidirectional = false

Constraint: connection[1].mayBeDisconnected = false

Constraint: connection[1].mayBeInput = true

Constraint: connection[1].mayBeOutput = false

Constraint: connection[1].presentState = input

Constraint: connection[2] connects to the load side of the electrical system.

Constraint: connection[2].mayBeBidirectional = false

Constraint: connection[2].mayBeDisconnected = false

Constraint: connection[2].mayBeInput = false

Constraint: connection[2].mayBeOutput = true

Constraint: connection[2].presentState = output

Parent Class: EnergyRouter (See Clause 5.6.7.14.3.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\UnidirectionalCombiner.*

This element has the following connections to other elements:

Table 5.82 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	aggregationRuleset	The ruleset that defines how an Aggregation.aggregateQuantity is influenced by a UnidirectionalCombiner. (This association redefines the aggregationRuleset association between the EnergyRouter Class (see Clause 5.6.7.14.3) and the RuleSet Class (see Clause 5.6.7.11.5).)	UnidirectionalCombinerAggregationRuleSet See Clause 5.6.7.14.10.	[1]

5.6.7.14.10. UnidirectionalCombinerAggregationRuleSet (Class)

- When calculating an Aggregation on Circuits connected to a UnidirectionalCombiner, calculate the Aggregation for the Circuit connected to UnidirectionalCombiner.connection[1] and UnidirectionalCombiner.connection[2] separately and then:
 - take the logical OR of AggregationProperties.hasElectricalGenerators for the Circuit connected to UnidirectionalCombiner.connection[1] and AggregationProperties.hasElectricalGenerators for the Circuit connected to UnidirectionalCombiner.connection[2];
 - take the logical OR of AggregationProperties.hasLoads for the Circuit connected to UnidirectionalCombiner.connection[1] and AggregationProperties.hasLoads for the Circuit connected to UnidirectionalCombiner.connection[2] and;
 - sum the Aggregation.aggregateQuantity for the Circuit connected to UnidirectionalCombiner.connection[1] and the Aggregation.aggregateQuantity for the Circuit connected to UnidirectionalCombiner.connection[2],to yield the resulting contributions of the UnidirectionalCombiner to the Aggregation for the Circuit connected to UnidirectionalCombiner.connection[0].

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\UnidirectionalCombinerAggregationRuleSet.*

5.6.7.14.11. ConnectionPointState (Enumeration)

This enumeration is used to specify the present configuration of a ConnectionPoint.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\ConnectionPointState.*

Table 5.83 - ConnectionPointState Enumerated Values

Enumerated Value	Definition
input	This RouterConnectionPoint is currently configured as an input which allows energy to flow into, but not out of, the EnergyRouter.
output	This RouterConnectionPoint is currently configured as an input which allows energy to flow out of, but not into, the EnergyRouter.
bidirectional	This RouterConnectionPoint is currently configured as an input which allows energy to flow into and out of the EnergyRouter.
disconnected	This RouterConnectionPoint is currently disconnected from the EnergyRouter.

5.7. Model Elements from External Sources

The four orthogonal components of the Facility Smart Grid Information Model make use of elements from the external models shown in this clause.

UML Package Location: *Model\Model Elements from External Sources*

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

5.7.1. Energy Interoperation wd36/PR03

The Energy Interoperation Component is the portion of the FSGIM that provides an abstract representation of information needed for collaborative and transactive use of energy. It defines information used to coordinate energy supply and use, including power and ancillary services, between two parties such as energy suppliers and facilities.

UML Package Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03*

5.7.1.1. EIClasses

The Energy Interoperation Component contains classes that support the ability to act upon dynamic price signals, energy market trading, grid reliability events, and grid emergency events.

UML Package Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses*

5.7.1.1.1. EI Events Diagram

The EI Events Diagram illustrates the UML model that is used to call for performance under a transaction. The service parameters and event information distinguish different types of events. Event types include reliability events, emergency events, and others.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



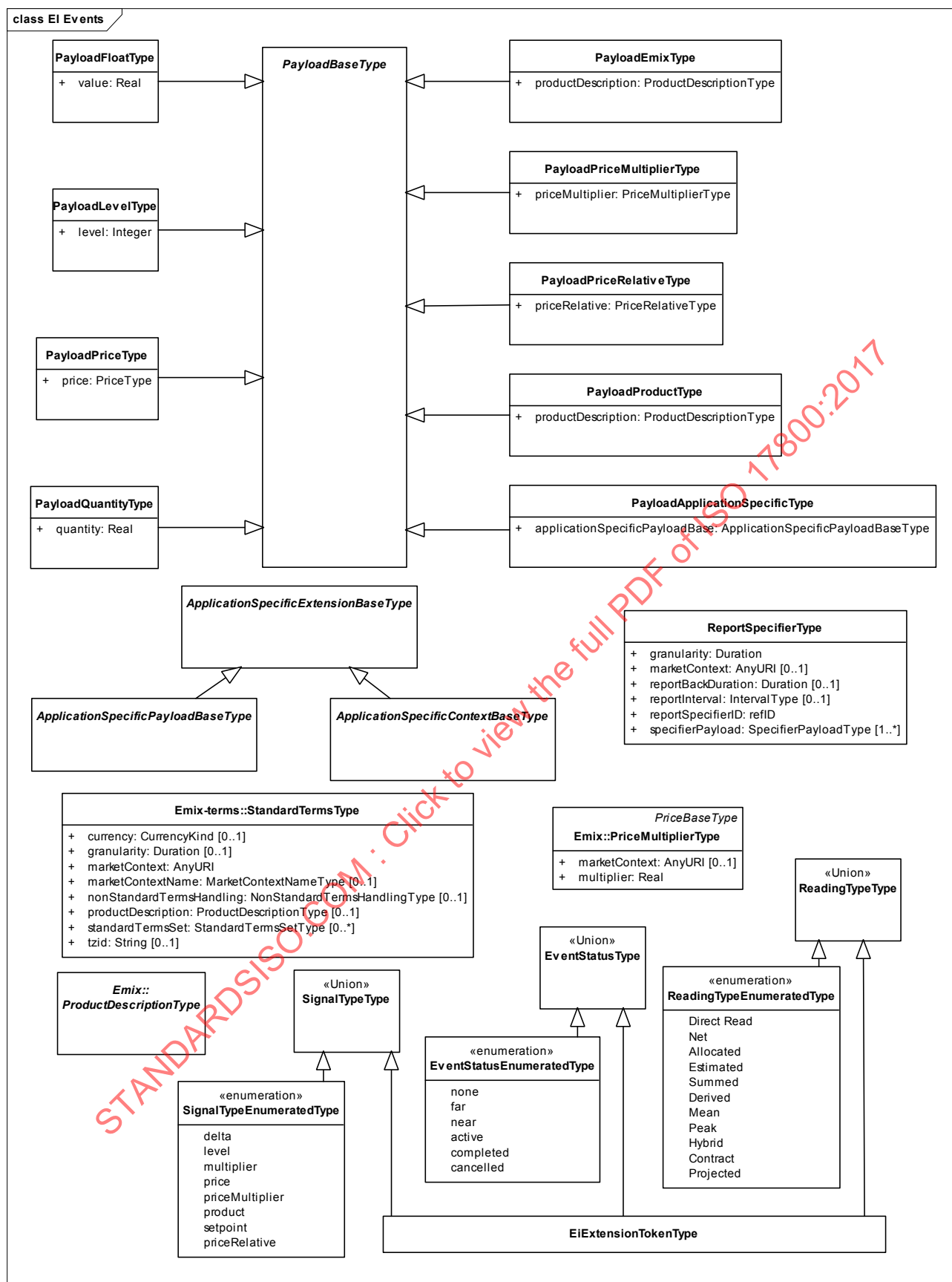


Figure 5.38 - EI Events

The EI Events Diagram makes use of the following elements:

ApplicationSpecificContextBaseType (Abstract Class)

The Application Specific Context Base is an abstract class to exchange invariant or setup information with an Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificContextBaseType*. See Clause 5.7.1.2.1.1 for the complete definition of this element.

ApplicationSpecificExtensionBaseType (Abstract Class)

Application Extensions are used to provide hints to or interactions with Applications running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificExtensionBaseType*. See Clause 5.7.1.2.1.2 for the complete definition of this element.

ApplicationSpecificPayloadBaseType (Abstract Class)

The Application Specific Payload Base is an abstract class to exchange feedback with an Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificPayloadBaseType*. See Clause 5.7.1.2.1.3 for the complete definition of this element.

ArrayOfSignals (Class)

Collection of Signal Base derived elements

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ArrayOfSignals*. See Clause 5.7.1.2.1.4 for the complete definition of this element.

CurrentValueType (Class)

This class contains the common information, at the time of an event, in an Energy Interoperation signal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\CurrentValueType*. See Clause 5.7.1.2.1.5 for the complete definition of this element.

EiEventBaselineType (Class)

This type is used for describing baseline for the event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiEventBaselineType*. See Clause 5.7.1.2.1.6 for the complete definition of this element.

EiEventSignalType (Class)

This type is used for describing the signal type communications.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiEventSignalType*. See Clause 5.7.1.2.1.7 for the complete definition of this element.

EiEventType (Class)

Core element of event-based demand response that specifies the duration and applicability of an Event. Some deadlines, time periods, and transitions may not be applicable to all products or services.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiEventType*. See Clause 5.7.1.2.1.8 for the complete definition of this element.

EiExtensionTokenType (Class)

Pattern used for extending string enumeration, where allowed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiExtensionTokenType*. See Clause 5.7.1.2.1.9 for the complete definition of this element.

EiMarketContextType (Class)

Terms that define the market context of this transaction. The Market Context is the overall market or program rules that govern this event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiMarketContextType*. See Clause 5.7.1.2.1.10 for the complete definition of this element.

EiTargetType (Class)

A set of elements to that collectively name who is participating or should participate in an EI interactions

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiTargetType*. See Clause 5.7.1.2.1.11 for the complete definition of this element.

EmixInterfaceType (Abstract Class)

Abstract base class for the interfaces for EMIX Product delivery, measurement, and/or pricing.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixInterfaceType*. See Clause 5.7.2.13.1.3 for the complete definition of this element.

EnvelopeContentsType (Class)

EMIX Warrants are delivered as Envelope Contents. Assertions that distinguish the commodity product are called EMIX Warrants. A common definition of a warrant is a written assurance that some product or service will be provided or will meet certain specifications.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EnvelopeContentsType*. See Clause 5.7.2.13.1.5 for the complete definition of this element.

EventDescriptorType (Class)

A description of the type of event being requested, indicating the market context and priority of the event as well as when the event was created, modified, and is scheduled to become active.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventDescriptorType*. See Clause 5.7.1.2.1.12 for the complete definition of this element.

EventSignalTypeBase (Abstract Class)

This class conveys the kind of demand response event being signaled and the desired resulting load or generation.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventSignalTypeBase*. See Clause 5.7.1.2.1.13 for the complete definition of this element.

EventStatusEnumeratedType (Enumeration)

Indicates the present status of an event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventStatusEnumeratedType*. See Clause 5.7.1.2.1.32 for the complete definition of this element.

EventStatusType (Class)

Indicates the present status of an event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventStatusType*. See Clause 5.7.1.2.1.14 for the complete definition of this element.

FSGIMEventSignalType (Class)

This class conveys the kind of demand response event being signaled from an energy manager to a curtailable load or a dispatchable generator. The signal indicates the desired resulting load or generation.

For FSGIMEventSignalType, the multiplicity of eventInterval is 0.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\FSGIMEventSignalType*. See Clause 5.7.1.2.1.15 for the complete definition of this element.

GluonType (Class)

A GluonType influences the serialization of IntervalTypes in a sequence. A GluonType is often used as a pointer to the beginning interval in a sequence, although in general it can point to any interval in a sequence, or simply describe an isolated interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\GluonType*. See Clause 5.7.3.2.6 for the complete definition of this element.

IntervalType (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\IntervalType*. See Clause 5.7.3.2.7 for the complete definition of this element.

ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType*. See Clause 6.6.4.3 for the complete definition of this element.

PayloadApplicationSpecificType (Class)

This Payload contains the information that changes in each Stream payload.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadApplicationSpecificType*. See Clause 5.7.1.2.1.16 for the complete definition of this element.

PayloadBaseType (Abstract Class)

base for information in Signal / Baseline / Report Payloads

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadBaseType*. See Clause 5.7.1.2.1.17 for the complete definition of this element.

PayloadEmixType (Class)

This is the Payload for Signals that require an EMIX Product Description.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadEmixType*. See Clause 5.7.1.2.1.18 for the complete definition of this element.

PayloadFloatType (Class)

This is the Payload for Signals that require a Quantity.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadFloatType*. See Clause 5.7.1.2.1.19 for the complete definition of this element.

PayloadLevelType (Class)

This is the Payload for Signals that convey Simple Levels.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadLevelType*. See Clause 5.7.1.2.1.20 for the complete definition of this element.

PayloadPriceMultiplierType (Class)

A Price Multiplier that when applied to a reference price produces the actual price.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadPriceMultiplierType*. See Clause 5.7.1.2.1.21 for the complete definition of this element.

PayloadPriceRelativeType (Class)

A Price Relative is added to a reference price to compute the actual price. Price Relative may be positive or negative.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadPriceRelativeType*. See Clause 5.7.1.2.1.22 for the complete definition of this element.

PayloadPriceType (Class)

This is the Payload for Signals that require a Quantity.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadPriceType*. See Clause 5.7.1.2.1.23 for the complete definition of this element.

PayloadProductType (Class)

This Payload contains the information that changes in each Stream payload.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadProductType*. See Clause 5.7.1.2.1.24 for the complete definition of this element.

PayloadQuantityType (Class)

This is the Payload for Signals that require a Quantity.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadQuantityType*. See Clause 5.7.1.2.1.25 for the complete definition of this element.

PriceMultiplierType (Class)

Multiplier times market price, 1 for same as market

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceMultiplierType*. See Clause 5.7.2.13.1.9 for the complete definition of this element.

PriceRelativeType (Class)

Price Relative is a fixed charge (positive or negative) applied to base price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceRelativeType*. See Clause 5.7.2.13.1.10 for the complete definition of this element.

PriceType (Class)

Simple Price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceType*. See Clause 5.7.2.13.1.11 for the complete definition of this element.

ProductDescriptionType (Abstract Class)

In EMIX, the Product Description is placed in the Interval or Gluon attachment. The respective product schemas extend this abstract class.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\ProductDescriptionType*. See Clause 5.7.2.13.1.12 for the complete definition of this element.

ReadingTypeEnumeratedType (Enumeration)

Indicates metadata about the Readings, i.e., direct measurement or computation.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ReadingTypeEnumeratedType*. See Clause 5.7.1.2.1.33 for the complete definition of this element.

ReadingTypeType (Class)

Type of Reading.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ReadingTypeType*. See Clause 5.7.1.2.1.26 for the complete definition of this element.

refID (Class)

Reference ID for a particular instance, transmittal, or artifact. Note: not the same as the native ID of the object being transmitted or shared.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\refID*. See Clause 5.7.1.2.1.27 for the complete definition of this element.

ReportSpecifierType (Class)

Parameters that define the content of a Report Stream

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ReportSpecifierType*. See Clause 5.7.1.2.1.28 for the complete definition of this element.

Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be surmised by monitoring the time references.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*. See Clause 5.6.7.10.4 for the complete definition of this element.

SignalTypeEnumType (Enumeration)

SignalTypeEnumType lists the pre-defined Types used to specify the Payload Types and conformance in a Stream

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SignalTypeEnumType*. See Clause 5.7.1.2.1.34 for the complete definition of this element.

SignalTypeType (Class)

SignalType is used in EventSignals to specify the Payload Types in a Signal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SignalTypeType*. See Clause 5.7.1.2.1.29 for the complete definition of this element.

SimpleLevelContextType (Class)

Simple Levels are a set of simple indicators about scarcity and value, in which an ordered set of values indicate energy scarcity is above normal, normal, or below normal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SimpleLevelContextType*. See Clause 5.7.1.2.1.30 for the complete definition of this element.

SpecifierPayloadType (Class)

Payload for use in Report Specifiers.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SpecifierPayloadType*. See Clause 5.7.1.2.1.31 for the complete definition of this element.

StandardTermsType (Class)

Standard Terms express overarching terms for a market context

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\StandardTermsType*. See Clause 5.7.2.13.2.23 for the complete definition of this element.

5.7.1.2. Elements Defined in the Energy Interoperation wd36/PR03 Model

The following elements are defined in this model.

5.7.1.2.1. Elements Defined in the EIClasses Model

The following elements are defined in this model.

5.7.1.2.1.1. ApplicationSpecificContextBaseType (Abstract Class)

The Application Specific Context Base is an abstract class to exchange invariant or setup information with an Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

Parent Class: ApplicationSpecificExtensionBaseType (See Clause 5.7.1.2.1.2.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificContextBaseType.*

5.7.1.2.1.2. ApplicationSpecificExtensionBaseType (Abstract Class)

Application Extensions are used to provide hints to or interactions with Applications running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificExtensionBaseType.*

5.7.1.2.1.3. ApplicationSpecificPayloadBaseType (Abstract Class)

The Application Specific Payload Base is an abstract class to exchange feedback with an Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

Parent Class: ApplicationSpecificExtensionBaseType (See Clause 5.7.1.2.1.2.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificPayloadBaseType.*

5.7.1.2.1.4. ArrayOfSignals (Class)

Collection of Signal Base derived elements

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ArrayOfSignals.*

Table 5.84 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
eiEventSignal	This type is used for describing the signal type communications.	EiEventSignalType See Clause 5.7.1.2.1.7.	[0..*]
eiEventBaseline	This type is used for describing baseline for the event.	EiEventBaselineType See Clause 5.7.1.2.1.6.	[0..*]

5.7.1.2.1.5. CurrentValueType (Class)

This class contains the common information, at the time of an event, in an Energy Interoperation signal.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\CurrentValueType*.

Table 5.85 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
payloadBase	Common information in all streams. This commonality is derived from the commonality of a request for performance (Signal), a report of performance (Report and Delivery), projections of performance (Projection), and a baseline of performance (Baseline).	PayloadBaseType See Clause 5.7.1.2.1.17.	[1]

5.7.1.2.1.6. EiEventBaselineType (Class)

This type is used for describing baseline for the event.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\EiEventBaselineType*.

Table 5.86 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
baselineID	Reference identifier of a baseline.	GlobalId See Clause 6.6.3.4.	[1]
baselineInterval	One or more baseline intervals indicate comparable times that may be used as a baseline. Information for the baseline intervals is not conveyed in this type, and the interval may be past, present, or future.	IntervalType See Clause 5.7.3.2.7.	[0..*]
baselineName	Human readable name of the baseline.	String See Clause 6.6.3.7.	[0..1]
currentValue	The present value (as of the signal creation) of the resource.	CurrentValueType See Clause 5.7.1.2.1.5.	[0..1]
itemBase	This indicates the units of the signal.	ItemBaseType See Clause 6.6.4.3.	[0..1]
resourceID	Unique identifier of the resource being baselined.	GlobalId See Clause 6.6.3.4.	[0..*]

This element has the following connections to other elements:

Table 5.87 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	eventInterval	Identifies the time interval to which the event baseline applies.	Sequence See Clause 5.6.7.10.4.	[0..1]

5.7.1.2.1.7. EiEventSignalType (Class)

This type is used for describing the signal type communications.

Parent Class: EventSignalTypeBase (See Clause 5.7.1.2.1.13.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\EiEventSignalType.*

Table 5.88 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
eiTarget	A set of elements that collectively name which Parties should participate in an event. A Target can include Service Areas, named Groups, Resources, and other standard identifiers.	EiTargetType See Clause 5.7.1.2.1.11.	[0..*]
itemBase	Enumerates what is being measured, and perhaps paid for, by the signal.	ItemBaseType See Clause 6.6.4.3.	[0..1]
marketContext	The Market Context is the overall market or program rules that govern this event.	AnyURI See Clause 6.6.3.2.	[0..1]
schemaVersion	Version number of the schema used in the signal.	String See Clause 6.6.3.7.	[1]
signalID	Reference identifier of this signal.	refID See Clause 5.7.1.2.1.27.	[1]
signalName	Human readable name of the signal.	String See Clause 6.6.3.7.	[1]

5.7.1.2.1.8. EiEventType (Class)

Core element of event-based demand response that specifies the duration and applicability of an Event. Some deadlines, time periods, and transitions may not be applicable to all products or services.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\EiEventType.*

Table 5.89 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
eiActivePeriod	Duration that encompasses the event.	GlueonType See Clause 5.7.3.2.6.	[1]
eiEventSignals	The detailed information about the schedule for an event	ArrayOfSignals See Clause 5.7.1.2.1.4.	[1]
eventMessageID	Reference ID for the EI Event instance	refID See Clause 5.7.1.2.1.27.	[0..1]
schemaVersion	Version number of the schema used in the event signal.	String See Clause 6.6.3.7.	[1]

This element has the following connections to other elements:

Table 5.90 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	eiTarget	A set of elements that collectively names the parties participating in an event.	EiTargetType See Clause 5.7.1.2.1.11.	[0..1]
Association	eventDescriptor	A description of the type of event being requested, indicating the market context and priority of the event as well as when the event was created, modified, and is scheduled to become active.	EventDescriptorType See Clause 5.7.1.2.1.12.	[1]

5.7.1.2.1.9. EiExtensionTokenType (Class)

Pattern used for extending string enumeration, where allowed

Parent Classes: EventStatusType (See Clause 5.7.1.2.1.14), SignalTypeType (See Clause 5.7.1.2.1.29), ReadingTypeType (See Clause 5.7.1.2.1.26)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\EiExtensionTokenType*.

5.7.1.2.1.10. EiMarketContextType (Class)

Terms that define the market context of this transaction. The Market Context is the overall market or program rules that govern this event.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\EiMarketContextType*.

Table 5.91 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
applicationSpecificContextBase	Invariant or setup information with an Application running on the other side of an interaction.	ApplicationSpecificContextBaseType See Clause 5.7.1.2.1.1.	[0..1]
createdDateTime	Indicates when this artifact was created.	DateTime See Clause 6.6.1.7.	[0..1]
envelopeContents	The extrinsic information that may distinguish the product from being a pure commodity.	EnvelopeContentsType See Clause 5.7.2.13.1.5.	[0..1]
marketContext	Terms that define the market context of this transaction. The Market Context is the overall market or program rules that govern this event.	AnyURI See Clause 6.6.3.2.	[1]
marketName	Name associated with the Market Context	String See Clause 6.6.3.7.	[0..1]
reportSpecifier	Specification of what reports are being requested. It includes granularity, reporting interval, market context, and a specifier payload.	ReportSpecifierType See Clause 5.7.1.2.1.28.	[0..*]
schemaVersion	Version number of the schema used in the event signal.	String See Clause 6.6.3.7.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
simpleLevelContext	An agreement-based interaction abstracted away from expressions of value or actual amounts. Simple Levels define levels of energy scarcity and abundance, at an agreed upon granularity.	SimpleLevelContextType See Clause 5.7.1.2.1.30.	[0..1]
standardTerms	Common expectations outside of any single product-related artifact.	StandardTermsType See Clause 5.7.2.13.2.23.	[0..1]

5.7.1.2.1.11. EITargetType (Class)

A set of elements to that collectively name who is participating or should participate in an EI interactions

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EITargetType*.

Table 5.92 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
emixInterface	The interfaces for EMIX Product delivery, measurement, and/or pricing.	EmixInterfaceType See Clause 5.7.2.13.1.3.	[0..*]
groupID	Unique identifier of a group which may be the target of an event.	GlobalId See Clause 6.6.3.4.	[0..*]
groupName	Name of a group which may be the target of an event.	String See Clause 6.6.3.7.	[0..*]
partyID	Identifier for a party within a transaction.	GlobalId See Clause 6.6.3.4.	[0..*]
resourceID	Unique identifier for a resource.	GlobalId See Clause 6.6.3.4.	[0..*]
venID	An identifier of a party acting in a specific role in a market managed by another party that sends events market information to it.	GlobalId See Clause 6.6.3.4.	[0..*]

5.7.1.2.1.12. EventDescriptorType (Class)

A description of the type of event being requested, indicating the market context and priority of the event as well as when the event was created, modified, and is scheduled to become active.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventDescriptorType*.

Table 5.93 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
createdDateTime	Indicates when this artifact was created.	Date Time See Clause 6.6.1.7.	[0..1]
eiMarketContext	The Market Context is the overall market or program rules that govern this event.	EiMarketContextType See Clause 5.7.1.2.1.10.	[1]
eventId	Unique identifier of an event.	GlobalId See Clause 6.6.3.4.	[1]
eventStatus	Indicates the present status of an event.	EventStatusType See Clause 5.7.1.2.1.14.	[0..1]
modificationDateTime	The date and time a modification takes effect.	Date Time See Clause 6.6.1.7.	[0..1]
modificationNumber	Used to indicate if the event has been modified and is incremented each time the event is modified.	Integer See Clause 6.6.3.5.	[1]
modificationReason	The reason the event is being cancelled or modified.	String See Clause 6.6.3.7.	[0..1]
operatingDay	Date of Start of Event	Date Time See Clause 6.6.1.7.	[0..1]
priority	This is the priority of this event relative to other events. The lower the number higher the priority. A value of zero (0) indicates NO priority and in essence is the lowest priority by default.	Integer See Clause 6.6.3.5.	[0..1]
testEvent	testEvent can be treated as a boolean by either not including it (= false) or using the null string.	String See Clause 6.6.3.7.	[0..1]
vtnComment	Additional Event information provided by the Operator.	String See Clause 6.6.3.7.	[0..1]

5.7.1.2.1.13. EventSignalTypeBase (Abstract Class)

This class conveys the kind of demand response event being signaled and the desired resulting load or generation

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\EventSignalTypeBase.*

Table 5.94 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
currentValue	The present value (as of the signal creation) of the resource.	CurrentValueType See Clause 5.7.1.2.1.5.	[0..1]
signalType	Type of signal (delta, multiplier, level, price, product, or set-point) being conveyed.	SignalTypeType See Clause 5.7.1.2.1.29.	[1]

This element has the following connections to other elements:

Table 5.95 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	eventInterval	Identifies the time interval to which the event signal applies.	Sequence See Clause 5.6.7.10.4.	[0..1]

5.7.1.2.1.14. **EventStatusType «Union» (Class)**

Indicates the present status of an event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\EventStatusType.*

5.7.1.2.1.15. **FSGIMEventSignalType (Class)**

This class conveys the kind of demand response event being signaled from an energy manager to a curtailable load or a dispatchable generator. The signal indicates the desired resulting load or generation.

For FSGIMEventSignalType, the multiplicity of eventInterval is 0.

Constraint: For FSGIMEventSignalType, currentValue.payloadBase must be of type FSGIMPayloadBaseType.

Constraint: For FSGIMEventSignalType, signalType must be of type SignalTypeEnumeratedType.

Constraint: For FSGIMEventSignalType, the multiplicity of eventInterval is 0.

Parent Class: EventSignalTypeBase (See Clause 5.7.1.2.1.13.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\FSGIMEventSignalType.*

5.7.1.2.1.16. **PayloadApplicationSpecificType (Class)**

This Payload contains the information that changes in each Stream payload.

Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EI\Classes\PayloadApplicationSpecificType.*

Table 5.96 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
applicationSpecificPayloadBase	An abstract class to exchange feedback with an Application running on the other side of an interaction.	ApplicationSpecificPayloadBaseType See Clause 5.7.1.2.1.3.	[1]

5.7.1.2.1.17. PayloadBaseType (Abstract Class)
base for information in Signal / Baseline / Report Payloads

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\PayloadBaseType.*

5.7.1.2.1.18. PayloadEmixType (Class)
This is the Payload for Signals that require an EMIX Product Description.
Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\PayloadEmixType.*

Table 5.97 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
productDescription	The terms of energy tenders and transactions for which time and location are essential characteristics.	ProductDescriptionType ^e See Clause 5.7.2.13.1.12.	[1]

5.7.1.2.1.19. PayloadFloatType (Class)
This is the Payload for Signals that require a Quantity.
Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\PayloadFloatType.*

Table 5.98 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
value	Payload quantity expressed as a floating point number.	Real See Clause 6.6.3.6.	[1]

5.7.1.2.1.20. PayloadLevelType (Class)
This is the Payload for Signals that convey Simple Levels.
Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\PayloadLevelType.*

Table 5.99 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
level	Payload quantity expressed as a simple level.	Integer See Clause 6.6.3.5.	[1]

5.7.1.2.1.21. PayloadPriceMultiplierType (Class)

A Price Multiplier that when applied to a reference price produces the actual price.

Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\ElClasses\PayloadPriceMultiplierType.*

Table 5.100 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
priceMultiplier	Elements of a price multiplier that when applied to a reference price produces the actual price of a quantity.	PriceMultiplierType See Clause 5.7.2.13.1.9.	[1]

5.7.1.2.1.22. PayloadPriceRelativeType (Class)

A Price Relative is added to a reference price to compute the actual price. Price Relative may be positive or negative.

Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\ElClasses\PayloadPriceRelativeType.*

Table 5.101 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
priceRelative	A price relative is added to a reference price to compute the actual price. Price relative may be positive or negative. It optionally includes a Market Context for the reference price.	PriceRelativeType See Clause 5.7.2.13.1.10.	[1]

5.7.1.2.1.23. PayloadPriceType (Class)

This is the Payload for Signals that require a Quantity.

Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\ElClasses\PayloadPriceType.*

Table 5.102 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
price	This is the number that quantifies the actual price per unit of the product.	PriceType See Clause 5.7.2.13.1.11.	[1]

5.7.1.2.1.24. PayloadProductType (Class)

This Payload contains the information that changes in each Stream payload.

Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadProductType*.

Table 5.103 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
productDescription	The terms of energy tenders and transactions for which time and location are essential characteristics.	ProductDescriptionType See Clause 5.7.2.13.1.12.	[1]

5.7.1.2.1.25. PayloadQuantityType (Class)

This is the Payload for Signals that require a Quantity.

Parent Class: PayloadBaseType (See Clause 5.7.1.2.1.17.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadQuantityType*.

Table 5.104 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
quantity	Amount, rate, or number of an item.	Real See Clause 6.6.3.6.	[1]

5.7.1.2.1.26. ReadingTypeType «Union» (Class)

Type of Reading.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ReadingTypeType*.

5.7.1.2.1.27. refID (Class)

Reference ID for a particular instance, transmittal, or artifact. Note: not the same as the native ID of the object being transmitted or shared.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\refID*.

5.7.1.2.1.28. ReportSpecifierType (Class)

Parameters that define the content of a Report Stream

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\ReportSpecifierType*.

Table 5.105 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
granularity	How frequently the measurement is to be recorded.	Duration See Clause 6.6.1.9.	[1]
marketContext	The Market Context is the overall market or program rules that govern this event.	AnyURI See Clause 6.6.3.2.	[0..1]
reportBackDuration	Report back with the Report-To-Date each the passing of this Duration during the Report Interval.	Duration See Clause 6.6.1.9.	[0..1]
reportInterval	This is the overall period of reporting.	IntervalType See Clause 5.7.3.2.7.	[0..1]
reportSpecifierID	Reference identifier for the Report Specifier for this Request. The Specifier may be known from a previous request, or may be a standard Specifier within this Market Context.	refID See Clause 5.7.1.2.1.27.	[1]
specifierPayload	Elements that indicate exactly what is to be in a report.	SpecifierPayloadType See Clause 5.7.1.2.1.31.	[1..*]

5.7.1.2.1.29. SignalTypeType «Union» (Class)

SignalType is used in EventSignals to specify the Payload Types in a Signal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\SignalTypeType*.

5.7.1.2.1.30. SimpleLevelContextType (Class)

Simple Levels are a set of simple indicators about scarcity and value, in which an ordered set of values indicate energy scarcity is above normal, normal, or below normal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\SimpleLevelContextType*.

Table 5.106 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
levelNormalValue	The "normal" level indicating normal energy availability. Levels below normal indicate surplus, levels above normal indicate increasing scarcity. If levelUpperLimit is 7, the levels are 1-7, and the levelNormalValue might be 3.	Integer See Clause 6.6.3.5.	[0..1]
levelUpperLimit	The upper level for this context. If levelUpperLimit is 5, the levels are 1-5, where 5 indicates the greatest scarcity.	Integer See Clause 6.6.3.5.	[0..1]

5.7.1.2.1.31. SpecifierPayloadType (Class)

Payload for use in Report Specifiers.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\SpecifierPayloadType*.

Table 5.107 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
itemBase	The Item Base is the core of an EMIX Product Description. Examples of an Item Base denominated value include Real Power, Real Energy, and Voltage.	ItemBaseType See Clause 6.6.4.3.	[0..1]
readingType	Defines what is being measured and reported. Measurements are in units of Item Base unless the Report Type indicates otherwise.	ReadingTypeType See Clause 5.7.1.2.1.26.	[1]
rID	Identifies this Payload. If only one Payload is requested, the rID should be omitted; if multiple Payloads are requested in the same Report, each should have a rID.	String See Clause 6.6.3.7.	[0..1]

5.7.1.2.1.32. EventStatusEnumeratedType (Enumeration)

Indicates the present status of an event.

Parent Class: EventStatusType (See Clause 5.7.1.2.1.14.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\Energy Interoperation wd36\PR03\EIClasses\EventStatusEnumeratedType*.

Table 5.108 – EventStatusEnumeratedType Enumerated Values

Enumerated Value	Definition
none	No event pending
far	event pending in the far future. The exact definition of how far in the future this refers is dependent upon the market context, but typically means the next day.
near	event pending in the near future. The exact definition of how near in the future the pending event is active is dependent on the market context
active	The event has been initiated and is active.
completed	The event has completed.
cancelled	The event has been canceled.

5.7.1.2.1.33. ReadingTypeEnumeratedType (Enumeration)

Indicates metadata about the Readings, i.e., direct measurement or computation.

Parent Class: ReadingTypeType (See Clause 5.7.1.2.1.26.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\ElClasses\ReadingTypeEnumeratedType.*

Table 5.109 - ReadingTypeEnumeratedType Enumerated Values

Enumerated Value	Definition
Direct Read	Reading is read from a device that increases monotonically, and usage must be computed from pairs of start and stop readings.
Net	Meter or [resource] prepares its own calculation of total use over time.
Allocated	Meter covers several [resources] and usage is inferred through some sort of pro rata computation.
Estimated	Used when a reading is absent in a series in which most readings are present.
Summed	Several meters together provide the reading for this [resource]. This is specifically a different than aggregated, which refers to multiple [resources] in the same payload. See also Hybrid.
Derived	Usage is inferred through knowledge of run-time, normal operation, etc.
Mean	Reading is the mean value over the period indicated in Granularity
Peak	Reading is the peak (highest) value over the period indicated in granularity. For some measurements, it may make more sense as the lowest value. It may not be consistent with aggregate readings and is only valid for flow-rate item bases, i.e., power not energy.
Hybrid	If aggregated, refers to different reading types in the aggregate number.
Contract	Indicates reading is pro forma, i.e., is reported at agreed upon rates
Projected	Indicates reading is in the future, and has not yet been measured.

5.7.1.2.1.34. SignalTypeEnumeratedType (Enumeration)

SignalTypeEnumerated lists the pre-defined Types used to specify the Payload Types and conformance in a Stream

Parent Class: SignalTypeType (See Clause 5.7.1.2.1.29.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\ElClasses\SignalTypeEnumeratedType.*

Table 5.110 – SignalTypeEnumeratedType Enumerated Values

Enumerated Value	Definition
delta	Signal indicates the amount to change (denominated in Itembase or in the EMIX Product) from what one would have used without the Signal. This may or may not be accompanied by a baseline. Payload Type Quantity
level	Signal indicates a Program Level. Payload Type is Program Level
multiplier	Signal indicates a multiplier applied to the present rate of delivery or usage (denominated in Itembase or in the EMIX Product) from what one would have used without the Signal. This may or may not be accompanied by a baseline. Payload Type is Float
price	Signal indicates the Price. Extended Price is the value multiplied by the number of units (denominated in Itembase or in the EMIX Product). Payload Type is Price
priceMultiplier	Signal indicates the Price. Extended Price is the value multiplied by the number of units (denominated in Itembase or in the EMIX Product). Payload Type is Price
product	Signal indicates the Product for each interval. Payload Type is an EMIX Product Description
setpoint	Signal indicates a target amount of units (denominated in Itembase or in the EMIX Product). Payload Type is Quantity
priceRelative	Signal indicates the Price. Extended Price is the value multiplied by the number of units (denominated in Itembase or in the EMIX Product). Payload Type is Price

5.7.2. EMIX PR04

The Energy Market Information Exchange (EMIX) Component is the portion of the FSGIM that provides an abstract representation of information needed to exchange Price and Product information for power and energy markets. Product definition includes quantity and quality of supply as well as attributes of interest to consumers distinguishing between power and energy sources.

UML Package Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04*

5.7.2.1. EMIX Price Diagram

This diagram shows how prices are modeled in EMIX.

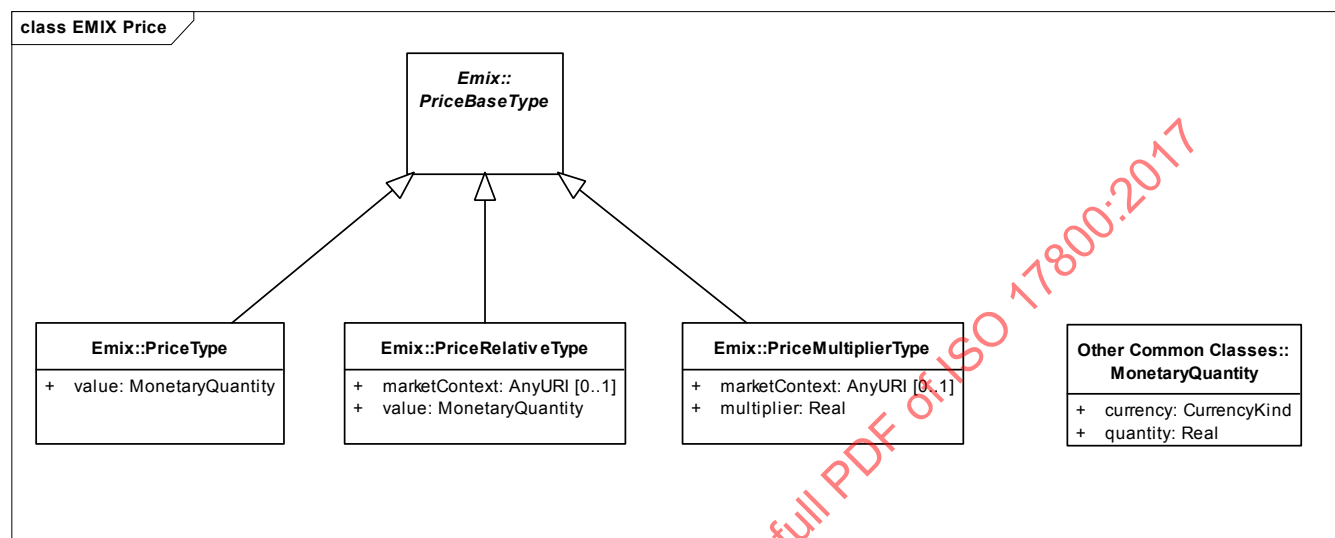


Figure 5.39 - EMIX Price

The EMIX Price Diagram makes use of the following elements:

MonetaryQuantity (Class)

A monetary quantity used to express the cost or value of something in some currency.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\MonetaryQuantity*. See Clause 6.6.5.4 for the complete definition of this element.

PriceBaseType (Abstract Class)

Type of Abstract base for EMIX Prices

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceBaseType*. See Clause 5.7.2.13.1.8 for the complete definition of this element.

PriceMultiplierType (Class)

Multiplier times market price, 1 for same as market

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceMultiplierType*. See Clause 5.7.2.13.1.9 for the complete definition of this element.

PriceRelativeType (Class)

Price Relative is a fixed charge (positive or negative) applied to base price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceRelativeType*. See Clause 5.7.2.13.1.10 for the complete definition of this element.

PriceType (Class)

Simple Price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceType*. See Clause 5.7.2.13.1.11 for the complete definition of this element.

5.7.2.2. EMIX ProductDescriptionType, LoadReductionType, GenerationType, and PowerResponseType Diagram

This diagram shows how the LoadReductionType, GenerationType, and PowerResponseType are modeled in EMIX. These three classes are the basis for the EMLoadReductionType, EMGenerationType, and EMPowerResponseType classes and represent the grid view of resources whose demand may be modified in response to signals from the grid.

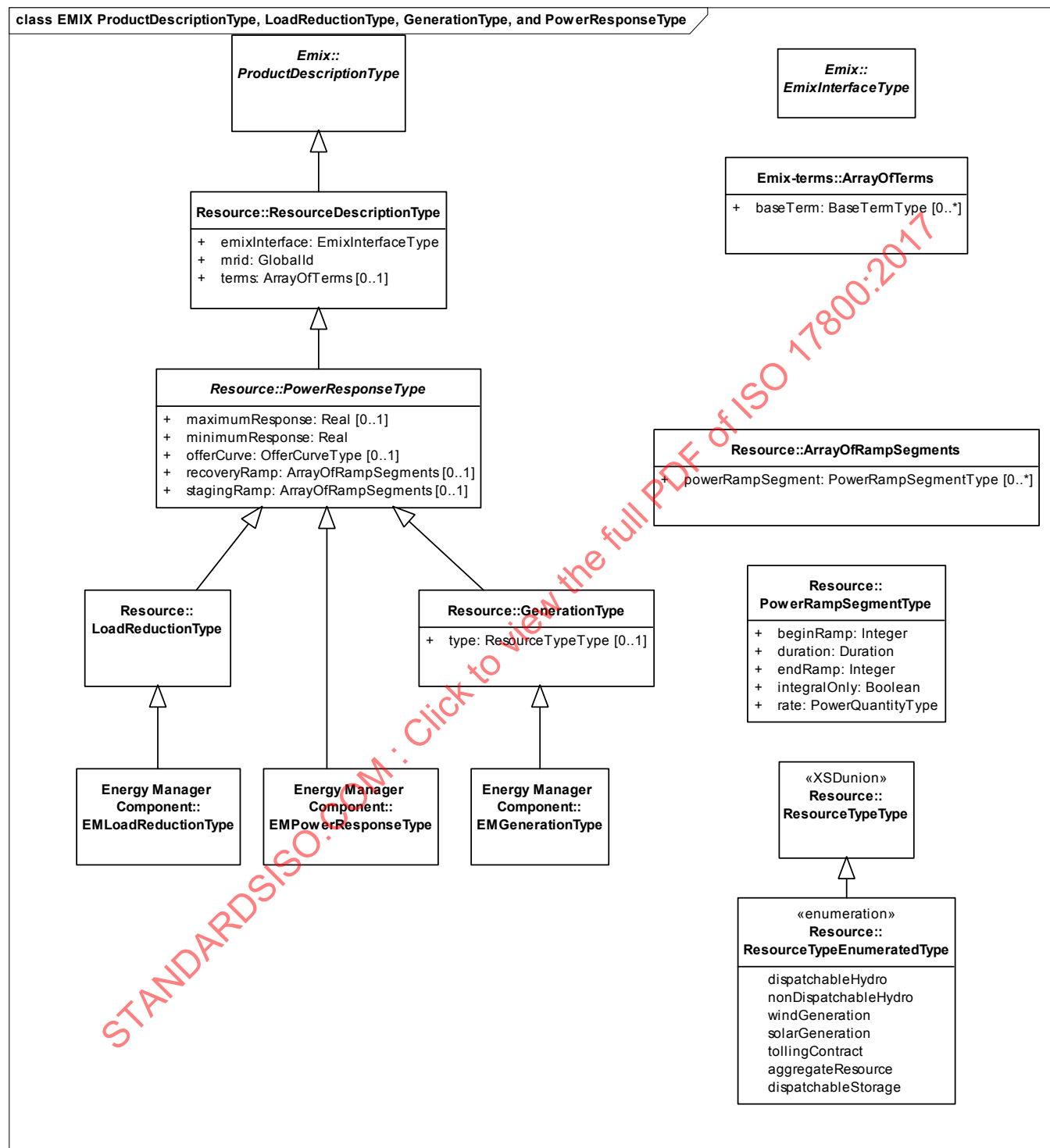


Figure 5.40 - EMIX ProductDescriptionType, LoadReductionType, GenerationType, and PowerResponseType

The EMIX ProductDescriptionType, LoadReductionType, GenerationType, and PowerResponseType Diagram makes use of the following elements:

ArrayOfRampSegments (Class)

Collection of Power Ramp Segments.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX*

PR04\Resource\ArrayOfRampSegments. See Clause 5.7.2.13.6.1 for the complete definition of this element.

ArrayOfTerms (Class)

Collection of Emix Terms

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\ArrayOfTerms*. See Clause 5.7.2.13.2.1 for the complete definition of this element.

EMGenerationType (Class)

This class represents net demand reductions resulting from supply increases. The class supports power ramps that describe the power profile for a generation resource when it is energized (stagingRamps) or de-energized (recoveryRamps).

For each array element of a generation stagingRamp, the stagingRamp[n].rate defines the amount of power increase and the stagingRamp[n].duration defines the length of time in seconds upon which this increase is in effect. If the stagingRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the generation stagingRamp attribute name implies that the demand is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

For each array element of a generator recoveryRamp, the recoveryRamp[n].rate defines the amount of power decrease and the recoveryRamp[n].duration defines the length of time in seconds upon which this decrease is in effect. If the recoveryRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the generator recoveryRamp attribute name implies that the demand reduction is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMGenerationType*. See Clause 5.6.7.3 for the complete definition of this element.

EmixInterfaceType (Abstract Class)

Abstract base class for the interfaces for EMIX Product delivery, measurement, and/or pricing.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixInterfaceType*. See Clause 5.7.2.13.1.3 for the complete definition of this element.

EMLoadReductionType (Class)

This class represents demand reductions from Loads in response to a demand response event. The class supports power ramps that describe the power profile for a resource when it is de-energized (stagingRamps) or energized (recoveryRamps).

For each array element of a stagingRamp, the stagingRamp[n].rate defines the amount of power decrease and the stagingRamp[n].duration defines the length of time in seconds upon which this decrease is in effect. If the stagingRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the stagingRamp attribute name implies that the demand reduction is monotonically decreasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be decreasing.

For each array element of a recoveryRamp, the recoveryRamp[n].rate defines the amount of power increase and the recoveryRamp[n].duration defines the length of time in seconds upon which this increase is in effect. If the recoveryRamp[n].beginRamp is defined for a ramp segment, this is the initial value of the ramp segment; if it is not present the initial value of the ramp equals the ending value of the previous ramp segment. Although the recoveryRamp attribute name implies that the demand is monotonically increasing, individual array elements may have slopes less than, greater than, or equal to 0. However, the overall trend of the function shall be increasing.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMLoadReductionType*. See Clause 5.6.7.5 for the complete definition of this element.

EMPowerResponseType (Class)

This class represents net demand reductions resulting from a combination of reductions in demand (EMLoadReductionType) and increases in supply (EMGenerationType).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPowerResponseType*. See Clause 5.6.7.6 for the complete definition of this element.

GenerationType (Class)

A Generation Resource ramps up, stays up, and then ramps down. For stagingRamps, endRamp is greater than beginRamp. For recoveryRamps, endRamp is less than beginRamp.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\GenerationType*. See Clause 5.7.2.13.6.2 for the complete definition of this element.

LoadReductionType (Class)

A Load Reduction Resource ramps down, stays down, and then ramps up. For stagingRamps, endRamp is less than beginRamp. For recoveryRamps, endRamp is greater than beginRamp.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\LoadReductionType*. See Clause 5.7.2.13.6.3 for the complete definition of this element.

PowerRampSegmentType (Class)

A Power Ramp Segment describes a change up or down in units/duration. A ramp rate holds for the duration between beginRamp to endRamp.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\PowerRampSegmentType*. See Clause 5.7.2.13.6.6 for the complete definition of this element.

PowerResponseType (Abstract Class)

Generic model describing the power response capabilities of a resource

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\PowerResponseType*. See Clause 5.7.2.13.6.7 for the complete definition of this element.

ProductDescriptionType (Abstract Class)

In EMIX, the Product Description is placed in the Interval or Gluon attachment. The respective product schemas extend this abstract class.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\ProductDescriptionType*. See Clause 5.7.2.13.1.12 for the complete definition of this element.

ResourceDescriptionType (Class)

Resource Description based on the EMIX Product Description.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\ResourceDescriptionType*. See Clause 5.7.2.13.6.8 for the complete definition of this element.

ResourceTypeEnumeratedType (Enumeration)

Resource types share common responsiveness and predictability characteristics, sometimes covarying across resources in the same class. (Example: Solar in the same region failing at the same time)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\ResourceTypeEnumeratedType*. See Clause 5.7.2.13.6.10 for the complete definition of this element.

ResourceTypeType (Class)

Abstract class for the type and dispatch ability of a resource.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\ResourceTypeType*. See Clause 5.7.2.13.6.9 for the complete definition of this element.

5.7.2.3. EMIX Terms Diagram

This diagram shows how the terms of sale for power and energy are modeled in EMIX.

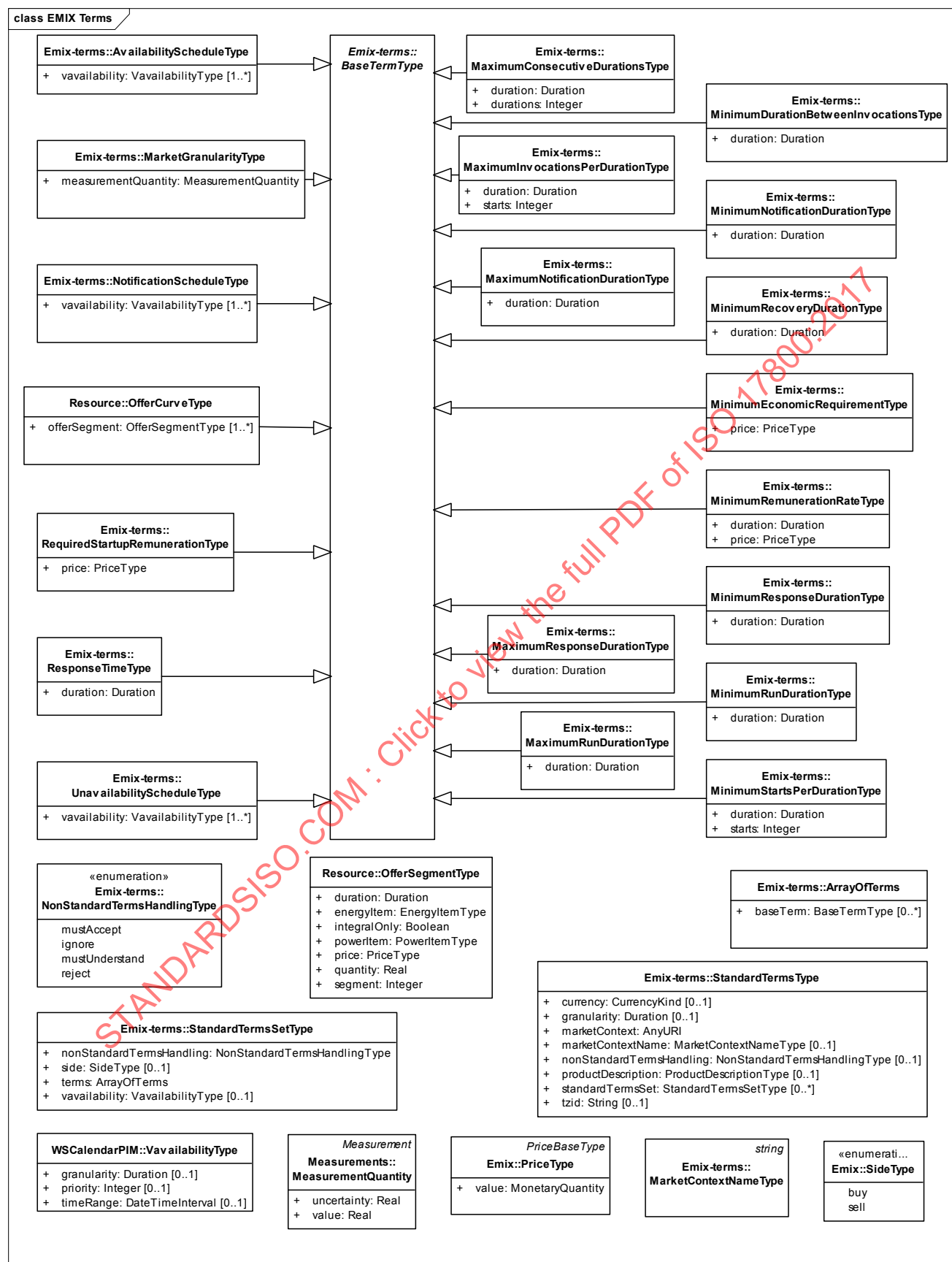


Figure 5.41 - EMIX Terms

The EMIX Terms Diagram makes use of the following elements:

ArrayOfTerms (Class)

Collection of Emix Terms

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\ArrayOfTerms*. See Clause 5.7.2.13.2.1 for the complete definition of this element.

AvailabilityScheduleType (Class)

A Schedule of time windows during which a response may be scheduled. A scheduled Duration must be entirely within a single instance of an availability window.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\AvailabilityScheduleType*. See Clause 5.7.2.13.2.2 for the complete definition of this element.

BaseTermType (Abstract Class)

Type of Abstract extension object for Emix Terms

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\BaseTermType*. See Clause 5.7.2.13.2.3 for the complete definition of this element.

MarketContextNameType (Class)

A descriptive name applied to any market context through the associated Standard Terms.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MarketContextNameType*. See Clause 5.7.2.13.2.4 for the complete definition of this element.

MarketGranularityType (Class)

Granularity defines the size of a market "bundle". For example, a Market with a granularity of 10 MW, will only accept tenders, process transactions, and pay for delivery of Power in multiples of 10 MW.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MarketGranularityType*. See Clause 5.7.2.13.2.5 for the complete definition of this element.

MaximumConsecutiveDurationsType (Class)

Maximum consecutive Durations in which a notification will be accepted, e.g., it will not accept requests on more than three consecutive days.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumConsecutiveDurationsType*. See Clause 5.7.2.13.2.6 for the complete definition of this element.

MaximumInvocationsPerDurationType (Class)

Maximum number of requests for response that will be accepted during a Duration.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumInvocationsPerDurationType*. See Clause 5.7.2.13.2.7 for the complete definition of this element.

MaximumNotificationDurationType (Class)

The maximum Duration in advance of a proposed response that a notification will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumNotificationDurationType*. See Clause 5.7.2.13.2.8 for the complete definition of this element.

MaximumResponseDurationType (Class)

The longest Duration for which a request will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumResponseDurationType*. See Clause 5.7.2.13.2.9 for the complete definition of this element.

MaximumRunDurationType (Class)

Maximum acceptable Duration for a proposed response

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumRunDurationType*. See Clause 5.7.2.13.2.10 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MinimumDurationBetweenInvocationsType (Class)

The minimum Duration between successive responses that will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumDurationBetweenInvocationsType*. See Clause 5.7.2.13.2.11 for the complete definition of this element.

MinimumEconomicRequirementType (Class)

Minimum net remuneration for any single response.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumEconomicRequirementType*. See Clause 5.7.2.13.2.12 for the complete definition of this element.

MinimumNotificationDurationType (Class)

The minimum Duration in advance of a proposed response that a notification will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumNotificationDurationType*. See Clause 5.7.2.13.2.13 for the complete definition of this element.

MinimumRecoveryDurationType (Class)

The minimum Duration after completion of a response before a new response can be begun.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRecoveryDurationType*. See Clause 5.7.2.13.2.14 for the complete definition of this element.

MinimumRemunerationRateType (Class)

Minimum remuneration acceptable per stated Duration of response, e.g., a minimum remuneration of \$100 / hour.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRemunerationRateType*. See Clause 5.7.2.13.2.15 for the complete definition of this element.

MinimumResponseDurationType (Class)

The shortest Duration for which a request will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumResponseDurationType*. See Clause 5.7.2.13.2.16 for the complete definition of this element.

MinimumRunDurationType (Class)

Minimum acceptable Duration for a proposed response.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRunDurationType*. See Clause 5.7.2.13.2.17 for the complete definition of this element.

MinimumStartsPerDurationType (Class)

The fewest requests that the resource will accept during any Duration.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumStartsPerDurationType*. See Clause 5.7.2.13.2.18 for the complete definition of this element.

NonStandardTermsHandlingType (Enumeration)

Defines how participants should handle terms not listed in the RuleSet

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\NonStandardTermsHandlingType*. See Clause 5.7.2.13.2.25 for the complete definition of this element.

NotificationScheduleType (Class)

A schedule of time windows during which requests can be made.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\NotificationScheduleType*. See Clause 5.7.2.13.2.19 for the complete definition of this element.

OfferCurveType (Class)

A collection of Offer Segments used to compute cost requirements across a range of power.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\OfferCurveType*. See Clause 5.7.2.13.6.4 for the complete definition of this element.

OfferSegmentType (Class)

Marginal offer for Power within a range. Marginal costs must be computed within the context of a range of segments as conformed by the Offer Type.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\OfferSegmentType*. See Clause 5.7.2.13.6.5 for the complete definition of this element.

PriceType (Class)

Simple Price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceType*. See Clause 5.7.2.13.1.11 for the complete definition of this element.

RequiredStartupRemunerationType (Class)

Minimum remuneration required for initiating a response.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\RequiredStartupRemunerationType*. See Clause 5.7.2.13.2.20 for the complete definition of this element.

ResponseTimeType (Class)

Duration required from receipt of a request to supplying the full requested level of response, i.e., notification time plus response time.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\ResponseTimeType*. See Clause 5.7.2.13.2.21 for the complete definition of this element.

SideType (Enumeration)

Identifies whether information originator is on the Buy or Sell side.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\SideType*. See Clause 5.7.2.13.1.15 for the complete definition of this element.

StandardTermsSetType (Class)

A collection of Standard Terms for this market

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\StandardTermsSetType*. See Clause 5.7.2.13.2.22 for the complete definition of this element.

StandardTermsType (Class)

Standard Terms express overarching terms for a market context

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\StandardTermsType*. See Clause 5.7.2.13.2.23 for the complete definition of this element.

UnavailabilityScheduleType (Class)

A schedule of time windows for which no request for response will be accepted. No part of a requested Duration must coincide with an unavailability window.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\UnavailabilityScheduleType*. See Clause 5.7.2.13.2.24 for the complete definition of this element.

VavailabilityType (Class)

The VavailabilityType class is intended to establish a baseline range of time to scope a sequence. The class can specify a start time and an end time or duration, using the timeRange attribute. If "start" is not present, then the start time is unbounded. If "end" or "duration" is not present, then the end time is unbounded. Within the specified time period, availability defaults to "not available", except for any time periods corresponding to associated AvailabilityType instances.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WScalendarPIM\VavailabilityType*. See Clause 5.7.3.2.10 for the complete definition of this element.

5.7.2.4. EMIX Warrants Diagram

This diagram shows how power quality warrants are modeled in EMIX.

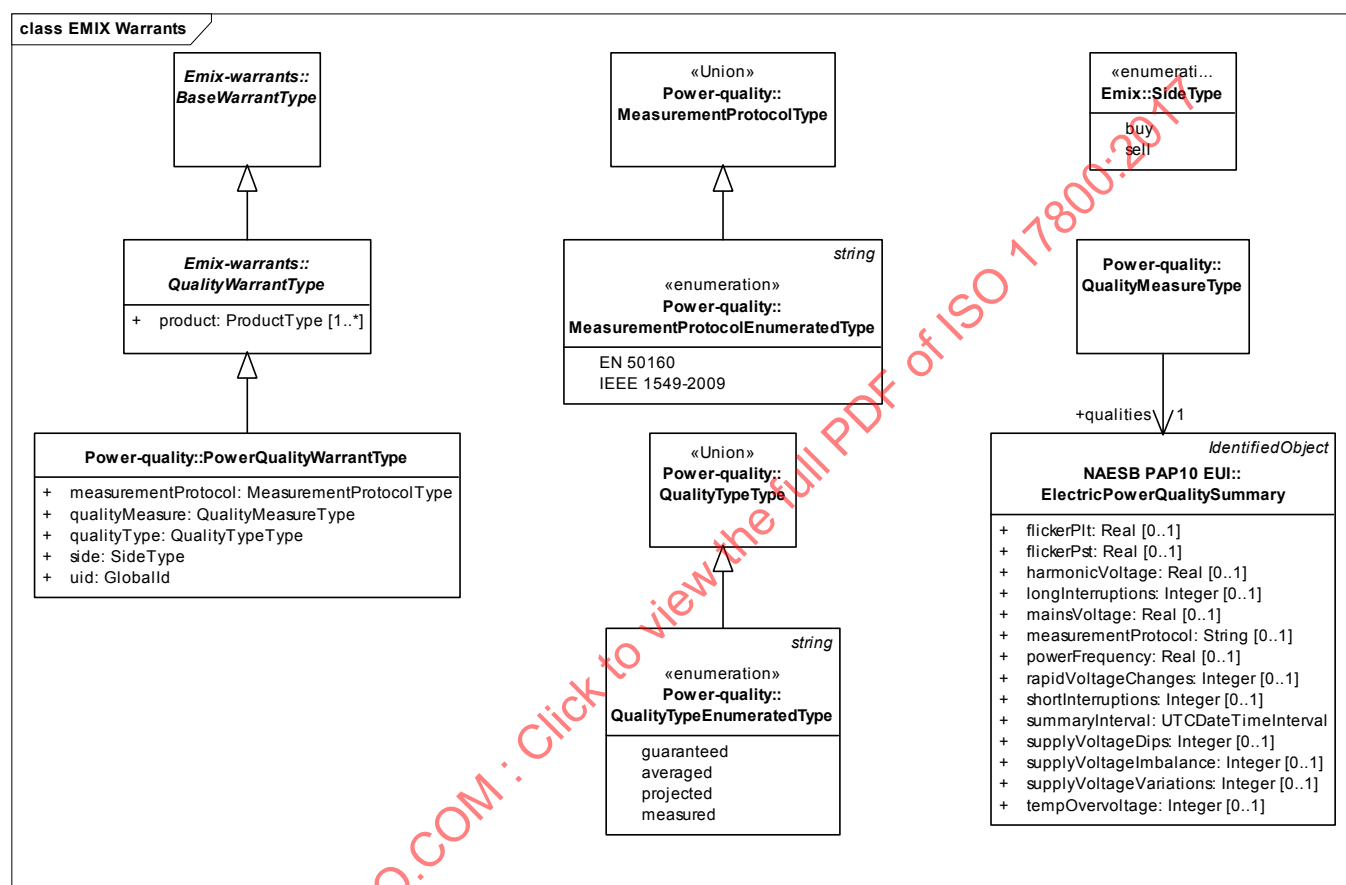


Figure 5.42 - EMIX Warrants

The EMIX Warrants Diagram makes use of the following elements:

BaseWarrantType (Abstract Class)

Type of Abstract extension object for Emix Warrants

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-warrants\BaseWarrantType*. See Clause 5.7.2.13.3.1 for the complete definition of this element.

ElectricPowerQualitySummary (Class)

A summary of power quality events. This information represents a summary of power quality information typically required by energy management systems. It is not intended to satisfy the detailed requirements of power quality monitoring. All values are as defined by measurementProtocol during the period. The standards typically also give ranges of allowed values. See referenced standards for definition, measurement protocol and period.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\ElectricPowerQualitySummary*. See Clause 5.7.5.2.1.5 for the complete definition of this element.

MeasurementProtocolEnumeratedType (Enumeration)

An identification of the standard or other protocol used to measure power quality. Sets definition for all other power attributes.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\MeasurementProtocolEnumeratedType*. See Clause 5.7.2.13.5.5 for the complete definition of this element.

MeasurementProtocolType (Class)

An identification of the standard or other protocol used to measure power quality.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\MeasurementProtocolType*. See Clause 5.7.2.13.5.1 for the complete definition of this element.

PowerQualityWarrantType (Class)

A Power Quality Warrant asserts or requires that the power be of a certain quality or better. This type extends the EMIX Quality Warrant for use in describing power. The Power Quality Warrant is a collection of the most commonly used Quality Measurements. Users who require a Quality Measure not on this list can extend the Quality Measure Type to create new Quality Measures.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\PowerQualityWarrantType*. See Clause 5.7.2.13.5.2 for the complete definition of this element.

QualityMeasureType (Class)

Quality Measure is a collection of Power Quality Indicators. The Quality Measure is placed in a WS-Calendar (OASIS WS-Calendar v1.0) Artifact so it can populate a schedule.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\QualityMeasureType*. See Clause 5.7.2.13.5.3 for the complete definition of this element.

QualityTypeEnumeratedType (Enumeration)

A string indicating the type of Assertion being made.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\QualityTypeEnumeratedType*. See Clause 5.7.2.13.5.6 for the complete definition of this element.

QualityTypeType (Class)

Abstract class for the origin of a power quality measurement.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\QualityTypeType*. See Clause 5.7.2.13.5.4 for the complete definition of this element.

QualityWarrantType (Abstract Class)

A Quality Warrant asserts or requires that the product be of a certain quality or better. A quality warrant includes an array of Quality Measures. The Quality Measure type is extensible to support the definition of additional quality measures. Numerous quality measures are defined as part of the POWER schema of EMIX; parties wishing to specify EMIX extensions for other commodities and resources will want to define their Quality Measures.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-warrants\QualityWarrantType*. See Clause 5.7.2.13.3.2 for the complete definition of this element.

SideType (Enumeration)

Identifies whether information originator is on the Buy or Sell side.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\SideType*. See Clause 5.7.2.13.1.15 for the complete definition of this element.

5.7.2.5. EmixBaseType Diagram

This diagram shows how EmixBaseType is modeled in EMIX. EmixBaseType is used to describe prices in the FSGIM.

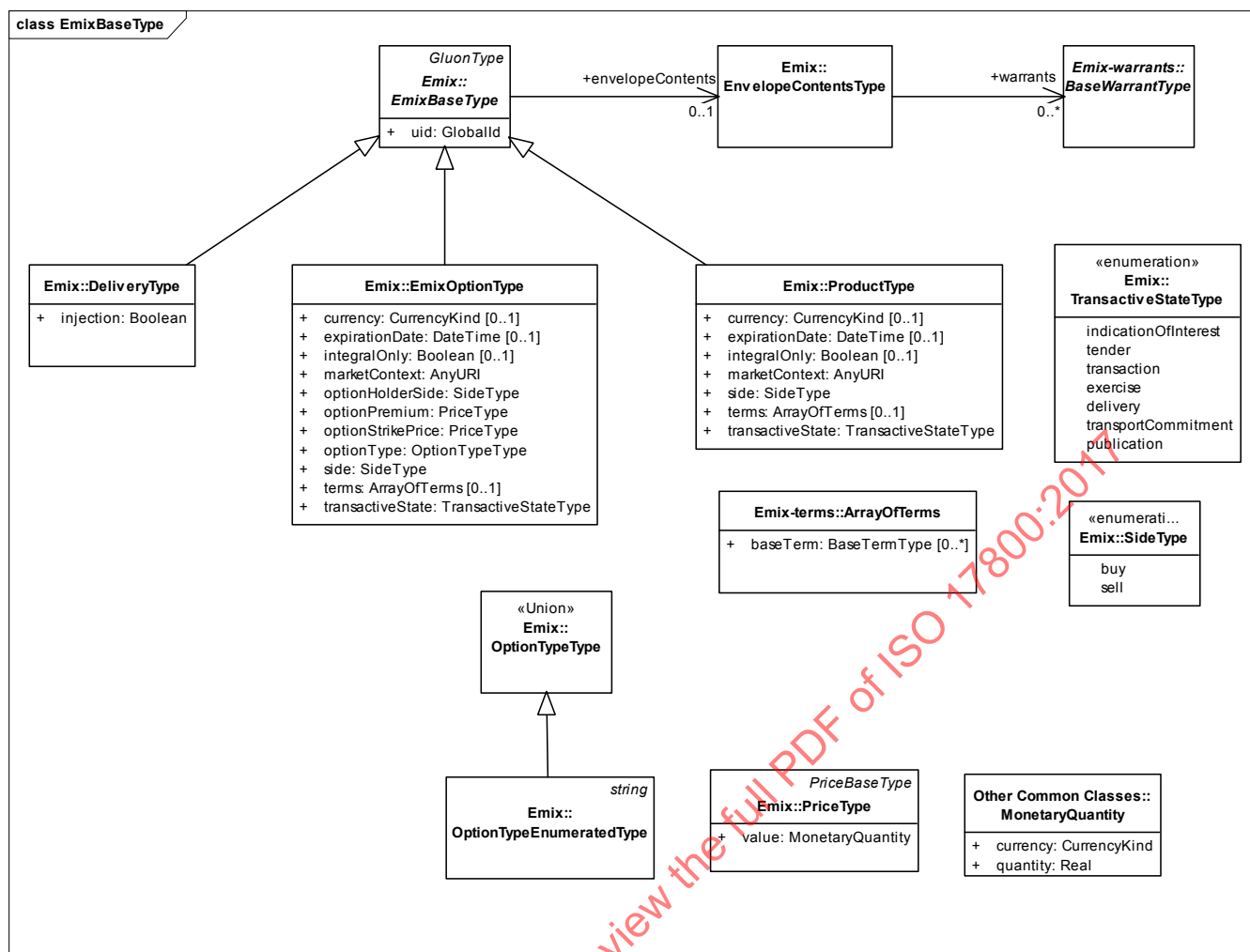


Figure 5.43 - EmixBaseType

The EmixBaseType Diagram makes use of the following elements:

ArrayOfTerms (Class)

Collection of Emix Terms

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\ArrayOfTerms*. See Clause 5.7.2.13.2.1 for the complete definition of this element.

BaseWarrantType (Abstract Class)

Type of Abstract extension object for Emix Warrants

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-warrants\BaseWarrantType*. See Clause 5.7.2.13.3.1 for the complete definition of this element.

DeliveryType (Class)

Receipt / Report of Product Delivery. Injection flag is true for adding product to market supply, false for taking from market.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\DeliveryType*. See Clause 5.7.2.13.1.1 for the complete definition of this element.

EmixBaseType (Abstract Class)

iCalendar-derived object to host EMIX (OASIS EMIX v1.0) elements.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixBaseType*. See Clause 5.7.2.13.1.2 for the complete definition of this element.

EmixOptionType (Class)

Option to buy an Emix Product.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixOptionType*. See Clause 5.7.2.13.1.4 for the complete definition of this element.

EnvelopeContentsType (Class)

EMIX Warrants are delivered as Envelope Contents. Assertions that distinguish the commodity product are called EMIX Warrants. A common definition of a warrant is a written assurance that some product or service will be provided or will meet certain specifications.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EnvelopeContentsType*. See Clause 5.7.2.13.1.5 for the complete definition of this element.

MonetaryQuantity (Class)

A monetary quantity used to express the cost or value of something in some currency.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\MonetaryQuantity*. See Clause 6.6.5.4 for the complete definition of this element.

OptionTypeEnumeratedType (Class)

Enumerated Option Types.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\OptionTypeEnumeratedType*. See Clause 5.7.2.13.1.6 for the complete definition of this element.

OptionTypeType (Class)

The EMIX Option is an elaboration of the EMIX Product. An option is an instrument that gives the buyer the right, but not the obligation, to buy or sell a product at a set price during given time windows.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\OptionTypeType*. See Clause 5.7.2.13.1.7 for the complete definition of this element.

PriceType (Class)

Simple Price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceType*. See Clause 5.7.2.13.1.11 for the complete definition of this element.

ProductType (Class)

EMIX Product Type, i.e., a Product Description applied to a Schedule

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\ProductType*. See Clause 5.7.2.13.1.13 for the complete definition of this element.

SideType (Enumeration)

Identifies whether information originator is on the Buy or Sell side.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\SideType*. See Clause 5.7.2.13.1.15 for the complete definition of this element.

TransactiveStateType (Enumeration)

Indication of the present state of the transaction.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\TransactiveStateType*. See Clause 5.7.2.13.1.16 for the complete definition of this element.

5.7.2.6. EmixInterfaceType Diagram

This diagram shows how various interfaces with the grid are modeled in EMIX.

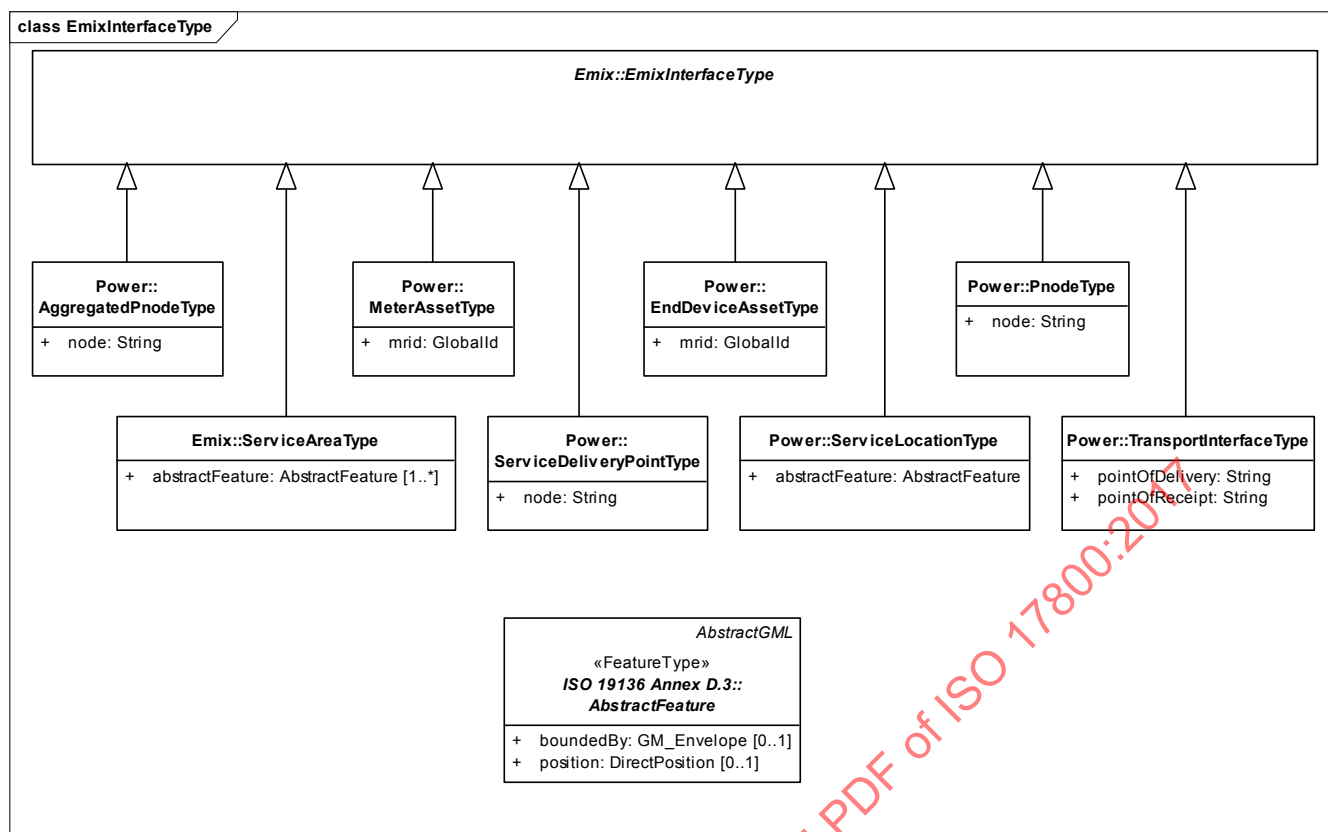


Figure 5.44 - EmixInterfaceType

The EmixInterfaceType Diagram makes use of the following elements:

AbstractFeature (Abstract Class)

Abstract class representing all classes stereotyped <<FeatureType>> i.e., classes that realize GF_FeatureType, representing features of all types from all application schemas. This class may be used as a "wildcard" when it is necessary to refer to a feature of any type.

N.B. Any class stereotyped <<FeatureType>> is implicitly substitutable for AnyFeature.

Implemented by gml:AbstractFeature (gml 3.2) or gml:_Feature (earlier versions).

UML Element Location: *Model\Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractFeature*. See Clause 5.7.6.6.3.1.1 for the complete definition of this element.

AggregatedPnodeType (Class)

An aggregated pricing node is a specialized type of pricing node used to model items such as System Zone, Default Price Zone, Custom Price Zone, Control Area, Aggregated Generation, Aggregated Participating Load, Aggregated Non-Participating Load, Trading Hub, DCA Zone

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\AggregatedPnodeType*. See Clause 5.7.2.13.4.1 for the complete definition of this element.

EmixInterfaceType (Abstract Class)

Abstract base class for the interfaces for EMIX Product delivery, measurement, and/or pricing.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixInterfaceType*. See Clause 5.7.2.13.1.3 for the complete definition of this element.

EndDeviceAssetType (Class)

The EndDeviceAssets are the physical device or devices which could be meters or other types of devices that may be of interest

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\EndDeviceAssetType*. See Clause 5.7.2.13.4.2 for the complete definition of this element.

MeterAssetType (Class)

The MeterAsset is the physical device or devices that performs the role of the meter

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\MeterAssetType*. See Clause 5.7.2.13.4.3 for the complete definition of this element.

PnodeType (Class)

A pricing node is directly associated with a connectivity node. It is a pricing location for which market participants submit their bids, offers, buy/sell CRRs, and settle.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\PnodeType*. See Clause 5.7.2.13.4.4 for the complete definition of this element.

ServiceAreaType (Class)

The Service Area is the geographic region that is affected by the EMIX market condition

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\ServiceAreaType*. See Clause 5.7.2.13.1.14 for the complete definition of this element.

ServiceDeliveryPointType (Class)

Logical point on the network where the ownership of the service changes hands. It is one of potentially many service points within a ServiceLocation, delivering service in accordance with a CustomerAgreement. Used at the place where a meter may be installed.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\ServiceDeliveryPointType*. See Clause 5.7.2.13.4.5 for the complete definition of this element.

ServiceLocationType (Class)

A customer ServiceLocation has one or more ServiceDeliveryPoint(s), which in turn relate to Meters. For distribution, the ServiceLocation is typically the location of the utility customer's premise.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\ServiceLocationType*. See Clause 5.7.2.13.4.6 for the complete definition of this element.

TransportInterfaceType (Class)

The Transport Interface delineates the edges at either end of a transport segment.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\TransportInterfaceType*. See Clause 5.7.2.13.4.7 for the complete definition of this element.

5.7.2.7. Emix

The Energy Market Information Exchange (EMIX) Component contains classes that support the ability to exchange Price and Product information for power and energy markets.

UML Package Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix*

5.7.2.8. Emix-terms

EMIX Products can be subject to a number of Terms and Market Requirements. These terms can apply at each transactive state.

UML Package Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms*

5.7.2.9. Emix-warrants

Warrants are specific assertions about the extrinsic characteristics of EMIX Products that may affect market pricing. Warrants are "a written assurances that some product or service will be provided or will meet certain specifications."

Parties may use warrants to exchange information about the source of the energy or about its environmental characteristics.

UML Package Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-warrants*

5.7.2.10. Power

The Base EMIX Product Descriptions can be extended to define a variety of power products, in particular extending the Item Base to create Items for Real Power, Apparent Power, and Reactive Power among others.

UML Package Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power*

5.7.2.11. Power-quality

Power Quality defines the electrical limits of power that does not provide any significant loss or performance or equipment life. Higher quality power can obtain a market premium. A buyer willing to accept lower quality power may be able to obtain it at lower expense. Power qualities must be measurable, discrete, and allow buyers and sellers to make choices. They must also be auditable and measurable by a specific defined protocol, so performance can be compared to promise.

UML Package Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality*

5.7.2.12. Resource

Resources include generators that can produce power and other services, storage devices that can consume, store and then produce power, and loads that produce power through load curtailment.

UML Package Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource*

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

5.7.2.13. Elements Defined in the EMIX PR04 Model

The following elements are defined in this model.

5.7.2.13.1. Elements Defined in the Emix Model

The following elements are defined in this model.

5.7.2.13.1.1. DeliveryType (Class)

Receipt / Report of Product Delivery. Injection flag is true for adding product to market supply, false for taking from market.

Parent Class: EmixBaseType (See Clause 5.7.2.13.1.2.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix\DeliveryType*.

Table 5.111 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
injection	True means positive Delivery is injection into the grid. False means positive Delivery is extraction from Grid.	Boolean See Clause 6.6.3.3.	[1]

5.7.2.13.1.2. EmixBaseType (Abstract Class)

iCalendar-derived object to host EMIX (OASIS EMIX v1.0) elements.

Parent Class: GluonType (See Clause 5.7.3.2.6.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixBaseType*.

Table 5.112 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
uid	A unique identifier for an EMIX element. Note: different markets and specifications that use EMIX may have their own rules for specifying a UID.	GlobalId See Clause 6.6.3.4.	[1]

This element has the following connections to other elements:

Table 5.113 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	envelopeContents	The extrinsic information that may distinguish the product from being a pure commodity.	EnvelopeContentsType See Clause 5.7.2.13.1.5.	[0..1]

5.7.2.13.1.3. EmixInterfaceType (Abstract Class)

Abstract base class for the interfaces for EMIX Product delivery, measurement, and/or pricing.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixInterfaceType*.

5.7.2.13.1.4. EmixOptionType (Class)

Option to buy an Emix Product.

Parent Class: EmixBaseType (See Clause 5.7.2.13.1.2.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixOptionType*.

Table 5.114 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
currency	Currency denominating this product.	CurrencyKind See Clause 6.6.2.1.	[0..1]
expirationDate	The date and time when a Tender expires. Meaningful only when the value of Transactive State is Tender.	DateTime See Clause 6.6.1.7.	[0..1]
integralOnly	If true, each Option Call must be for the full amount specified	Boolean See Clause 6.6.3.3.	[0..1]
marketContext	The Market Context is the overall market or program rules that govern this event.	AnyURI See Clause 6.6.3.2.	[1]
optionHolderSide	The side which enjoys the benefit of choosing whether to exercise the terms specified in the option.	SideType See Clause 5.7.2.13.1.15.	[1]
optionPremium	The Price paid by the Option Holder Side for the rights involved.	PriceType See Clause 5.7.2.13.1.11.	[1]
optionStrikePrice	The Price that the Option Holder Side pays to exercise the option.	PriceType See Clause 5.7.2.13.1.11.	[1]
optionType	An enumerated list of Option types.	OptionTypeType See Clause 5.7.2.13.1.7.	[1]
side	Identifies whether information originator is on the Buy or Sell side.	SideType See Clause 5.7.2.13.1.15.	[1]
terms	A collection of business and performance rules that define the product offering.	ArrayOfTerms See Clause 5.7.2.13.2.1.	[0..1]
transactiveState	Indication of the present state of the transaction.	TransactiveStateType See Clause 5.7.2.13.1.16.	[1]

5.7.2.13.1.5. **EnvelopeContentsType (Class)**

EMIX Warrants are delivered as Envelope Contents. Assertions that distinguish the commodity product are called EMIX Warrants. A common definition of a warrant is a written assurance that some product or service will be provided or will meet certain specifications.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EnvelopeContentsType*.

This element has the following connections to other elements:

Table 5.115 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	warrants	The assertions that distinguish the commodity product.	Base WarrantType See Clause 5.7.2.13.3.1.	[0..*]

5.7.2.13.1.6. **OptionTypeEnumeratedType (Class)**

Enumerated Option Types.

Parent Class: OptionTypeType (See Clause 5.7.2.13.1.7.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\OptionTypeEnumeratedType*.

5.7.2.13.1.7. **OptionTypeType «Union» (Class)**

The EMIX Option is an elaboration of the EMIX Product. An option is an instrument that gives the buyer the right, but not the obligation, to buy or sell a product at a set price during given time windows.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\OptionTypeType*.

5.7.2.13.1.8. **PriceBaseType (Abstract Class)**

Type of Abstract base for EMIX Prices

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceBaseType*.

5.7.2.13.1.9. **PriceMultiplierType (Class)**

Multiplier times market price, 1 for same as market

Parent Class: PriceBaseType (See Clause 5.7.2.13.1.8.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceMultiplierType*.

Table 5.116 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
marketContext	Market Context for base price. If blank, Market Context from hosting artifact.	AnyURI See Clause 6.6.3.2.	[0..1]
multiplier	Multiplier that when applied to a reference price produces the actual price of a quantity.	Real See Clause 6.6.3.6.	[1]

5.7.2.13.1.10. PriceRelativeType (Class)

Price Relative is a fixed charge (positive or negative) applied to base price

Parent Class: PriceBaseType (See Clause 5.7.2.13.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceRelativeType.*

Table 5.117 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
marketContext	Market Context for base price. If blank, Market Context from hosting artifact.	AnyURI See Clause 6.6.3.2.	[0..1]
value	Value added to a reference price to compute the actual price. The value may be positive or negative.	MonetaryQuantity See Clause 6.6.5.4.	[1]

5.7.2.13.1.11. PriceType (Class)

Simple Price

Parent Class: PriceBaseType (See Clause 5.7.2.13.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceType.*

Table 5.118 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
value	This is the number that quantifies the actual price per unit of the product.	MonetaryQuantity See Clause 6.6.5.4.	[1]

5.7.2.13.1.12. ProductDescriptionType (Abstract Class)

In EMIX, the Product Description is placed in the Interval or Gluon attachment. The respective product schemas extend this abstract class.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix\ProductDescriptionType*.

5.7.2.13.1.13. ProductType (Class)

EMIX Product Type, i.e., a Product Description applied to a Schedule

Parent Class: EmixBaseType (See Clause 5.7.2.13.1.2.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix\ProductType*.

Table 5.119 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
currency	Currency denominating this product.	CurrencyKind See Clause 6.6.2.1.	[0..1]
expirationDate	The date and time when a Tender expires. Meaningful only when the value of Transactive State is Tender.	DateTime See Clause 6.6.1.7.	[0..1]
integralOnly	As part of a Tender, indicates element must be accepted in full or rejected	Boolean See Clause 6.6.3.3.	[0..1]
marketContext	The Market Context is the overall market or program rules that govern this event.	AnyURI See Clause 6.6.3.2.	[1]
side	Identifies whether information originator is on the Buy or Sell side.	SideType See Clause 5.7.2.13.1.15.	[1]
terms	A collection of business and performance rules that define the product offering.	ArrayOfTerms See Clause 5.7.2.13.2.1.	[0..1]
transactiveState	Indication of the present state of the transaction.	TransactiveStateType See Clause 5.7.2.13.1.16.	[1]

5.7.2.13.1.14. ServiceAreaType (Class)

The Service Area is the geographic region that is affected by the EMIX market condition

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix\ServiceAreaType*.

Table 5.120 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
abstractFeature	The Service Area is the only Interface defined for all derived schemas. The Service Area expresses locations or geographic regions relevant to price communication. For example, a change in price for a power product could apply to all customers in an urban area. Service Areas are defined using [GML] in its simplest profile, i.e., level 0.	AbstractFeature See Clause 5.7.6.6.3.1.1.	[1..*]

5.7.2.13.1.15. SideType (Enumeration)

Identifies whether information originator is on the Buy or Sell side.

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix\SideType*.

Table 5.121 - SideType Enumerated Values

Enumerated Value	Definition
buy	A party interested in purchasing a product or service.
sell	A party offering a product or service for sale.

5.7.2.13.1.16. TransactiveStateType (Enumeration)

Indication of the present state of the transaction.

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix\TransactiveStateType*.

Table 5.122 - TransactiveStateType Enumerated Values

Enumerated Value	Definition
indicationOfInterest	The act of making interest in an offer generally known.
tender	An offer to supply a product or a service at a stated fixed price.
transaction	An instance of buying or selling a product or service.
exercise	The financial transaction specified by a contract is to be enacted immediately between the two parties, and the contract itself is terminated.
delivery	The action of delivering a product or service.
transportCommitment	The act of committing to transport a product.
publication	The action of making trade offering generally known.

5.7.2.13.2. Elements Defined in the Emix-terms Model

The following elements are defined in this model.

5.7.2.13.2.1. ArrayOfTerms (Class)

Collection of Emix Terms

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\ArrayOfTerms.*

Table 5.123 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
baseTerm	A collection of business and performance rules that define the product offering.	BaseTermType See Clause 5.7.2.13.2.3.	[0..*]

5.7.2.13.2.2. AvailabilityScheduleType (Class)

A Schedule of time windows during which a response may be scheduled. A scheduled Duration must be entirely within a single instance of an availability window.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\AvailabilityScheduleType.*

Table 5.124 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
vavailability	Availability communicates acceptable schedule times for Sequences.	VavailabilityType See Clause 5.7.3.2.10.	[1..*]

5.7.2.13.2.3. BaseTermType (Abstract Class)

Type of Abstract extension object for Emix Terms

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\BaseTermType.*

5.7.2.13.2.4. MarketContextNameType (Class)

A descriptive name applied to any market context through the associated Standard Terms.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MarketContextNameType.*

5.7.2.13.2.5. MarketGranularityType (Class)

Granularity defines the size of a market "bundle". For example, a Market with a granularity of 10 MW, will only accept tenders, process transactions, and pay for delivery of Power in multiples of 10 MW.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MarketGranularityType*.

Table 5.125 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
measurementQuantity	Amount of a measurement along with its associated metadata.	MeasurementQuantity See Clause 6.6.4.5.	[1]

5.7.2.13.2.6. MaximumConsecutiveDurationsType (Class)

Maximum consecutive Durations in which a notification will be accepted, e.g., it will not accept requests on more than three consecutive days.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumConsecutiveDurationsType*.

Table 5.126 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]
durations	Number of the durations.	Integer See Clause 6.6.3.5.	[1]

5.7.2.13.2.7. MaximumInvocationsPerDurationType (Class)

Maximum number of requests for response that will be accepted during a Duration.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumInvocationsPerDurationType*.

Table 5.127 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]
starts	Number of starts.	Integer See Clause 6.6.3.5.	[1]

5.7.2.13.2.8. MaximumNotificationDurationType (Class)

The maximum Duration in advance of a proposed response that a notification will be accepted.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumNotificationDurationType.*

Table 5.128 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.9. MaximumResponseDurationType (Class)

The longest Duration for which a request will be accepted.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumResponseDurationType.*

Table 5.129 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.10. MaximumRunDurationType (Class)

Maximum acceptable Duration for a proposed response

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumRunDurationType.*

Table 5.130 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.11. MinimumDurationBetweenInvocationsType (Class)

The minimum Duration between successive responses that will be accepted.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumDurationBetweenInvocationsType.*

Table 5.131 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.12. MinimumEconomicRequirementType (Class)

Minimum net remuneration for any single response.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumEconomicRequirementType.*

Table 5.132 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
price	Price of the quantity.	PriceType See Clause 5.7.2.13.1.11.	[1]

5.7.2.13.2.13. MinimumNotificationDurationType (Class)

The minimum Duration in advance of a proposed response that a notification will be accepted.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumNotificationDurationType.*

Table 5.133 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.14. MinimumRecoveryDurationType (Class)

The minimum Duration after completion of a response before a new response can be begun.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRecoveryDurationType.*

Table 5.134 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.15. MinimumRemunerationRateType (Class)

Minimum remuneration acceptable per stated Duration of response, e.g. a minimum remuneration of \$100 / hour.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRemunerationRateType.*

Table 5.135 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]
price	Price of the quantity.	PriceType See Clause 5.7.2.13.1.11.	[1]

5.7.2.13.2.16. MinimumResponseDurationType (Class)

The shortest Duration for which a request will be accepted.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumResponseDurationType.*

Table 5.136 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.17. MinimumRunDurationType (Class)

Minimum acceptable Duration for a proposed response.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRunDurationType*.

Table 5.137 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.18. MinimumStartsPerDurationType (Class)

The fewest requests that the resource will accept during any Duration.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumStartsPerDurationType*.

Table 5.138 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of a scheduled event.	Duration See Clause 6.6.1.9.	[1]
starts	Number of starts.	Integer See Clause 6.6.3.5.	[1]

5.7.2.13.2.19. NotificationScheduleType (Class)

A schedule of time windows during which requests can be made.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\NotificationScheduleType*.

Table 5.139 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
vavailability	Availability communicates acceptable schedule times for Sequences.	VavailabilityType See Clause 5.7.3.2.10.	[1..*]

5.7.2.13.2.20. RequiredStartupRemunerationType (Class)

Minimum remuneration required for initiating a response.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\RequiredStartupRemunerationType.*

Table 5.140 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
price	This is the number that quantifies the actual price per unit of the product.	PriceType See Clause 5.7.2.13.1.11.	[1]

5.7.2.13.2.21. ResponseTimeType (Class)

Duration required from receipt of a request to supplying the full requested level of response, i.e., notification time plus response time.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\ResponseTimeType.*

Table 5.141 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	Length of time.	Duration See Clause 6.6.1.9.	[1]

5.7.2.13.2.22. StandardTermsSetType (Class)

A collection of Standard Terms for this market

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\StandardTermsSetType.*

Table 5.142 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
nonStandardTermsHandling	A string enumeration indicating how to handle terms received that are different than those in the Market Term Set. Permissible values are: Reject (the information artifact), Ignore (the terms), Must Understand, Must Accept.	NonStandardTermsHandlingType See Clause 5.7.2.13.2.25.	[1]
side	Some terms may have different interpretation as to side. A Buyer term may indicate "meet or exceed" well a seller expressing the same term may indicate "no worse than".	SideType See Clause 5.7.2.13.1.15.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
terms	A collection of business and performance rules that define the product offering.	ArrayOfTerms See Clause 5.7.2.13.2.1.	[1]
vavailability	If present, this defines the period(s) that this RuleSet applies to, i.e., weekdays 9:00-5:00.	VavailabilityType See Clause 5.7.3.2.10.	[0..1]

5.7.2.13.2.23. StandardTermsType (Class)

Standard Terms express overarching terms for a market context

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\StandardTermsType.*

Table 5.143 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
currency	Currency for interactions. If present, becomes the default for all information models.	CurrencyKind See Clause 6.6.2.1.	[0..1]
granularity	Temporal granularity of market, i.e., a 5 minute market operates in 5 minute chunks, with a fixed offset from the beginning of availability	Duration See Clause 6.6.1.9.	[0..1]
marketContext	Terms that define the market context of this transaction.	AnyURI See Clause 6.6.3.2.	[1]
marketContextName	Human readable name of the market context.	MarketContextNameType See Clause 5.7.2.13.2.4.	[0..1]
nonStandardTermsHandling	A string enumeration indicating how to handle terms received that are different than those in the Market Term Set. Permissible values are: Reject (the information artifact), Ignore (the terms), Must Understand, Must Accept.	NonStandardTermsHandlingType See Clause 5.7.2.13.2.25.	[0..1]
productDescription	If present in Standard Terms, then it specifies a single Product Description for all products in the market.	ProductDescriptionType See Clause 5.7.2.13.1.12.	[0..1]
standardTermsSet	Common expectations outside of any single product-related artifact.	StandardTermsSetType See Clause 5.7.2.13.2.22.	[0..*]
tzid	Time Zone Database ID as per IANA https://www.iana.org/time-zones. This optional attribute can be used to retrieve the IANA database and construct the values for the other attributes of this class.	String See Clause 6.6.3.7.	[0..1]

5.7.2.13.2.24. UnavailabilityScheduleType (Class)

A schedule of time windows for which no request for response will be accepted. No part of a requested Duration must coincide with an unavailability window.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Emix-terms\UnavailabilityScheduleType.*

Table 5.144 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
vavailability	Availability communicates acceptable schedule times for Sequences.	VavailabilityType See Clause 5.7.3.2.10.	[1..*]

5.7.2.13.2.25. NonStandardTermsHandlingType (Enumeration)

Defines how participants should handle terms not listed in the RuleSet

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\NonStandardTermsHandlingType*.

Table 5.145 - NonStandardTermsHandlingType Enumerated Values

Enumerated Value	Definition
mustAccept	Participants must accept messages even if they contain rules not listed here that the party does not understand.
ignore	Participants must ignore any rule not listed
mustUnderstand	Participants must reject any message containing a rule not listed here unless it is understood.
reject	Participants in this market will reject all message containing rules not itemized in market rules.

5.7.2.13.3. Elements Defined in the Emix-warrants Model

The following elements are defined in this model.

5.7.2.13.3.1. BaseWarrantType (Abstract Class)

Type of Abstract extension object for Emix Warrants

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-warrants\BaseWarrantType*.

5.7.2.13.3.2. QualityWarrantType (Abstract Class)

A Quality Warrant asserts or requires that the product be of a certain quality or better. A quality warrant includes an array of Quality Measures. The Quality Measure type is extensible to support the definition of additional quality measures. Numerous quality measures are defined as part of the POWER schema of EMIX; parties wishing to specify EMIX extensions for other commodities and resources will want to define their Quality Measures.

Parent Class: BaseWarrantType (See Clause 5.7.2.13.3.1.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Emix-warrants\QualityWarrantType*.

Table 5.146 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
product	Type of commodity.	ProductType See Clause 5.7.2.13.1.13.	[1..*]

5.7.2.13.4. Elements Defined in the Power Model

The following elements are defined in this model.

5.7.2.13.4.1. AggregatedPnodeType (Class)

An aggregated pricing node is a specialized type of pricing node used to model items such as System Zone, Default Price Zone, Custom Price Zone, Control Area, Aggregated Generation, Aggregated Participating Load, Aggregated Non-Participating Load, Trading Hub, DCA Zone

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\AggregatedPnodeType.*

Table 5.147 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
node	A place where something changes (often ownership) or connects on the grid. It is a utility or transmission grid operator assigned designator for this point on the electrical network.	String See Clause 6.6.3.7.	[1]

5.7.2.13.4.2. EndDeviceAssetType (Class)

The EndDeviceAssets are the physical device or devices which could be meters or other types of devices that may be of interest

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\EndDeviceAssetType.*

Table 5.148 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
mrid	Master Resource Identifier that can identify a physical device that may be a Customer Meter or other types of End Devices.	GlobalId See Clause 6.6.3.4.	[1]

5.7.2.13.4.3. MeterAssetType (Class)

The MeterAsset is the physical device or devices that performs the role of the meter

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power\MeterAssetType.*

Table 5.149 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
mrid	Master Resource Identifier that can identify a physical device that may be a Customer Meter or other types of End Devices.	GlobalId See Clause 6.6.3.4.	[1]

5.7.2.13.4.4. PnodeType (Class)

A pricing node is directly associated with a connectivity node. It is a pricing location for which market participants submit their bids, offers, buy/sell CRRs, and settle.

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Power\PnodeType*.

Table 5.150 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
node	A place where something changes (often ownership) or connects on the grid. It is a utility or transmission grid operator assigned designator for this point on the electrical network.	String See Clause 6.6.3.7.	[1]

5.7.2.13.4.5. ServiceDeliveryPointType (Class)

Logical point on the network where the ownership of the service changes hands. It is one of potentially many service points within a ServiceLocation, delivering service in accordance with a CustomerAgreement. Used at the place where a meter may be installed

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Power\ServiceDeliveryPointType*.

Table 5.151 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
node	A place where something changes (often ownership) or connects on the grid. It is a utility or transmission grid operator assigned designator for this point on the electrical network.	String See Clause 6.6.3.7.	[1]

5.7.2.13.4.6. ServiceLocationType (Class)

A customer ServiceLocation has one or more ServiceDeliveryPoint(s), which in turn relate to Meters. For distribution, the ServiceLocation is typically the location of the utility customer's premise.

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Power\ServiceLocationType*.

Table 5.152 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
abstractFeature	The Service Area is the only Interface defined for all derived schemas. The Service Area expresses locations or geographic regions relevant to price communication. For example, a change in price for a power product could apply to all customers in an urban area. Service Areas are defined using [GML] in its simplest profile, i.e., level 0.	AbstractFeature See Clause 5.7.6.6.3.1.1.	[1]

5.7.2.13.4.7. TransportInterfaceType (Class)

The Transport Interface delineates the edges at either end of a transport segment.

Parent Class: EmixInterfaceType (See Clause 5.7.2.13.1.3.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Power\TransportInterfaceType*.

Table 5.153 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
pointOfDelivery	Where power exits a network or changes ownership. It is a utility or transmission grid operator assigned designator for this point on the electrical network.	String See Clause 6.6.3.7.	[1]
pointOfReceipt	Where power enters a network or changes ownership. It is a utility or transmission grid operator assigned designator for this point on the electrical network.	String See Clause 6.6.3.7.	[1]

5.7.2.13.4.8. UnitEnergyPriceType (Class)

Price per Unit of Energy, i.e., Power times Duration

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Power\UnitEnergyPriceType*.

Table 5.154 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
energyItem	Base class for the measurement of energy.	EnergyItemType See Clause 6.6.4.7.8.	[1]
priceBase	Simple, absolute, or other types of energy prices that rely on foreknowledge and computation.	PriceBaseType See Clause 5.7.2.13.1.8.	[1]

5.7.2.13.5. Elements Defined in the Power-quality Model

The following elements are defined in this model.

5.7.2.13.5.1. MeasurementProtocolType «Union» (Class)

An identification of the standard or other protocol used to measure power quality.

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Power-quality\MeasurementProtocolType*.

5.7.2.13.5.2. PowerQualityWarrantType (Class)

A Power Quality Warrant asserts or requires that the power be of a certain quality or better. This type extends the EMIX Quality Warrant for use in describing power. The Power Quality Warrant is a collection of the most commonly used Quality Measurements. Users who require a Quality Measure not on this list can extend the Quality Measure Type to create new Quality Measures.

Parent Class: QualityWarrantType (See Clause 5.7.2.13.3.2).

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Power-quality\PowerQualityWarrantType*.

Table 5.155 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
measurementProtocol	An identification of the standard or other protocol used to measure power quality.	MeasurementProtocolType See Clause 5.7.2.13.5.1.	[1]
qualityMeasure	A collection of power quality indicators.	QualityMeasureType See Clause 5.7.2.13.5.3.	[1]
qualityType	The origin of a power quality measurement.	QualityTypeType See Clause 5.7.2.13.5.4.	[1]
side	Identifies whether information originator is on the Buy or Sell side of the transaction.	SideType See Clause 5.7.2.13.1.15.	[1]
uid	Unique Identifier between transacting parties.	GlobalId See Clause 6.6.3.4.	[1]

5.7.2.13.5.3. QualityMeasureType (Class)

Quality Measure is a collection of Power Quality Indicators. The Quality Measure is placed in a WS-Calendar (OASIS WS-Calendar v1.0) Artifact so it can populate a schedule.

UML Element Location: *Model\UML Elements from External Sources\OASIS\EMIX PR04\Power-quality\QualityMeasureType*.

This element has the following connections to other elements:

Table 5.156 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	qualities	A collection of power quality indicators.	ElectricPowerQualitySummary See Clause 5.7.5.2.1.5.	[1]

5.7.2.13.5.4. QualityTypeType «Union» (Class)

Abstract class for the origin of a power quality measurement.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\QualityTypeType*.

5.7.2.13.5.5. MeasurementProtocolEnumeratedType (Enumeration)

An identification of the standard or other protocol used to measure power quality. Sets definition for all other power attributes.

Parent Class: MeasurementProtocolType (See Clause 5.7.2.13.5.1.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\MeasurementProtocolEnumeratedType*.

Table 5.157 - MeasurementProtocolEnumeratedType Enumerated Values

Enumerated Value	Definition
EN 50160	EN50160 means of measuring power quality.
IEEE 1549-2009	IEEE 1549 means of measuring power quality.

5.7.2.13.5.6. QualityTypeEnumeratedType (Enumeration)

A string indicating the type of Assertion being made.

Parent Class: QualityTypeType (See Clause 5.7.2.13.5.4.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Power-quality\QualityTypeEnumeratedType*.

Table 5.158 - QualityTypeEnumeratedType Enumerated Values

Enumerated Value	Definition
guaranteed	Formal assurance or promise, that certain quality conditions shall be fulfilled relating to a product, service, or transaction.
averaged	A quality number that typifies a set of quality numbers.
projected	Estimate or forecast of quality based upon trends.
measured	Quality as ascertained by comparison with a standard.

5.7.2.13.6. Elements Defined in the Resource Model

The following elements are defined in this model.

5.7.2.13.6.1. ArrayOfRampSegments (Class)

Collection of Power Ramp Segments.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\ArrayOfRampSegments*.

Table 5.159 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
powerRampSegment	Characteristics of a change up or down in units/duration during a response.	PowerRampSegmentType See Clause 5.7.2.13.6.6.	[0..*]

5.7.2.13.6.2. GenerationType (Class)

A Generation Resource ramps up, stays up, and then ramps down. For stagingRamps, endRamp is greater than beginRamp. For recoveryRamps, endRamp is less than beginRamp.

Parent Class: PowerResponseType (See Clause 5.7.2.13.6.7.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\GenerationType*.

Table 5.160 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
type	Type and dispatch-ability of generation resource.	ResourceType See Clause 5.7.2.13.6.9.	[0..1]

5.7.2.13.6.3. LoadReductionType (Class)

A Load Reduction Resource ramps down, stays down, and then ramps up. For stagingRamps, endRamp is less than beginRamp. For recoveryRamps, endRamp is greater than beginRamp.

Parent Class: PowerResponseType (See Clause 5.7.2.13.6.7.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\LoadReductionType*.

5.7.2.13.6.4. OfferCurveType (Class)

A collection of Offer Segments used to compute cost requirements across a range of power.

Parent Class: BaseTermType (See Clause 5.7.2.13.2.3.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\OfferCurveType*.

Table 5.161 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
offerSegment	The offer price for the quantity offered in a segment.	OfferSegmentType See Clause 5.7.2.13.6.5.	[1..*]

5.7.2.13.6.5. OfferSegmentType (Class)

Marginal offer for Power within a range. Marginal costs must be computed within the context of a range of segments as conformed by the Offer Type.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\OfferSegmentType*.

Table 5.162 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	Duration of this Segment.	Duration See Clause 6.6.1.9.	[1]
energyItem	Description of energy item being offered. (e.g., Real energy, measured in Watt Hours)	EnergyItemType See Clause 6.6.4.7.8.	[1]
integralOnly	If true, offer is all or none; no partial acceptance of this segment.	Boolean See Clause 6.6.3.3.	[1]
powerItem	Description of power item being offered. (e.g., Reactive power, measured in volt-amperes reactive)	PowerItemType See Clause 6.6.4.6.10.	[1]
price	Energy Price for this Segment.	PriceType See Clause 5.7.2.13.1.11.	[1]
quantity	Power Quantity for this Segment.	Real See Clause 6.6.3.6.	[1]
segment	Numeric identifier of this segment	Integer See Clause 6.6.3.5.	[1]

5.7.2.13.6.6. PowerRampSegmentType (Class)

A Power Ramp Segment describes a change up or down in units/duration. A ramp rate holds for the duration between beginRamp to endRamp.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\PowerRampSegmentType*.

Table 5.163 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
beginRamp	Quantity of power at the start of a ramp segment.	Integer See Clause 6.6.3.5.	[1]
duration	the time period component of a ratio specifying the maximum rate of change of power for this segment.	Duration See Clause 6.6.1.9.	[1]
endRamp	Quantity of power at the end of a ramp segment.	Integer See Clause 6.6.3.5.	[1]
integralOnly	If true, offer is all or none; no partial acceptance of this segment.	Boolean See Clause 6.6.3.3.	[1]
rate	the power component of a ratio specifying the maximum rate of change of power for this segment.	PowerQuantityType See Clause 6.6.4.6.12.	[1]

5.7.2.13.6.7. PowerResponseType (Abstract Class)

Generic model describing the power response capabilities of a resource

Parent Class: ResourceDescriptionType (See Clause 5.7.2.13.6.8.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX\PR04\Resource\PowerResponseType*.

Table 5.164 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
maximumResponse	The greatest Response for which this resource will accept a request.	Real See Clause 6.6.3.6.	[0..1]
minimumResponse	The least Response for which this resource will accept a request.	Real See Clause 6.6.3.6.	[1]
offerCurve	A collection of Offer Segments that define the offer price (as expressed in EMIX Requirements) for the quantity offered in each segment.	OfferCurveType See Clause 5.7.2.13.6.4.	[0..1]
recoveryRamp	An array Power Ramp Segments describing how a Resource's returns to its original state following a response.	ArrayOfRampSegments See Clause 5.7.2.13.6.1.	[0..1]
stagingRamp	Characteristics describing a resource's ability to change level at the initiation of a response.	ArrayOfRampSegments See Clause 5.7.2.13.6.1.	[0..1]

5.7.2.13.6.8. ResourceDescriptionType (Class)

Resource Description based on the EMIX Product Description.

Parent Class: ProductDescriptionType (See Clause 5.7.2.13.1.12.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Resource\ResourceDescriptionType*.

Table 5.165 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
emixInterface	The interfaces for EMIX Product delivery, measurement, and/or pricing.	EmixInterfaceType See Clause 5.7.2.13.1.3.	[1]
mrId	Master Resource Identifier that identifies a physical device that may be a Customer Meter or other types of End Devices.	GlobalId See Clause 6.6.3.4.	[1]
terms	A collection of business and performance rules that define the product offering.	ArrayOfTerms See Clause 5.7.2.13.2.1.	[0..1]

5.7.2.13.6.9. ResourceTypeType «XSDunion» (Class)

Abstract class for the type and dispatch ability of a resource.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Resource\ResourceTypeType*.

5.7.2.13.6.10. ResourceTypeEnumeratedType (Enumeration)

Resource types share common responsiveness and predictability characteristics, sometimes covarying across resources in the same class. (Example: Solar in the same region failing at the same time)

Parent Class: ResourceTypeType (See Clause 5.7.2.13.6.9.)

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\EMIX PR04\Resource\ResourceTypeEnumeratedType*.

Table 5.166 - ResourceTypeEnumeratedType Enumerated Values

Enumerated Value	Definition
dispatchableHydro	Hydro-based electrical generators dispatchable by power grid operators.
nonDispatchableHydro	Hydro-based electrical generators that are not dispatchable by power grid operators.
windGeneration	Electrical power generation created from wind.
solarGeneration	Electrical power generation created from solar photovoltaic cells.
tollingContract	A take-or-pay contract for conversion or transportation of energy.
aggregateResource	The characteristics of the summation of all the energy resources of an entity.
dispatchableStorage	Energy storage dispatchable by power grid operators.

5.7.3. WSCalendarPIM

This is a tailored version of the WSCalendar PIM for use in FSGIM.

A PIM is a model of a system designed to be independent of the details needed for particular implementation technologies.

UML Package Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM*

5.7.3.1. WSCalendarPIM Diagram

This diagram is derived from a platform independent model (PIM) of WS-Calendar (OASIS WS-Calendar v1.0).

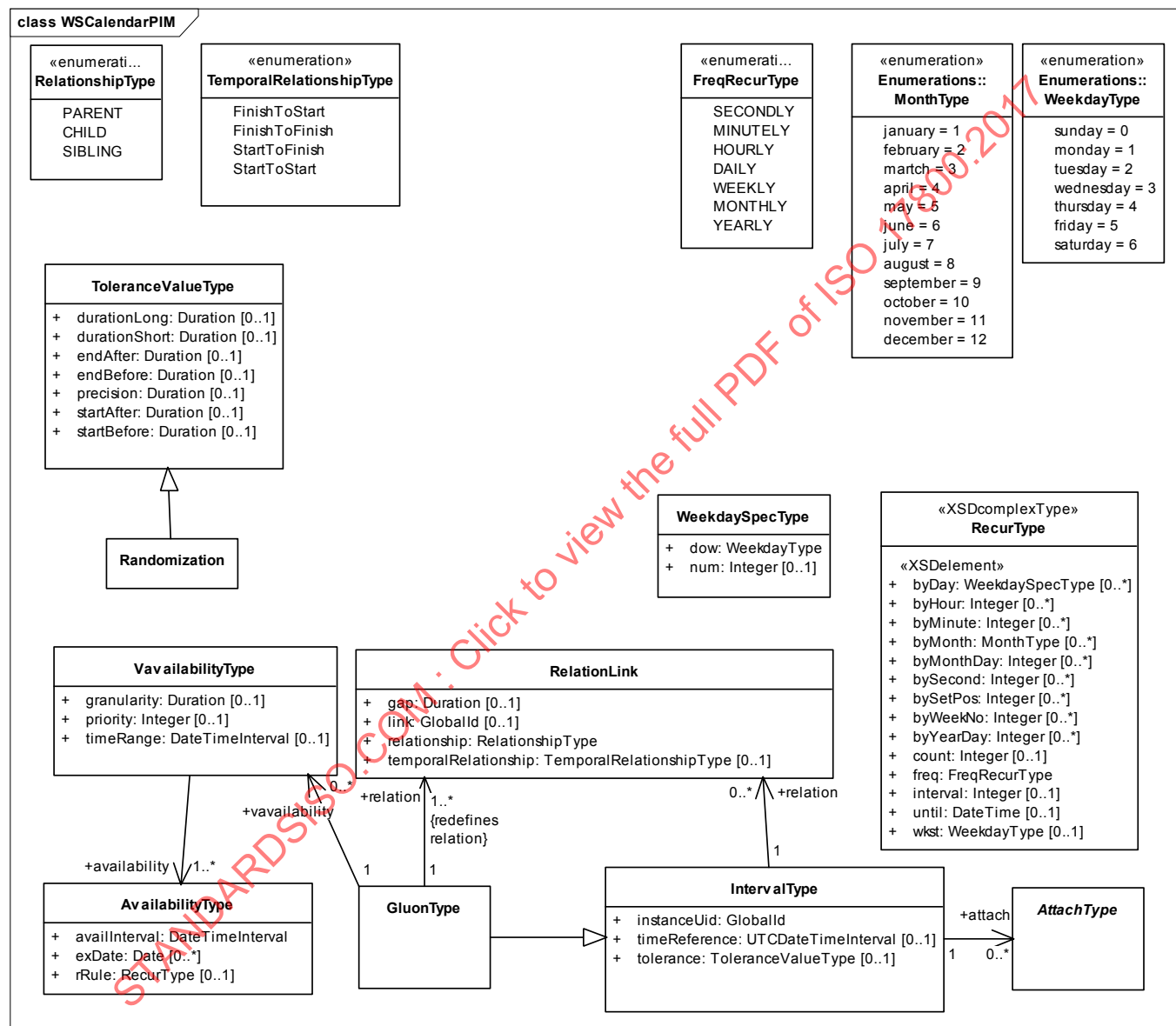


Figure 5.45 - WSCalendarPIM

The WSCalendarPIM Diagram makes use of the following elements:

AttachType (Abstract Class)

This class represents data in the domain of concern over some interval of time.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\AttachType*. See Clause 5.7.3.2.4 for the complete definition of this element.

AvailabilityType (Class)

This class expresses the range of times in which an Interval or Sequence can be Scheduled.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\AvailabilityType*. See Clause 5.7.3.2.5 for the complete definition of this element.

FreqRecurType (Enumeration)

Used to identify the unit of time to be used in specifying the frequency of a recurring event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\FreqRecurType*. See Clause 5.7.3.2.11 for the complete definition of this element.

GluonType (Class)

A GluonType influences the serialization of IntervalTypes in a sequence. A GluonType is often used as a pointer to the beginning interval in a sequence, although in general it can point to any interval in a sequence, or simply describe an isolated interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\GluonType*. See Clause 5.7.3.2.6 for the complete definition of this element.

IntervalType (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\IntervalType*. See Clause 5.7.3.2.7 for the complete definition of this element.

MonthType (Enumeration)

An enumerated list of months of the year.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\MonthType*. See Clause 6.6.2.2 for the complete definition of this element.

Randomization (Class)

A specialized version of ToleranceValueType used to indicate the degree of randomization required. In this use, the startafter and startbefore attributes indicate the range of the starting times.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\Randomization*. See Clause 5.7.3.2.1 for the complete definition of this element.

RecurType (Class)

This value type is used to identify properties that contain a recurrence rule specification.

From RFC5545 section 3.3.10. Recurrence Rule.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\RecurType*. See Clause 5.7.3.2.2 for the complete definition of this element.

RelationLink (Class)

The RelationLink class serves to link a set of intervals together, in support of a linked-list method of creating a sequence. It is also used to link a GluonType instance to a specific interval within a sequence, typically used to point to the beginning interval of a sequence.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\RelationLink*. See Clause 5.7.3.2.8 for the complete definition of this element.

RelationshipType (Enumeration)

Specifies the type of hierarchical relationship associated with the calendar component specified by the property. The parameter value can be PARENT, to indicate that the referenced calendar component is a superior of calendar component; CHILD to indicate that the referenced calendar component is a subordinate of the calendar component; or SIBLING to indicate that the referenced calendar component is a peer of the calendar component. If this parameter is not specified on an allowable property, the default relationship type is PARENT.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\RelationshipType*. See Clause 5.7.3.2.12 for the complete definition of this element.

TemporalRelationshipType (Enumeration)

The temporal relation of one interval to the interval it is linked to.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\TemporalRelationshipType*. See Clause 5.7.3.2.13 for the complete definition of this element.

ToleranceValueType (Class)

A tolerance value is a set of durations which indicate the allowed tolerance for the indicated value. Note that there may be many tolerance values associated with a Duration.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\ToleranceValueType*. See Clause 5.7.3.2.9 for the complete definition of this element.

VavailabilityType (Class)

The VavailabilityType class is intended to establish a baseline range of time to scope a sequence. The class can specify a start time and an end time or duration, using the timeRange attribute. If "start" is not present, then the start time is unbounded. If "end" or "duration" is not present, then the end time is unbounded. Within the specified time period, availability defaults to "not available", except for any time periods corresponding to associated AvailabilityType instances.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\VavailabilityType*. See Clause 5.7.3.2.10 for the complete definition of this element.

WeekdaySpecType (Class)

Specifies a week day and a recurrence number. For example, within a MONTHLY rule:

+1,MO represents the first Monday within the month, whereas -1,MO represents the last Monday of the month.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\WeekdaySpecType*. See Clause 5.7.3.2.3 for the complete definition of this element.

WeekdayType (Enumeration)

An enumerated list of day of the week.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\WeekdayType*. See Clause 6.6.2.4 for the complete definition of this element.

5.7.3.2. Elements Defined in the WSCalendarPIM Model

The following elements are defined in this model.

5.7.3.2.1. Randomization (Class)

A specialized version of ToleranceValueType used to indicate the degree of randomization required. In this use, the startafter and startbefore attributes indicate the range of the starting times.

Parent Class: ToleranceValueType (See Clause 5.7.3.2.9.)

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\Randomization.*

5.7.3.2.2. RecurType «XSDcomplexType» (Class)

This value type is used to identify properties that contain a recurrence rule specification.

From RFC5545 section 3.3.10. Recurrence Rule.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\RecurType.*

Table 5.167 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
freq	The freq rule part identifies the type of recurrence rule Valid values include SECONDLY, to specify repeating events based on an interval of a second or more; MINUTELY, to specify repeating events based on an interval of a minute or more; HOURLY, to specify repeating events based on an interval of an hour or more; DAILY, to specify repeating events based on an interval of a day or more; WEEKLY, to specify repeating events based on an interval of a week or more; MONTHLY, to specify repeating events based on an interval of a month or more; and YEARLY, to specify repeating events based on an interval of a year or more. The until rule part defines a date/time value that bounds the recurrence rule in an inclusive manner. If the value specified by until is synchronized with the specified recurrence, this date/time becomes the last instance of the recurrence.	FreqRecurType See Clause 5.7.3.2.11.	[1]
until	Only one of until or count may be used in a single RecurType instance. Constraint: The choice of either LocalDateTime or UTCDateTime as the datatype shall be consistent with the choice made within the associated AvailabilityType.DateTimeInterval and AvailabilityType.DateTimeInterval. The COUNT rule part defines the number of occurrences at which to range-bound the recurrence.	DateTime See Clause 6.6.1.7.	[0..1]
count		Integer See Clause 6.6.3.5.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
interval	Only one of until or count may be used in a single RecurType instance. The interval rule part contains a positive integer representing at which intervals the recurrence rule repeats. The default value is "1", meaning every second for a SECONDLY rule, every minute for a MINUTELY rule, every hour for an HOURLY rule, every day for a DAILY rule, every week for a WEEKLY rule, every month for a MONTHLY rule, and every year for a YEARLY rule. For example, within a DAILY rule, a value of "8" means every eight days.	Integer See Clause 6.6.3.5.	[0..1]
bySecond	Specifies a sequence of seconds within a minute -- 0..60.	Integer See Clause 6.6.3.5.	[0..*]
byMinute	Specifies a sequence of minutes within an hour -- 0..59.	Integer See Clause 6.6.3.5.	[0..*]
byHour	Specifies a sequence of hours within a day -- 0..23.	Integer See Clause 6.6.3.5.	[0..*]
byDay	A sequence of weekdays.	WeekdaySpecType See Clause 5.7.3.2.3.	[0..*]
byMonthDay	Specifies a sequence of day numbers within a month. Valid values are 1 to 31 or -31 to -1. For example, -10 represents the tenth to the last day of the month.	Integer See Clause 6.6.3.5.	[0..*]
byYearDay	Specifies a sequence of day numbers within a year. Valid values are 1 to 3366 or -366 to -1. For example, -1 represents the last day of the year.	Integer See Clause 6.6.3.5.	[0..*]
byWeekNo	A sequence of week-numbers of a year, based on the definitions in ISO.8601.2004.	Integer See Clause 6.6.3.5.	[0..*]
byMonth	A sequence of month numbers in a year -- 1..12.	MonthType See Clause 6.6.2.2.	[0..*]
bySetPos	The bysetpos rule part specifies a sequence of values that corresponds to the nth occurrence within the set of recurrence instances specified by the RecurType instance.	Integer See Clause 6.6.3.5.	[0..*]
wkst	Specifies the day on which the workweek starts.	WeekdayType See Clause 6.6.2.4.	[0..1]

5.7.3.2.3. WeekdaySpecType (Class)

Specifies a week day and a recurrence number. For example, within a MONTHLY rule:

+1,MO represents the first Monday within the month, whereas -1,MO represents the last Monday of the month.

UML Element Location: *Model\UML Elements from External Sources\OASIS\WCalendarPIM\WeekdaySpecType*.

Table 5.168 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
dow	Specifies a day of the week.	WeekdayType See Clause 6.6.2.4.	[1]
num	If present, this indicates the nth occurrence of a specific day within the MONTHLY or YEARLY RecurType.	Integer See Clause 6.6.3.5.	[0..1]

5.7.3.2.4. AttachType (Abstract Class)

This class represents data in the domain of concern over some interval of time.

UML Element Location: *Model\UML Elements from External Sources\OASIS\WCalendarPIM\AttachType.*

5.7.3.2.5. AvailabilityType (Class)

This class expresses the range of times in which an Interval or Sequence can be Scheduled.

UML Element Location: *Model\UML Elements from External Sources\OASIS\WCalendarPIM\AvailabilityType.*

Table 5.169 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
rRule	A recurrence rule specification as defined in RFC5545, (2009) Section 3.3.10.	RecurType See Clause 5.7.3.2.2.	[0..1]
exDate	A set of one or more exception dates that override the sequence specified in the rRule attribute, derived from RFC5545, (2009) Section 3.8.5.1	Date See Clause 6.6.1.8.	[0..*]
availInterval	This serves as the template "anchoring" interval that the recurrence rule (stored in rRule) is evaluated against. At instantiation time the choice must be made whether to express the interval as a UTCDateTimeInterval or a LocalDateTimeInterval.	DateTimeInterval See Clause 6.6.1.1.	[1]

5.7.3.2.6. GluonType (Class)

A GluonType influences the serialization of IntervalTypes in a sequence. A GluonType is often used as a pointer to the beginning interval in a sequence, although in general it can point to any interval in a sequence, or simply describe an isolated interval.

Parent Class: IntervalType (See Clause 5.7.3.2.7.)

UML Element Location: *Model\UML Elements from External Sources\OASIS\WCalendarPIM\GluonType.*

This element has the following connections to other elements:

Table 5.170 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	relation	This role links a GluonType instance to a specific interval within a sequence (typically the beginning interval). The target RelationLink instance would have a RelationLink.relationship value of CHILD, and would identify the interval with the RelationLink.link attribute. (This association redefines the relation association between the IntervalType Class (see Clause 5.7.3.2.7) and the RelationLink Class (see Clause 5.7.3.2.8).)	RelationLink See Clause 5.7.3.2.8.	[1..*]
Association	vavailability	The specification of the time span and pattern of intervals that defines a sequence.	VavailabilityType See Clause 5.7.3.2.10.	[0..*]

5.7.3.2.7. IntervalType (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\W3C\CalendarPIM\IntervalType*.

Table 5.171 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
instanceUid	Unique identifier to the interval instance.	GlobalId See Clause 6.6.3.4.	[1]
timeReference	The time reference for an interval which includes some combination of start, duration, and end.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]
tolerance	The tolerance for the associated timeReference.	ToleranceValueType See Clause 5.7.3.2.9.	[0..1]

This element has the following connections to other elements:

Table 5.172 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	attach	Associates a set of AttachType data with this IntervalType.	AttachType See Clause 5.7.3.2.4.	[0..*]
Association	relation	Describes the relationships between intervals.	RelationLink See Clause 5.7.3.2.8.	[0..*]

5.7.3.2.8. RelationLink (Class)

The RelationLink class serves to link a set of intervals together, in support of a linked-list method of creating a sequence. It is also used to link a GluonType instance to a specific interval within a sequence, typically used to point to the beginning interval of a sequence.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\WCalendarPIM\RelationLink*.

Table 5.173 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
gap	An optional attribute to specify a gap in time between one interval and the next. The default interpretation is a value of zero.	Duration See Clause 6.6.1.9.	[0..1]
link	The unique identifier of the interval being linked to.	GlobalId See Clause 6.6.3.4.	[0..1]
relationship	The kind of link (see the RelationshipType enumeration).	RelationshipType See Clause 5.7.3.2.12.	[1]
temporalRelationship	The specification of the temporal relation of one interval to the interval being linked to. By default, this will be FinishToStart, meaning the next interval immediately begins after the previous interval.	TemporalRelationshipType See Clause 5.7.3.2.13.	[0..1]

5.7.3.2.9. ToleranceValueType (Class)

A tolerance value is a set of durations which indicate the allowed tolerance for the indicated value. Note that there may be many tolerance values associated with a Duration.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\WCalendarPIM\ToleranceValueType*.

Table 5.174 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
durationLong	Indicates by how much the performance Duration may exceed the Duration specified in the information exchange. Duration Long Tolerance SHALL NOT be used when Start and End Tolerances are both specified.	Duration See Clause 6.6.1.9.	[0..1]
durationShort	Indicates by how much the performance Duration may fall short of the Duration specified in the information exchange. Duration Short Tolerance SHALL NOT be used when Start and End Tolerances are both specified.	Duration See Clause 6.6.1.9.	[0..1]
endAfter	Indicates how far after the scheduled end time the requested service may commence, for example, if a service that ends at 2:01 is compliant with a request to end at 2:00	Duration See Clause 6.6.1.9.	[0..1]
endBefore	Indicates how far before scheduled end time may end, for example, if a service that ends at 1:57 is compliant with a request to end at 2:00	Duration See Clause 6.6.1.9.	[0..1]
precision	Whatever the time tolerance above, there is some minimum time that is considered significant. When used in Tolerance, precision defines the tracking and reporting requirements for a service.	Duration See Clause 6.6.1.9.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
startAfter	Indicates how far after the requested start time the requested service may commence, for example, if a service that begins at 2:01 is compliant with a request to start at 2:00	Duration See Clause 6.6.1.9.	[0..1]
startBefore	Indicates how far before the requested start time the requested service may commence, for example if a service that begins at 1:57 is compliant with a request to start at 2:00	Duration See Clause 6.6.1.9.	[0..1]

5.7.3.2.10. VavailabilityType (Class)

The VavailabilityType class is intended to establish a baseline range of time to scope a sequence. The class can specify a start time and an end time or duration, using the timeRange attribute. If "start" is not present, then the start time is unbounded. If "end" or "duration" is not present, then the end time is unbounded. Within the specified time period, availability defaults to "not available", except for any time periods corresponding to associated AvailabilityType instances.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\WCalendarPIM\AvailabilityType*.

Table 5.175 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
granularity	The finest precision that can be used to define times, such as the value of dtStart or duration	Duration See Clause 6.6.1.9.	[0..1]
timeRange	The span of time that bounds the associated sequence of intervals defined using one or more AvailabilityType instances. At instantiation time the choice must be made whether to express the interval as a UTCDateTimeInterval or a LocalDateTimeInterval.	DateTimeInterval See Clause 6.6.1.1.	[0..1]
priority	Integer value from 1 to 9. Higher numbers have lower priority per RFC5545.	Integer See Clause 6.6.3.5.	[0..1]

This element has the following connections to other elements:

Table 5.176 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	availability	The specification of a single interval, and optionally the rules for the pattern of intervals that constitute a sequence.	AvailabilityType See Clause 5.7.3.2.5.	[1..*]

5.7.3.2.11. FreqRecurType (Enumeration)

Used to identify the unit of time to be used in specifying the frequency of a recurring event.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\WCalendarPIM\FreqRecurType*.

Table 5.177 – FreqRecurType Enumerated Values

Enumerated Value	Definition
SECONDLY	Used to specify repeating events based on an interval of a second or more.
MINUTELY	Used to specify repeating events based on an interval of a minute or more.
HOURLY	Used to specify repeating events based on an interval of an hour or more.
DAILY	Used to specify repeating events based on an interval of a day or more.
WEEKLY	Used to specify repeating events based on an interval of a week or more.
MONTHLY	Used to specify repeating events based on an interval of a month or more.
YEARLY	Used to specify repeating events based on an interval of a year or more.

5.7.3.2.12. RelationshipType (Enumeration)

Specifies the type of hierarchical relationship associated with the calendar component specified by the property. The parameter value can be PARENT, to indicate that the referenced calendar component is a superior of calendar component; CHILD to indicate that the referenced calendar component is a subordinate of the calendar component; or SIBLING to indicate that the referenced calendar component is a peer of the calendar component. If this parameter is not specified on an allowable property, the default relationship type is PARENT.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\WSCalendarPIM\RelationshipType*.

Table 5.178 – RelationshipType Enumerated Values

Enumerated Value	Definition
PARENT	Indicates that the referenced calendar component is a superior of calendar component - default
CHILD	Indicates that the referenced calendar component is a subordinate of the calendar component. For example, a GluonType would have a CHILD relationship with the IntervalType it is pointing to.
SIBLING	Indicates that the referenced calendar component is a peer of the calendar component. For example, an IntervalType would have a SIBLING relationship with another IntervalType it is pointing to.

5.7.3.2.13. TemporalRelationshipType (Enumeration)

The temporal relation of one interval to the interval it is linked to.

UML Element Location: *Model\UML Model Elements from External Sources\OASIS\WSCalendarPIM\TemporalRelationshipType*.

Table 5.179 – TemporalRelationshipType Enumerated Values

Enumerated Value	Definition
FinishToStart	The next interval immediately begins after the previous interval (the default interpretation).
FinishToFinish	The next interval ends at the same time as the previous interval.
StartToFinish	The next interval finishes immediately before the previous interval (i.e., each interval in a sequence points to the interval immediately preceding it in time).
StartToStart	The next interval begins at the same time as the previous interval.

5.7.4. IEC61850

IEC 61850 is the international standard for communications with electric power and related system intelligent electric devices (IED). There are many parts to this standard.

FSGIM adopts parts of the information model from IEC61850-7-2,3,4 and 420. The latter part deals with the modeling of distributed energy resources (DER). It contains a set of model classes that allow for the nameplate, status, and control of DERs.

These model elements were extracted from the corresponding 61850 parts and were modified to use the primitives of FSGIM where there was a one to one correspondence. LogicalNode class names were expanded to make them more "human readable". No semantic changes were made to the imported model elements.

UML Package Location: *Model\Model Elements from External Sources\IEC61850*

5.7.4.1. IEC61850_7_420

This package contains Common Data Classes, Logical Nodes, and enumerations from IEC 61850-Part 7-420: Basic communication structure – Distributed energy resources logical nodes.

These information models are designed to be used in the exchange of information with distributed energy resources (DER), which comprise dispersed generation devices and dispersed storage devices, including reciprocating engines, fuel cells, microturbines, photovoltaics, combined heat and power, and energy storage.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420*

5.7.4.1.1. CommonDataClasses

These are Common Data Classes (CDC)s defined in part 7-420.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\CommonDataClasses*

5.7.4.1.2. DAEnums

These are enumerations defined for use in the part 7-420 CDCs.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums*

5.7.4.1.3. LogicalNodes

These Logical Nodes comprise the details of information modeling of distributed energy resources for IEC 61850-7-420.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes*

5.7.4.2. IEC61850_7_2

A formal meta-model and CoreAcsi API, which should reflect (and hopefully improve) the contents of IEC61850-2 Ed.1. Meta-model allows, among others, for a uniform definition of data-related concepts (LNs, CDCs, FCDAs), while the CoreAcsi API defines ACSI services.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_2*

5.7.4.2.1. BasicDAs

part 7-2, sec.5

Primitive data attribute types.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_2\BasicDAs*

5.7.4.2.2. CoreTypes

part 7-2, sec.5

Core types, mainly primitive, common to the whole model, that can be mapped to a given implementation technology thus keeping the rest of the model implementation agnostic.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_2\CoreTypes*

5.7.4.2.3. MetaModel

Meta-model reflecting contents of IEC61850-7-x. Can be instantiated in at least two ways:

1. By connecting to one or more ACSI Servers at the given IP address(es), without any other configuration required.
2. By reading SCL file (and potentially connecting to one or more ACSI Servers).

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_2\MetaModel*

5.7.4.2.4. ObjectReferences

Not explicitly defined in the standard. We define subtypes for ObjectReference type (which is just a string), so as to allow for validation per type of element. This is useful when there is no detailed LN/CDC/FCDA/DA model, and when an application receives only strings as references from the underlying MMS stack - it allows easy building of the meta-model objects.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_2\ObjectReferences*

5.7.4.3. IEC61850_7_3

part 7-3, clauses 5, 6 (except for 6.2.7 and 6.2.8 - discussion on quality), 7, 8, annex A, and enums part of annex D ().

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3*

5.7.4.3.1. CommonDataClasses

Common data classes are defined for use in part IEC 61850-7-4. Common data classes are composed of constructed attribute classes defined in Clause 6 of this part or of types defined in IEC 61850-7-2 or of Common data classes defined in this clause. IEC 61850-7-1 provides the basic notation used in this clause.

The common data classes define the relation between their attributes and the functional constraint as well as the possible trigger options. If two trigger options are stated, then a concrete implementation shall select one of them. The selection is based on the purpose of the data object of this common data class and is fix for the data object within a LN class. The semantic of the SubDataObjects and DataAttributes is defined in Clause 8.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses*

5.7.4.3.1.1. CDCAnalogueInfo

Measured values as defined in the following clauses may also be applied to calculated values.
For applicable services, see Annex B.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo*

5.7.4.3.1.2. CDCAnalogueSet

For applicable services, see Annex B.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueSet*

5.7.4.3.1.3. CDCControl

All common data classes for control include both the control and the related status information.
For applicable services, see Annex B.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControl*

5.7.4.3.1.4. CDCDescription

For applicable services, see Annex B.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCDescription*

5.7.4.3.1.5. CDCStatusInfo

For applicable services, see Annex B.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo*

5.7.4.3.1.6. CDCStatusSet

For applicable services, see Annex B.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet*

5.7.4.3.2. ConstructedDAs

Constructed attribute classes are defined for the use in common data classes (CDC) in Clause 7.

IEC 61850-7-1 provides an overview of all IEC 61850-7 documents (IEC 61850-7-2, IEC 61850-7-3, and IEC 61850-7-4). IEC 61850-7-1 also describes the basic notation used in IEC 61850-7-3 and the description of the relations between the IEC 61850-7 documents.

NOTE The common ACSI type 'TimeStamp' is specified in IEC 61850-7-2.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs*

5.7.4.3.3. DAEnums

This package contains explicit definition of enumerated types used in IEC 61850-7-3. Except for two of them, not explicitly defined in the standard.

Note: Annex D in the original defines one type per data object, so there is some repetition.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums*

5.7.4.4. IEC61850_7_4

Part 7-4, from Clause 5.3 to the end of Clause 6, Annex H, and content of Annex A (but without table).

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4*

5.7.4.4.1. DerivedCDCs

This package contains CDC specialisations through enumeration or transient. Not explicitly defined in the standard.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\DerivedCDCs*

5.7.4.4.2. LogicalNodeEnums

This subclause contains explicit definition of enumerated types used in 61850-7-4.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums*

5.7.4.4.3. LogicalNodes

This clause specifies logical nodes defined in IEC 61850-7-4. The types for logical nodes' data (common data classes) are defined in IEC 61850-7-3, while logical nodes' services and relationships to different control blocks are defined in IEC 61850-7-2.

In IEC 61850-7-4, concrete logical nodes are grouped into 16 groups:

- 1 containing 'special', or system logical nodes, and,
- 15 groups of domain logical nodes.

The names of logical nodes shall begin with the character representing the group to which the logical node belongs. For modelling per phase (for example switches or instrument transformers), one instance per phase shall be created; for modelling protection per zone or level, one instance per zone or level shall be created.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes*

5.7.4.4.3.1. LNGroupF

This group of logical nodes represents various types of control function blocks. Logical nodes of this group do include some form of control algorithm. They will normally be part of a logical device providing overall functionality within the system. Therefore, no description and requirements for the logical nodes of group F are given in IEC 61850-5. The LNs of the group F are never located at the border to the process.

The logical nodes of the group F shall be used only if another LN class from other groups does not fit to the semantic and function to be modelled.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupF*

5.7.4.4.3.2. LNGroupL

This group of logical nodes defines the system specific information. This includes information related to the physical device (represented by logical node LPHD), which contains the logical devices (represented by logical node LLN0) and the domain specific logical nodes (represented by DomainLN and all the concrete logical nodes, defined in the rest of this clause). This information (e.g., logical node mode control, nameplate information, operation counters) is independent of the application domain.

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupL*

5.7.4.3.3. LNGroupM

This group of logical nodes represents TODO

UML Package Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupM*

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

5.7.4.5. Elements Defined in the IEC61850 Model

The following elements are defined in this model.

5.7.4.5.1. Elements Defined in the IEC61850_7_420 Model

The following elements are defined in this model.

5.7.4.5.1.1. Elements Defined in the CommonDataClasses Model

The following elements are defined in this model.

5.7.4.5.1.1.1. ChargeStatusENS (Class)

The ChargeStatusENS class is an FSGIM class derived from the 61850 ENS class. The ChargeStatusENS class attributes are exactly the same as those in the ENS class except the status value attribute references the DRCS.ChaSt.ING enumeration class rather than the ENS class.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_420\CommonDataClasses\ChargeStatusENS.*

Table 5.180 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
description	Textual description of the data.	String See Clause 6.6.3.7.	[0..1]
quality	Quality of the attribute(s) representing the value of the data.	X.q.X See Clause 5.7.4.5.1.2.11.	[1]
stVal	Enumerated values for the status.	DRCS.ChaSt.ING See Clause 5.7.4.5.1.2.6.	[1]
timeOfLastChange	Timestamp of the last change in one of the attribute(s) representing the value of the data or in the quality attribute.	UTCDateTime See Clause 6.6.1.11.	[1]

5.7.4.5.1.1.2. GeneratorOperationStatusENS (Class)

The GeneratorOperationStatusENS class is an FSGIM class derived from the 61850 ENS class. The GeneratorOperationStatusENS class attributes are exactly the same as those in the ENS class except the status value attribute references another derived FSGIM enumeration named GeneratorOperationStatusEnumeration. The GeneratorOperationStatusEnumeration class is an explicitly defined enumeration in FSGIM. In 61850, this enumeration is represented by a single value.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_420\CommonDataClasses\GeneratorOperationStatusENS.*

Table 5.181 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
description	Textual description of the data.	String See Clause 6.6.3.7.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
quality	Quality of the attribute(s) representing the value of the data.	X.q.X See Clause 5.7.4.5.1.2.11.	[1]
stVal	Enumerated values for the status.	GeneratorOperationStatusEnumeration See Clause 5.7.4.5.1.2.4.	[1]
timeOfLastChange	Timestamp of the last change in one of the attribute(s) representing the value of the data or in the quality attribute.	UTCDateTime See Clause 6.6.1.11.	[1]

5.7.4.5.1.1.3. ConnectionTypeENS (Class)

Type of electrical connection between the generator and the facility's electrical distribution system. This class is an FSGIM class derived from the 61850 ENS class. The ConnectTypeENS class attributes are exactly the same as those in the ENS class except the status value attribute references another derived FSGIM enumeration named ConnectionTypeEnumeration. The ConnectionTypeEnumeration class is an explicitly defined enumeration in FSGIM. In 61850, this enumeration is represented by a single value.

UML Element Location: *Model\Model Elements from External Sources\IEC 61850_7_420\CommonDataClasses\ConnectionTypeENS.*

Table 5.182 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
description	Textual description of the data.	String See Clause 6.6.3.7.	[0..1]
quality	Quality of the attribute(s) representing the value of the data.	X.q.X See Clause 5.7.4.5.1.2.11.	[1]
stVal	Enumerated values for the status.	ConnectionTypeEnumeration See Clause 5.7.4.5.1.2.3.	[1]
timeOfLastChange	Timestamp of the last change in one of the attribute(s) representing the value of the data or in the quality attribute.	UTCDateTime See Clause 6.6.1.11.	[1]

5.7.4.5.1.1.4. SequenceDirectionENS (Class)

The SequenceDirectionENS class is an FSGIM class derived from the 61850 ENS class. The SequenceDirectionENS class attributes are exactly the same as those in the ENS class except the status value attribute references another derived FSGIM enumeration named SequenceDirectionEnumeration. The SequenceDirectionEnumeration class is an explicitly defined enumeration in FSGIM. In 61850, this enumeration is represented by a single value.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_420\CommonDataClasses\SequenceDirectionENS.*

Table 5.183 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
description	Textual description of the data.	String See Clause 6.6.3.7.	[0..1]
quality	Quality of the attribute(s) representing the value of the data.	X.q.X See Clause 5.7.4.5.1.2.11.	[1]
stVal	Enumerated values for the status.	SequenceDirectionEnumeration See Clause 5.7.4.5.1.2.5.	[1]
timeOfLastChange	Timestamp of the last change in one of the attribute(s) representing the value of the data or in the quality attribute.	UTCDateTime See Clause 6.6.1.11.	[1]

5.7.4.5.1.1.5. SCR (Relative time schedule) (Class)

The SCR class defines the common data class definition for a "Relative Time Schedule Setting". This class provides a means for defining a relative time array of setting values, such as schedules, where the time interval between points may be variable. It has an array of value elements that is characterized by a set of parameters including the number of elements in each array, an array of times between each value and an indication how to ramp between consecutive values, units, multiplier, and a description.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_420\CommonDataClasses\SCR.*

Table 5.184 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
cur	Currency as per ISO 4217.	CurrencyKind See Clause 6.6.2.1.	[0..1]
d	Textual description of the data.	String See Clause 6.6.3.7.	[0..1]
multiplier	Units multiplier of the attribute valUnits.	SiScaleCodeType See Clause 6.6.2.3.	[0..1]
numPts	Number of points used to define a curve or a histogram. This contains the dimension of array attributes val, rmpTyp, and tmsOffset.	Integer See Clause 6.6.3.5.	[1]
occPer	Repeat period (Hour, Day, Week, Month, Year).	X.occPer.SCR See Clause 5.7.4.5.1.2.10.	[1]
rmpTyp	Array of ramp types defining how to interpret the transition from one value to the next. ARRAY 1..numPts OF the enumeration X.rmpTyp.X (1=Fixed, 2=Ramp, 3=Average)	X.rmpTyp.X See Clause 5.7.4.5.1.2.12.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
tmsOffset	ARRAY 1..numPts OF Integer time offsets in seconds	Integer See Clause 6.6.3.5.	[1]
val	ARRAY 1..numPts OF REAL	Real See Clause 6.6.3.6.	[1]
valD	Description of val.	String See Clause 6.6.3.7.	[0..1]
valEq	Equation for val. 1 = SI units. 2 = Currency as per ISO 4217 per SI unit. 3 = SI unit per currency.	X.valEq.X See Clause 5.7.4.5.1.2.13.	[0..1]
valUnits	SI Units.	UnitSymbolKind See Clause 6.6.2.6.	[0..1]

5.7.4.5.1.2. Elements Defined in the DAEnums Model

The following elements are defined in this model.

5.7.4.5.1.2.1. ScheduleOperatingModeKind (Enumeration)

Ed. Note: The ScheduleOperatingModeKind is a newly defined enumeration in 61850_7. It has not as yet (Sep 2013) been approved by the committee. Hence, this enumeration was cloned from the 61850 working documents and is subject to modification.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850_7_420\DAEnums\ScheduleOperatingModeKind.*

Table 5.185 - ScheduleOperatingModeKind Enumerated Values

Enumerated Value	Definition
NA	Not applicable/unknown
energy	Energy
continuousReserve	Continuous Reserve Spinning
continuousReserveSupplemental	Continuous reserve supplemental
energyReserve	Energy reserve
emissionReserve	Emission reserve
energyBalancing	Energy balancing
reactivePower	Reactive power
blackStart	Black start
islanding	Emergency islanding
other	Other

5.7.4.5.1.2.2. SequenceDirectionENS (Enumeration)

Enumeration that defines the phase sequence connection of the generator.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\SequenceDirectionENS.*

Table 5.186 - SequenceDirectionENS Enumerated Values

Enumerated Value	Definition
ABC	Wiring sequence that creates positive phase angles in the clockwise direction.
CBA	Wiring sequence that creates positive phase angles in the counter-clockwise direction.

5.7.4.5.1.2.3. ConnectionTypeEnumeration (Enumeration)

Enumerated set of the type of electrical connection between the generator and the electrical distribution system.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\ConnectionTypeEnumeration.*

Table 5.187 - ConnectionTypeEnumeration Enumerated Values

Enumerated Value	Definition
NA	Not Applicable.
SinglePhase	Single Phase electrical connection.
SplitPhase	Split Phase electrical connection.
TwoPhase	Two Phase electrical connection.
ThreePhaseDelta	Generator connects each winding to one phase of a three phase electrical interconnection.
ThreePhaseWye	Generator connects each winding from a three phase wire to a common neutral point.
ThreePhaseWyeGrounded	Generator connects each winding from a three phase wire to a common ground.
ThreePhase_ThreeWire_Inverter Type	Three phase generator connected using three-wires with symmetrical voltages between phases.
ThreePhase_FourWire_Inverter Type	Three phase generator connected using four-wires with symmetrical voltages between phase and neutral.
Other	The connection is some other type of electrical connection type.

5.7.4.5.1.2.4. GeneratorOperationStatusEnumeration (Enumeration)

The GeneratorOperationStatusEnumeration class is an FSGIM enumeration defining the operation state of the generator.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\GeneratorOperationStatusEnumeration.*

Table 5.188 - GeneratorOperationStatusEnumeration Enumerated Values

Enumerated Value	Definition
NA	Not Applicable
NotOperating	The generator is not operating.
Operating	The generator is operating.
StartingUp	The generator is starting.
ShuttingDown	The generator is shutting down.
AtDisconnectLevel	Generator is at disconnect level.
Ramping Power	Generator is ramping power.
Ramping_ReactivePower	Generator is ramping reactive power.
Other	The generator is in some other state.

5.7.4.5.1.2.5. SequenceDirectionEnumeration (Enumeration)
The SequenceDirectionEnumeration class is an FSGIM enumeration defining the sequence direction for the generator.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\SequenceDirectionEnumeration.*

Table 5.189 - SequenceDirectionEnumeration Enumerated Values

Enumerated Value	Definition
ABC	The sequence direction is ABC.
CBA	The sequence direction is CBA.

5.7.4.5.1.2.6. DRCSt.Charging (Enumeration)
This enumeration indicates the state of charge of a storage unit.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\DRCSt.Charging.*

Table 5.190 - DRCSt.Charging Enumerated Values

Enumerated Value	Definition
off	Charging is off.
fullyDischarged	The storage unit is fully discharged.
discharging	The storage unit is discharging.
charging	The storage unit is charging.
fullyCharged	The storage unit is fully charged.
holding	The storage unit is holding.
testing	In testing mode.

5.7.4.5.1.2.7. DRCT.DERtyp.ING (Enumeration)

Distributed energy resource type.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\DRCT.DERtyp.ING.*

Table 5.191 - DRCT.DERtyp.ING Enumerated Values

Enumerated Value	Definition
Unknown	Not applicable / Unknown.
virtual	Virtual or mixed DER.
reciprocatingEngine	Reciprocating engine.
fuelCell	Fuel cell.
photoVoltaic	Photovoltaic system.
combinedHeatAndPower	Combined heat and power.
compressedAir	Not applicable / Unknown.
pumpedHydroelectric	Pumped hydroelectric system.
flywheel	Fly wheels.
thermalEnergyStorage	Thermal energy storage.
electroChemicalBattery	Electro chemical battery.
flowBattery	Flow batteries.
capacitor	Capacitance storage.
superconductingMagneticEnergy	Super-conducting magnetic energy storage.
other	Other.

5.7.4.5.1.2.8. MMXN.range.CMV (Enumeration)

Measurement minimum and maximum range thresholds.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\MMXN.range.CMV.*

Table 5.192 - MMXN.range.CMV Enumerated Values

Enumerated Value	Definition
normal	Normal.
high	High value.
low	Low value.
high-high	High high value.
low-low	Low low value.

5.7.4.5.1.2.9. X.ctlModel.X (Enumeration)

ENUMERATED (status-only | direct-with-normal-security | sbo-with-normal-security | directwith-enhanced-security | sbo-with-enhanced-security).

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.ctlModel.X.*

Table 5.193 - X.ctlModel.X Enumerated Values

Enumerated Value	Definition
status-only	The object is not controllable, only the services that apply to a status object are supported. The attribute ctlVal does not exist.
direct-with-normal-security	Direct control with normal security according to IEC 61850-7-2.
sbo-with-normal-security	SBO control with normal security according to IEC 61850-7-2.
direct-with-enhanced-security	Direct control with enhanced security according to IEC 61850-7-2.
sbo-with-enhanced-security	SBO control with enhanced security according to IEC 61850-7-2.

5.7.4.5.1.2.10. X.occPer.SCR (Enumeration)

Repeat period.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.occPer.SCR.*

Table 5.194 - X.occPer.SCR Enumerated Values

Enumerated Value	Definition
Hour	Hourly.
Day	Daily.
Week	Weekly.
Month	Monthly.
Year	Yearly.

5.7.4.5.1.2.11. X.q.X (Enumeration)

Validity component of the complex "quality" attribute with only the following values: good, Invalid, Questionable.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.q.X.*

Table 5.195 - X.q.X Enumerated Values

Enumerated Value	Definition
Good	The value shall be marked good if no abnormal condition of the acquisition function or the information source is detected.
Invalid	The value shall be marked invalid when an abnormal condition of the acquisition function or the information source (missing or non-operating updating devices) is detected.
Questionable	The value shall be marked questionable if a supervision function detects an abnormal behaviour, however the value could still be valid.

5.7.4.5.1.2.12. X.rmpTyp.X (Enumeration)

1=Fixed, 2=Ramp, 3=Average.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.rmpTyp.X*.

Table 5.196 - X.rmpTyp.X Enumerated Values

Enumerated Value	Definition
Fixed	1=Fixed.
Ramp	2=Ramp.
Average	3=Average.

5.7.4.5.1.2.13. X.valEq.X (Enumeration)

1 = SI units.
2 = Currency as per ISO 4217 per SI unit.
3 = SI unit per currency.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\DAEnums\X.valEq.X*.

Table 5.197 - X.valEq.X Enumerated Values

Enumerated Value	Definition
1	1 = SI units.
2	2 = Currency as per ISO 4217 per SI unit.
3	3 = SI unit per currency.

5.7.4.5.1.3. Elements Defined in the LogicalNodes Model

The following elements are defined in this model.

5.7.4.5.1.3.1. DERCostParameters (Class)

Economic drivers of operating the DER unit – operating cost, carbon rate cost, ramp cost, ... (61850 Logical Node = DCCCT).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERCostParameters*.

Table 5.198 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
CarbonRatesCost	Incremental carbon emission curve costs.	SCR See Clause 5.7.4.5.1.1.5.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
ContractualAncillary	Ability to provide ancillary services. 0 = N/A / Unknown. 1 = Load following. 2 = Immediate reserve (i.e., spinning reserve). 3 = Operational reserve (i.e., nonspinning reserve). 4 = Base load. 5 = Black start. 99 = Other.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
ContractualExportWLimit	Contractual limit on export energy.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
ContractualHighV	Contractual voltage high limit.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
ContractualImportWLimit	Contractual limit on import energy.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
ContractualLowV	Contractual voltage low limit.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
ContractualPowerFactor	Contractual power factor to be provided by DER.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
Currency	ISO 4217 currency 3-character code.	CUG See Clause 5.7.4.5.3.1.26.	[1]
HeatRateCost	Incremental heat rate piecewise linear curve costs.	SCR See Clause 5.7.4.5.1.1.5.	[0..1]
OperatingCost	Marginal operational cost per hour.	CUG See Clause 5.7.4.5.3.1.26.	[1]
OperatingWhCost	Marginal operational cost per kWh.	CUG See Clause 5.7.4.5.3.1.26.	[1]
PriceCode	Price code for the pricing signal.	ING See Clause 5.7.4.5.3.1.28.	[1]
RampCost	Cost to ramp DER per kW per minute.	CUG	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
StartCost	Cost to start up DER.	See Clause 5.7.4.5.3.1.26. CUG	[1]
StopCost	Cost to stop DER.	See Clause 5.7.4.5.3.1.26. CUG	[1]

5.7.4.5.1.3.2. DERDeviceControllerCharacteristics (Class)

Setpoints for control characteristics and capabilities – max ramp rate, start delay time, stop delay time, ... (61850 Logical Node = DRCT)

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERDeviceControllerCharacteristics.*

Table 5.199 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
CalculationTotalVA	Calculation method used for total apparent power calculation (Vector Arithmetic).	ENG See Clause 5.7.4.5.3.1.27.	[0..1]
DERNumber	Number of DER units connected to controller.	ING See Clause 5.7.4.5.3.1.28.	[1]
DERType	Type of DER unit: Enumerated. See DRCT.DERtyp.ING.	DRCT.DERtyp.ING See Clause 5.7.4.5.1.2.7.	[1]
LoadRampRate	Nominal ramp load or unload rate, power versus time.	ING See Clause 5.7.4.5.3.1.28.	[1]
MaximumRampRate	Setpoint for maximum ramp rate as percentage of nominal maximum ramp rate.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
MaximumVArLimit	Nominal max output reactive power.	ASG See Clause 5.7.4.5.3.1.10.	[1]
MaximumWLimitPercent	Nominal max output power.	ASG See Clause	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
MinimumReservePercent	Setpoint for minimum reserve for storage, as a percentage of the nominal maximum storage.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
OutputVarNominal	Reactive power in percent of VarMax	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
PowerFactorSignConvention	Power factor convention: IEC=1 IEEE=2.	ENG See Clause 5.7.4.5.3.1.27.	[0..1]
PowerReference	Reference active power with the default set equal to Wmax.	ASG See Clause 5.7.4.5.3.1.10.	[1]
StartDelayTimeSeconds	Nominal time delay before starting or restarting.	ING See Clause 5.7.4.5.3.1.28.	[1]
StopDelayTimeSeconds	Nominal time delay before stopping.	ING See Clause 5.7.4.5.3.1.28.	[1]
VArChargerMaximum	Setpoint for maximum apparent charging power.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
VArMaximum	Setpoint for maximum apparent power.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
VArAction	Var action on change between charging and discharging: switch=1, remain=2.	ENC See Clause 5.7.4.5.3.1.15.	[0..1]
VArMaximum	Setpoint for maximum reactive power.	ASG See Clause 5.7.4.5.3.1.10.	[1]
VMaximum	Setpoint for maximum voltage.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
VMinimum	Setpoint for minimum voltage.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
VReference	Reference voltage for functions using grid voltage as input.	ASG See Clause 5.7.4.5.3.1.10.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
VReferenceOffset	Reference offset voltage for functions using grid voltage as input.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
WChargerGradient	Setpoint for maximum charging ramp rate: percentage of WChaMax.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
WChargerMaximum	Setpoint for maximum active charging power and reference value for functions.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
WGradient	Gradient: percentage of delta power per Hertz.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
WMaximum	Setpoint for maximum active power and reference value for functions.	ASG See Clause 5.7.4.5.3.1.10.	[1]

5.7.4.5.1.3.3. DERGeneratorRatings (Class)

Information on the nameplate ratings and other identifying data of the generation/storage unit.

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERGeneratorRatings.*

Table 5.200 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
ConnectionType	0 N/A. 1 Single phase. 2 Split phase. 3 2-phase. 4 3-phase delta. 5 3-phase wye. 6 3-phase wye grounded. 7 3-phase / 3-wire (inverter type). 8 3-phase / 4-wire (inverter type). 99 Other.	ConnectionTypeENS See Clause 5.7.4.5.1.1.3.	[1]
CurrentRating	Current rating under nominal voltage under nominal power factor.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
DERType	Type of Distributed Energy Resource.	DRCT.DERtyp.ING See Clause 5.7.4.5.1.2.7.	[1]
DisconnectLevelW	Generator disconnect level.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
EfficiencyRatingPercent	Efficiency at rated capacity.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
EmergencyMaximumWOutput	Max watt output - emergency limits for different minutes.	CSG See Clause 5.7.4.5.3.1.11.	[0..1]
EmergencyMinimumWOutput	Min watt output - emergency limits for different minutes.	CSG See Clause 5.7.4.5.3.1.11.	[0..1]
EmergencyRampRate	Emergency ramp rate.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
FaultARating	Max fault current rating.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
FaultDurationTimeInSeconds	Exposure to fault rates as percent.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
FaultRatingPercent	Max fault current rating.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
FrequencyRating	Nominal frequency.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
GroundZ	Grounding impedance.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
LowerLoadSetpointRate	Lower baseload setpoint rate.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
MaximumFaultRating	Max fault duration rating.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
MaximumLoadRamp	max load ramp rate.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
MaximumUnloadRamp	max unload ramp rate.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
MaximumVarOutput	Max VAR output.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
MaximumWOutput	Max watt output - continuous.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
MinimumWOutput	Min watt output - continuous.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
RaiseLoadSetpointRate	Raise baseload setpoint rate.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
SelfPowerFactor	Self-service nominal power factor.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
SelfV	Self-service voltage.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
SelfVRange	Self-service acceptable voltage range.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
SelfW	Self-service nominal power.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
SequenceDirection	Sequence (direction): ABC or CBA. 0 ABC. 1 CBA.	SequenceDirectionENS See Clause 5.7.4.5.1.2.2.	[0..1]
TemperatureRating	Max temperature rating.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
VARating	Max volt-amperes rating (apparent power).	ASG See Clause 5.7.4.5.3.1.10.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
VarRating	Max VAR rating (reactive power).	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
VRating	Voltage rating of storage device.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
WHRating	Energy rating of storage device.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
WRating	Max watt rating (real power).	ASG See Clause 5.7.4.5.3.1.10.	[0..1]

5.7.4.5.1.3.4. DEROperationalModeControls (Class)

Operating mode at the ECP.

Control of the operational modes of the DER (e.g., constant watts, constant vars.) More than one mode can be set simultaneously for certain logical combinations (61850 Logical Node = DOPM).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850-7_420\LogicalNodes\DEROperationalModeControls.*

Table 5.201 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
OperationalModeConstantW	Mode of operation - constant watts.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationalModeConstantPowerFactor	Mode of operation - constant power factor.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationalModeConstantV	Mode of operation - constant voltage.	SPG See Clause 5.7.4.5.3.1.30.	[0..1]
OperationalModeConstantVar	Mode of operation - constant vars.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationalModeExportImport	Mode of operation - constant export/import.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
OperationModeIslanded	Mode of operation - islanded at the ECP.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationModeMaximumVAr	Mode of operation - maximum vars.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationModePeak	Mode of operation - peak demand reduction.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationModePM	Mode of operation – driven by energy source (e.g., solar, water flow) so generation level is constrained by availability of that energy source.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationModeVOVERRIDE	Mode of operation - voltage override.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
RampTimeSeconds	Ramp time (in seconds) for moving from the present operational mode settings to new operational mode settings.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
RevertTimeSeconds	Timeout period (in seconds), after which the device will revert to its default status, such as closing the switch to reconnect to the grid.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
WindowTimeSeconds	Time window (in seconds) within which to randomly execute a command. If the time window is zero, the command will be executed immediately.	ING See Clause 5.7.4.5.3.1.28.	[0..1]

5.7.4.5.1.3.5. DERThermalStorage (Class)

This logical node describes the characteristics of the thermal storage. This LN applies both to heat storage and to coolant storage, and is used for measurements of heat exchanges.

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\LogicalNodes\DERThermalStorage*.

Table 5.202 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
ThermalCapacityPercent	Remaining actual thermal energy capacity as percent of total available capacity	MV See Clause 5.7.4.5.3.1.6.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
ThermalCapacityTotal	Total available thermal energy capacity	MV See Clause 5.7.4.5.3.1.6.	[0..1]
ThermalInput	Instantaneous thermal energy input into storage	MV See Clause 5.7.4.5.3.1.6.	[0..1]
ThermalLost	Thermal energy lost or dumped	MV See Clause 5.7.4.5.3.1.6.	[0..1]
ThermalOutput	Instantaneous thermal energy output from storage	MV See Clause 5.7.4.5.3.1.6.	[0..1]
ThermalOutputEstimated	Estimated instantaneous thermal energy output over time (using time offsets)	SCR See Clause 5.7.4.5.1.1.5.	[0..1]
ThermalStorageType	0 Not applicable / Unknown 1 For heating with storage 2 For heating without storage 3 For cooling with storage 4 For cooling without storage 99 Other	ENG See Clause 5.7.4.5.3.1.27.	[1]

5.7.4.5.1.3.6. DERUnitControlActions (Class)

Basic control operations – start / stop, test mode, ... (61850 Logical Node = DRCC).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitControlActions.*

Table 5.203 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
AutoManualControl	Sets operations mode to automatic or manual.	SPC See Clause 5.7.4.5.3.1.16.	[1]
DCPowerStatus	True = DC power control.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
DeratedRatePercent	Derated load target as percent.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
DERStart	True = Start DER unit or sequencer.	SPC See Clause 5.7.4.5.3.1.16.	[1]
DERStop	True = Stop DER unit or sequencer.	SPC See Clause 5.7.4.5.3.1.16.	[1]
EmergencyStop	Remote emergency stop.	DPC See Clause 5.7.4.5.3.1.14.	[0..1]
FaultAcknowledge	True = Acknowledge fault clearing.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
GeneratorSynchronization	True = Starts synchronizing generator to EPS.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
ImportExportSetting	Setpoint for maintaining constant import/export energy at ECP.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
LoadModeAvailable	Sets "available": True = for connection to load.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
LoadModeBase	Sets "base load" : True = load mode.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
LoadModeFixedExport	Sets "fixed export" True = Load mode.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
LoadModeFollowing	Sets "load following": True = load mode.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
LoadRamp	Ramp load or unload rate.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
LoadShareRamp	Load share/Do not share.	APC See Clause 5.7.4.5.3.1.13.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
LoadShutDown	Load shut down: Stop/Do not stop.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
LoadWPercent	Percent load power.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
LocalRemoteControl	Sets operations mode to remote or local.	SPC See Clause 5.7.4.5.3.1.16.	[1]
MaximumVArLimit	Derated max output reactive power.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
MaximumWLimitPercent	Percent of reference active power watts as maximum allowed watts output - in ING. * setVal = percentage value between minVal and maxVal. * minVal = 0% or greater. * maxVal = 100% of less.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
OperationModeAvailable	Sets operational mode: True = is available.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationModeOff	Sets operational mode: True = off-line.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationModeTest	Sets operational mode: True = test mode.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OperationTimeReset	True = Reset operational time.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
OutputHzSetting	Setpoint for maintaining fixed frequency offset.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
OutputPowerFactorSetting	Setpoint for maintaining fixed power factor: negative power factor is leading and positive power factor is lagging. OutPFSet is specified with CDC: APC in the -420 spec and CDC: ING and in the v15 document. APC v. ING is under investigation at the working group level.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
OutputVArNominal	Reactive power in percent of Wmax.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
OutputVArSetting	Output target reactive power setpoint.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
OutputVSetting	Voltage setpoint for maintaining fixed voltage level in percent offset.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
OutputWRate	Mode of operations / setpoint for charge/discharge rate.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
OutputWSetting	Output target power setpoint.	APC See Clause 5.7.4.5.3.1.13.	[0..1]
PowerFactorExcitation	Power Factor excitation: Underexcited = True; Overexcited = False.	SPG See Clause 5.7.4.5.3.1.30.	[0..1]
StartDelayTimeSeconds	Time delay before starting.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
StopDelayTimeSeconds	Time delay before stopping.	ING See Clause 5.7.4.5.3.1.28.	[0..1]

5.7.4.5.1.3.7. DERUnitGenerator (Class)

More detailed status of DER unit – operation time, operation counts, ... (61850 Logical Node = DGEN).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_420\LogicalNodes\DERUnitGenerator.*

Table 5.204 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
AutomaticVoltageRegulatorPercentDutyCycle	Automatic voltage regulator percent duty cycle.	MV See Clause 5.7.4.5.3.1.6.	[1]
DCPowerStatus	DC Power On/Off Status. True = On. False = Off.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
GeneratorBlocked	Set generator as blocked. True = blocked from being turned on.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
GeneratorControl	Starts or stops the generator: True = Start, False = Stop, other states indicate error condition.	DPC See Clause 5.7.4.5.3.1.14.	[0..1]
GeneratorCoolDownTime	Timer for generator to cool down; min = minimum time for cool down.	MV See Clause 5.7.4.5.3.1.6.	[1]
GeneratorHarmonics	Generator harmonics.	HMV See Clause 5.7.4.5.3.1.5.	[1]
GeneratorOnCount	The number of times that the generator has been turned on: count of "generator on" times, since the last time the counter was reset.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
GeneratorOperationStatus	Generation operational state. 0 = N/A. 1 = Not operating. 2 = Operating. 3 = Starting-up. 4 = Shutting-down. 5 = At disconnect level. 6 = Ramping (power). 7 = Ramping (reactive power). 99 = Other.	GeneratorOperationStatusENS See Clause 5.7.4.5.1.1.2.	[0..1]
GeneratorOperationTime	Elapsed time as the generator becomes ready after the GenOnOff command was issued. Max = maximum time before issuing a start-failure alarm.	BCR See Clause 5.7.4.5.3.1.21.	[1]
GeneratorRaiseLower	Raises or lowers the generation level by steps: True = Raise, False = Lower, other states indicate error condition.	DPC See Clause 5.7.4.5.3.1.14.	[0..1]
GeneratorStabilizationTime	Timer for stabilization time; max = maximum time before issuing a stabilization-failure alarm.	MV See Clause 5.7.4.5.3.1.6.	[1]
GeneratorSynchronization	Generator is synchronized to EPS, or not. True = Synchronized. False = Not synchronized.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
OperatingTimeHours	Operation time.	INS See Clause	[1]

Attribute Name	Description	Attribute Type	Multiplicity
OperatingTimeSecondsResettable	Total time generator has operated - re-settable: accumulated time since the last time the counter was reset.	INS See Clause 5.7.4.5.3.1.23.	[1]
ParallelStatus	Paralleling Status. True = Paralleling. False = Standby.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
PeriodStartCount	Count of starts in period since reset.	BCR See Clause 5.7.4.5.3.1.21.	[1]
PeriodWh	Energy in period since last reset.	MV See Clause 5.7.4.5.3.1.6.	[1]
RampLoadSwitch	Ramp Load/Unload Switch. True = Ramp Load. False = Ramp Unload.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
TotalStartCount	Count of total number of starts.	BCR See Clause 5.7.4.5.3.1.21.	[1]
TotalWh	Total energy delivered.	MV See Clause 5.7.4.5.3.1.6.	[1]

5.7.4.5.1.3.8. DERUnitStatus (Class)

Present status of the DER – current mode, auto/manual state, ... (61850 Logical Node = DRCS).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\7_420LogicalNodes\DERUnitStatus.*

Table 5.205 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
AutoManual	Automatic or manual mode. True = automatic. False = manual.	SPS See Clause 5.7.4.5.3.1.24.	[1]
ChargeStatus	Charge status of a storage device. 1 = Off 2 = Fully discharged	ChargeStatusENS See Clause 5.7.4.5.1.1.1.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
	3 = Discharging 4 = Charging 5 = Fully charged 6 = Holding 7 = Testing		
DCPowerStatus	DC Power status. True = Power on. False = Power not on.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
ElectricalConnectionPointConnected	Electrically connected to the ECP that it is physically connected to. True = connected. False = not connected.	SPS See Clause 5.7.4.5.3.1.24.	[1]
FaultRatePercent	Fault rates of DER as percent.	MV See Clause 5.7.4.5.3.1.6.	[0..1]
LoadModeAvailable	Load mode - True: Available.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
LoadModeBase	Load mode - True: Base load.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
LoadModeFixedExport	Load mode - True: fixed export.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
LoadModeFollowing	Load mode - True: Load following.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
Local	Remote or local mode. True = remote is allowed. False = Local.	SPS See Clause 5.7.4.5.3.1.24.	[1]
ModeOffAvailable	Operational mode - True: Off and available to start.	SPS See Clause 5.7.4.5.3.1.24.	[1]
ModeOffUnavailable	Operational mode: True Off and not available to start.	SPS See Clause 5.7.4.5.3.1.24.	[1]
ModeOnAvailable	Operational mode - True : On and available for connection.	SPS See Clause 5.7.4.5.3.1.24.	[1]
ModeOnConnected	Operational mode - True: on and connected.	SPS	[1]

Attribute Name	Description	Attribute Type	Multiplicity
ModeOnUnavailable	Operational mode - True: On but not available for connection.	See Clause 5.7.4.5.3.1.24.	
ModeStartingUp	Mode is starting up	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
ModeStop	Operational mode - True: Stopping/shutting down.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
ModeTest	Operational mode - True: Test mode.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
ModeVAr	Yes/No, VAr management is available.	SPS See Clause 5.7.4.5.3.1.24.	[1]
OperationTimeHours	Operation time.	INS See Clause 5.7.4.5.3.1.23.	[1]
SelfServeWh	Actual self service energy used.	MV See Clause 5.7.4.5.3.1.6.	[1]
SequencePosition	Sequence active position or step.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
SequencerStatus	Status of the sequencer.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
VAChargePercent	Actual percentage charged based on VAChaMax.	MV See Clause 5.7.4.5.3.1.6.	[0..1]
VAPercent	Actual percentage apparent power output based on VAMax.	MV See Clause 5.7.4.5.3.1.6.	[0..1]

5.7.4.5.2. Elements Defined in the IEC61850_7_2 Model

The following elements are defined in this model.

5.7.4.5.2.1. Elements Defined in the CoreTypes Model

The following elements are defined in this model.

5.7.4.5.2.1.1. DpStatusKind (Enumeration)

Double point status kind.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850_7_2\CoreTypes\DpStatusKind*.

Table 5.206 - DpStatusKind Enumerated Values

Enumerated Value	Definition
intermediateState	Intermediate state.
offOrFalse	Off or False state.
onOrTrue	On or True state.
badState	The server can not detect whether the position is open (off), closed (on) or in intermediate state.

5.7.4.5.2.2. Elements Defined in the MetaModel Model

The following elements are defined in this model.

5.7.4.5.2.2.1. EnumDA (Abstract Class)

Not explicitly defined in the standard. Enumerated data attribute has only a value field of some enumerated type.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_2\MetaModel\EnumDA.*

5.7.4.5.2.3. Elements Defined in the ObjectReferences Model

The following elements are defined in this model.

5.7.4.5.2.3.1. ObjectReference (Abstract Class)

part 7-2, sec. 5

Instances of classes in the hierarchical information model (ACSI class hierarchy of logical device, logical node, data, data attributes) shall be constructed by the concatenation of all instance names comprising the whole path-name of an instance of a class that identifies the instance uniquely. The ObjectReference syntax shall be: LDName/LNName[. ...]. The "/" shall separate the instance name of a logical device (LDName) from the name of an instance of a logical node (LNName). The "." shall separate the further names in the hierarchy. The "[]" shall indicate an option. The inner square bracket "[...]" shall indicate further names of nested definitions.

NOTE: Section 19 specifies constraints on the use of the type ObjectReference.

Modelling notes:

- We define this type as abstract class and extend it with the subtypes which carry semantics of the above naming scheme.
- Subtypes of ObjectReference provide methods that return separately all the elements that make the reference (e.g., FC, LDName, LNName), which have been parsed from the reference string.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_2\ObjectReferences\ObjectReference.*

5.7.4.5.3. Elements Defined in the IEC61850_7_3 Model

The following elements are defined in this model.

5.7.4.5.3.1. Elements Defined in the CommonDataClasses Model

The following elements are defined in this model.

5.7.4.5.3.1.1. Elements Defined in the CDCAnalogueInfo Model

The following elements are defined in this model.

5.7.4.5.3.1.2. CMV (Complex measured value) (Class)

This common data class is used to represent complex (phasor) measured values.

Parent Class: RangeDeadbandCDC (See Clause 5.7.4.5.3.1.7.)

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\CMV.*

Table 5.207 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
instCVal	Instantaneous complex (phasor) value, with magnitude and angle.	Vector See Clause 5.7.4.5.3.2.10.	[0..1]
cVal	Deadbanded complex value, based on a deadband calculation from the instantaneous value 'instCVal.mag' with 'db' and 'instCVal.ang' with 'dbAng', independently. See 'MV.mag'.	Vector See Clause 5.7.4.5.3.2.10.	[1]
rangeAng	Range in which the current instantaneous value is ('MV.instMag' or 'CMV.instCVal.mag'). It may be used to report an event if its value changes (due to the change in the instantaneous value). See RangeConfig. NOTE The use of algorithms to filter events based on transition from one range to another is a local issue.	RangeKind See Clause 5.7.4.5.3.3.6.	[0..1]
q	Quality of the data values 'instCVal', 'cVal', 'range'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change in one of the data values 'cVal', 'range', or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
subCVal	Value used to substitute 'instCVal'.	Vector See Clause 5.7.4.5.3.2.10.	[0..1]
units	Unit for 'instCVal.mag', 'cVal.mag', 'subCVal.mag', 'rangeC'.	Unit See Clause 5.7.4.5.3.2.9.	[0..1]
dbAng	Deadband is a configuration parameter used to calculate deadbanded angle of a complex value ('cVal.ang'). The value of 'dbAng' shall represent the percentage of difference ('rangeAngC.max' - 'rangeAngC.min') in units of 0.001%. See 'RangeDeadbandCDC.db'.	Integer See Clause 6.6.3.5.	[0..1]
rangeAngC	Configuration for 'rangeAng'.	RangeConfig See Clause 5.7.4.5.3.2.7.	[0..1]
magSVC	Configuration for scaled value representation of magnitudes 'instCVal.mag', 'cVal.mag', 'subCVal.mag'.	ScaledValueConfig See Clause 5.7.4.5.3.2.8.	[0..1]
angSVC	Configuration for scaled value representation of angles 'instCVal.ang', 'cVal.ang', 'rangeAngC.min' and 'rangeAngC.max'.	ScaledValueConfig See Clause 5.7.4.5.3.2.8.	[0..1]
angRef	Angle reference, indicating the quantity that is used as reference for the phase angles 'cVal.ang', 'instCVal.ang', 'subCVal.ang'. For the indicated quantity, the fundamental frequency (index = 1) is used as reference by convention.	AngleReferenceKind See Clause 5.7.4.5.3.3.1.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
smpRate	Number of samples per nominal period that has been used to determine instantaneous value 'instCVal.mag'. In the case of a dc system, it is number of samples per s.	Integer See Clause 6.6.3.5.	[0..1]

5.7.4.5.3.1.3. DEL (Phase-to-phase measured values of a three phase system) (Class)

This class is a collection of measurements of values in a three phase system that represent phase to phase values ("delta").

Values for 'phsAB', 'phsBC' and 'phsCA' have been simultaneously acquired or determined, and their respective time stamps hold the same value 't.secondSinceEpoch'. It shall be assumed that any jitter between the acquisition times dedicated for these values is neglectable. The jitter for simultaneity shall be as indicated in any of the respective 't.timeQuality' attributes of the above values.

Attribute 'angRef' is used in place of individual 'angRef' attributes of 'phsAB', 'phsBC' and 'phsCA'.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogInfo\DEL.*

Table 5.208 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
phsAB	Value of phase A to phase B measurement.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
phsBC	Value of phase B to phase C measurement.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
phsCA	Value of phase C to phase A measurement.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
angRef	Angle reference, indicating the quantity that is used as reference for the phase angles {'phsAB', 'phsBC', 'phsCA'}. 'instCVal.ang', instead of their own 'angRef'. For the indicated quantity, the fundamental frequency (index = 1) is used as reference by convention.	AngleReferenceKind See Clause 5.7.4.5.3.3.1.	[0..1]

5.7.4.5.3.1.4. HarmonicMeasurandCDC (Common harmonic measurand information) (Abstract Class)

This abstract supertype groups the configuration attributes for common data classes that provide harmonic and interharmonic measurand information. Attributes are used to configure arrays of harmonic values in concrete CDCs ('har[]' in H MV, {'phsAHar', 'phsBHar', 'phsCHar', 'neutHar', 'netHar', 'resHar'} in HWYE, and {'phsABHar', 'phsBCHar', 'phsCAHar'}) in HDEL).

An array (e.g., 'har[]') contains the harmonic and subharmonic ('evalTm' equals 'frequency' period), or the interharmonic values ('evalTm' differs from 'frequency' period). The first array element 'har[0]' contains the dc component, the further array elements 'har[1]'...'har[numHar-1]' contain the values for the harmonics 1...'numHar', respectively.

For the case of harmonic and subharmonic values ('evalTm' equals 'frequency' period), if 'numCyc' > 1, 'har[]' contains both harmonics, and subharmonics with their multiples. Array elements with the index that is a multiple of 2^{numCyc-1} are harmonics; all the others are subharmonics or multiple of subharmonics.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAanalogueInfo\HarmonicMeasurandCDC*.

Table 5.209 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
numHar	Number of harmonic and subharmonics, or interharmonic values that are to be returned as the array of values. Array index 0 refers to the dc component, therefore 'numHar' > 0. The maximum value can be calculated as follows: 'numHar' = $1 + 0.5 * ('smpRate' * 'frequency' * 'evalTm' * 2^{numCyc-1})$.	Integer See Clause 6.6.3.5.	[1]
numCyc	Number of cycles of power frequency, which are used for harmonic, subharmonic and interharmonic calculation. See 'numHar'.	Integer See Clause 6.6.3.5.	[1]
evalTm	Time window [ms] applied to interharmonic calculations. See 'numHar'.	Integer See Clause 6.6.3.5.	[1]
smpRate	Determines the highest possible harmonic or interharmonic detectable, according to the sampling theorem; the minimum value is 'smpRate' = $2 * 'frequency'$. 'smpRate' is the number of samples per nominal period. In the case of a dc system, it is number of samples per s.	Integer See Clause 6.6.3.5.	[0..1]
frequency	Nominal frequency of the power system (HVM, HDEL) or some other fundamental frequency (HWYE) [Hz].	Real See Clause 6.6.3.6.	[1]
hvRef	Reference type (i.e., ratio of harmonic to fundamental, to RMS or to absolute), of the values in 'CMV[i].instCVal.mag'.	HvReferenceKind See Clause 5.7.4.5.3.3.4.	[0..1]
rmsCyc	Number of cycles of power frequency used for the calculation of rms values.	Integer See Clause 6.6.3.5.	[0..1]

5.7.4.5.3.1.5. HVM (Harmonic value) (Class)

This class is a collection of values that represent the harmonic or interharmonic content of a process value.

NOTE Harmonics for a single circuit may have phase angles, but need no reference for the angle ('angRef'), since by convention the reference is always the fundamental frequency (index 1).

Parent Class: HarmonicMeasurandCDC (See Clause 5.7.4.5.3.1.4.)

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAanalogueInfo\HVM*.

Table 5.210 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
har	Array of harmonic and subharmonic, or the interharmonic values.	CMV See Clause 5.7.4.5.3.1.2.	[1..*]

5.7.4.5.3.1.6. V (Scalar measured value) (Class)

This common data class is used to represent scalar measured values.

Parent Class: RangeDeadbandCDC (See Clause 5.7.4.5.3.1.7.)

UML Element Location: *Model\UML Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\MV.*

Table 5.211 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
instMag	Magnitude of the instantaneous value. NOTE 'instMag' is optional from the perspective of the visibility of that value to the communication. The instantaneous value may be required for the internal behaviour of the function, e.g., to perform the deadband calculation for 'mag'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[0..1]
mag	Magnitude based on a deadband 'db' calculation from the instantaneous value 'instMag'. The value of 'mag' shall be updated to the current instantaneous value 'instMag' when the value has changed according to the configured deadband 'db'. If 'db'=0, 'mag'='instMag'. NOTE 1 This value is typically used to create reports for analogue values. Such a report sent "by exception" is not comparable to the transfer of sampled measured values as supported by the CDC SAV. NOTE 2 This 'mag' is not the same as 'Vector.mag'. In particular, the value 'CMV.instCVal.mag' is NOT a deadbanded value.	Analogue Value See Clause 5.7.4.5.3.2.1.	[1]
q	Quality of the data values 'instMag', 'mag', 'range'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change in one of the data values 'mag', 'range', or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
subMag	Value used to substitute 'instMag'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[0..1]
units	Unit for 'instMag', 'mag', 'subMag', 'rangeC'.	Unit See Clause 5.7.4.5.3.2.9.	[0..1]
sVC	Configuration for scaled value representation ('instMag', 'mag', 'subMag', 'range').	Scaled Value Config See Clause 5.7.4.5.3.2.8.	[0..1]
smpRate	Number of samples per nominal period that has been used to determine instantaneous value 'instMag'. In the case of a dc system, it is number of samples per s.	Integer See Clause 6.6.3.5.	[0..1]

5.7.4.5.3.1.7. RangeDeadbandCDC (Range and deadband) (Abstract Class)

This abstract supertype groups the attributes common to those common data classes that provide deadbanded values (in addition to instantaneous ones).

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\RangeDeadbandCDC*.

Table 5.212 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
range	Range in which the current instantaneous value is ('MV.instMag' or 'CMV.cVal.mag'). It may be used to report an event if its value changes (due to the change in the instantaneous value). See RangeConfig. NOTE The use of algorithms to filter events based on transition from one range to another is a local issue.	RangeKind See Clause 5.7.4.5.3.3.6.	[0..1]
db	Deadband is a configuration parameter used to calculate deadbanded value ('MV.mag' or 'CMV.cVal.mag'). The value of 'db' shall represent the percentage of difference ('rangeC.max' - 'rangeC.min') in units of 0.001%. Therefore, 'db'=[0, 100'000], corresponding to [0%, 100%], respectively. If an integral calculation is used to determine the deadbanded value, the value of 'db' shall be represented as 0,001% s. Value 0 shall suppress reporting events on the analog value, so that only changes of the 'range' value will lead to events.	Integer See Clause 6.6.3.5.	[0..1]
zeroDb	Configuration parameter used to calculate the range around zero, where the deadbanded value ('MV.mag' or 'CMV.cVal.mag') will be forced to zero. The value of 'zeroDb' shall represent the percentage of difference ('rangeC.max' - 'rangeC.min') in units of 0,001%. Therefore, 'zeroDb'=[0, 100'000], corresponding to [0%, 100%], respectively.	Integer See Clause 6.6.3.5.	[0..1]
rangeC	Configuration for 'range'.	RangeConfig See Clause 5.7.4.5.3.2.7.	[0..1]

5.7.4.5.3.1.8. WYE (Phase-to-ground/neutral measured values of a three phase system) (Class)

This class is a collection of simultaneous measurements of values in a three phase system that represent phase to ground values ("wye").

Values for 'phsA', 'phsB', 'phsC', 'neut', 'net' and 'res' have been simultaneously acquired or determined, and their respective time stamps hold the same value 't.secondSinceEpoch'. It shall be assumed that any jitter between the acquisition times dedicated for these values is neglectable. The jitter for simultaneity shall be as indicated in any of the respective 't.timeQuality' attributes of the above values.

Attribute 'angRef' is used in place of individual 'angRef' attributes of 'phsA', 'phsB', 'phsC', 'neut', 'net' and 'res'.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueInfo\WYE*.

Table 5.213 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
phsA	Value of phase A.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
phsB	Value of phase B.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
phsC	Value of phase C.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
neut	Value of the measured phase neutral. If a direct measurement of this value is not available, it is acceptable to substitute an estimate computed by creating the algebraic sum of the instantaneous values of currents flowing through all live conductors ('phsA.instCVal'+phsB.instCVal'+phsC.instCVal'); in that case, 'neut'=res'.	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
net	Net current, as the algebraic sum of the instantaneous values of currents flowing through all live conductors and neutral point of the electrical installation('phsA.instCVal'+phsB.instCVal'+phsC.instCVal'+neut.instCVal').	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
res	Residual current, as the algebraic sum of the instantaneous values of currents flowing through all live conductors of the electrical installation('phsA.instCVal'+phsB.instCVal'+phsC.instCVal').	CMV See Clause 5.7.4.5.3.1.2.	[0..1]
angRef	Angle reference, indicating the quantity that is used as reference for the phase angles {'phsA', 'phsB', 'phsC', 'neut', 'net', 'res'}. 'instCVal.ang', instead of their own 'angRef'. For the indicated quantity, the fundamental frequency (index = 1) is used as reference by convention.	AngleReferenceKind See Clause 5.7.4.5.3.3.1.	[0..1]
phsToNeut	True indicates that this WYE instance is used for phase to neutral values instead of phase to ground values ('neut' always indicates the neutral to ground value).	Boolean See Clause 6.6.3.3.	[0..1]

5.7.4.5.3.1.9. Elements Defined in the CDCAnalogueSet Model

The following elements are defined in this model.

5.7.4.5.3.1.10. ASG (Analogue setting) (Class)

This common data class is used to represent analogue settings.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueSet\ASG.*

Table 5.214 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
setMag	The value of the analogue setting.	Analogue Value See Clause 5.7.4.5.3.2.1.	[0..1]
units	Unit for 'setMag', 'minVal', 'maxVal', 'stepSize'.	Unit See Clause 5.7.4.5.3.2.9.	[0..1]
sVC	Configuration for scaled value representation ('setMag', 'minVal', 'maxVal', 'stepSize').	Scaled Value Config See Clause 5.7.4.5.3.2.8.	[0..1]
minVal	Minimum setting for 'setMag'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[0..1]
maxVal	Maximum setting for 'setMag'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[0..1]
stepSize	Step between the individual values of 'setMag'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[0..1]

5.7.4.5.3.1.11. CSG (Curve shape setting) (Class)

This common data class is used to hold curve points and describe curve settings.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCAnalogueSet\CSG.*

Table 5.215 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
pointZ	Position of the curve on z-axis.	Real See Clause 6.6.3.6.	[0..1]
numPts	Actual size of 'crvPts[]'.	Integer See Clause 6.6.3.5.	[1]
crvPts	The array with the points specifying a curve shape.	Point See Clause 5.7.4.5.3.2.4.	[0..*]
xUnit	Unit of the x-axis of a curve.	Unit See Clause 5.7.4.5.3.2.9.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
yUnit	Unit of the y-axis of a curve.	Unit See Clause 5.7.4.5.3.2.9.	[1]
zUnit	Unit of the z-axis of a curve.	Unit See Clause 5.7.4.5.3.2.9.	[1]
maxPts	Maximum supported size for 'crvPts[]'.	Integer See Clause 6.6.3.5.	[1]
xD	Description of the value of the x-axis of a curve.	String See Clause 6.6.3.7.	[1]
xDU	Description of the value of the x-axis of a curve in Unicode.	String See Clause 6.6.3.7.	[0..1]
yD	Description of the value of the y-axis of a curve.	String See Clause 6.6.3.7.	[1]
yDU	Description of the value of the x-axis of a curve in Unicode.	String See Clause 6.6.3.7.	[0..1]
zD	Description of the value of the z-axis of a curve.	String See Clause 6.6.3.7.	[0..1]
zDU	Description of the value of the z-axis of a curve in Unicode.	String See Clause 6.6.3.7.	[0..1]

5.7.4.5.3.1.12. Elements Defined in the CDCControl Model

The following elements are defined in this model.

5.7.4.5.3.1.13. APC (Controllable analogue process value) (Class)

This common data class is used to represent analogue controls.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControl\APC.*

Table 5.216 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
origin	See 'SPC.origin'.	Originator See Clause 5.7.4.5.3.2.3.	[0..1]
ctlNum	See 'SPC.ctlNum'.	Integer See Clause 6.6.3.5.	[0..1]
mxVal	Current value of the controllable analogue process value; it can be deadbanded for reporting.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
q	Quality of the data value 'mxVal'.	Quality See Clause 5.7.4.5.3.2.6.	[0..1]
t	Timestamp of the last change in the data value 'mxVal', or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[0..1]
stSeld	See 'SPC.stSeld'.	Boolean See Clause 6.6.3.3.	[0..1]
subVal	Value used to substitute 'mxVal'.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[0..1]
units	Units of the data values: 'mxVal', 'subVal', 'minVal', 'maxVal', 'stepSize', 'ctlVal'.	Unit See Clause 5.7.4.5.3.2.9.	[0..1]
db	See 'RangeDeadbandCDC.db'. Used in case 'mxVal' is deadbanded for reporting.	Integer See Clause 6.6.3.5.	[0..1]
sVC	Configuration for scaled value representation ('mxVal', 'subVal', 'minVal', 'maxVal', 'stepSize') and for service parameter 'ctlVal'.	ScaledValueConfig See Clause 5.7.4.5.3.2.8.	[0..1]
minVal	Minimum setting for 'ctlVal'.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[0..1]
maxVal	Maximum setting for 'ctlVal'.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[0..1]
stepSize	Step between the individual values of 'ctlVal'.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[0..1]
ctlVal	Service parameter that determines the control activity, containing the new value for 'mxVal'.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[1]

5.7.4.5.3.1.14. DPC (Controllable double point) (Class)

This common data class is used to represent double point controls.

UML Element Location: *Model\UML Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCControlDPC*

Table 5.217 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
origin	See 'SPC.origin'.	Originator See Clause 5.7.4.5.3.2.3.	[0..1]
ctlNum	See 'SPC.ctlNum'.	Integer See Clause 6.6.3.5.	[0..1]
stVal	Status data value.	DpStatusKind See Clause 5.7.4.5.2.1.1.	[1]
q	Quality of the data value 'stVal'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change of the data value 'stVal' or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
stSeld	See 'SPC.stSeld'.	Boolean See Clause 6.6.3.3.	[0..1]
subVal	Value used to substitute 'stVal'.	DpStatusKind See Clause 5.7.4.5.2.1.1.	[0..1]
pulseConfig	See 'SPC.pulseConfig'.	PulseConfig See Clause 5.7.4.5.3.2.5.	[0..1]
ctlVal	Service parameter that determines the control activity, containing the new value for 'stVal' ('false' for off, 'true' for on).	Boolean See Clause 6.6.3.3.	[1]

5.7.4.5.3.1.15. ENC (Controllable enumerated status) (Abstract Class)

This common data class is used to represent integer controls with the value restricted to those in an enumeration.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\ODCCControl\ENC.*

Table 5.218 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
origin	See 'SPC.origin'.	Originator See Clause 5.7.4.5.3.2.3.	[0..1]
ctlNum	See 'SPC.ctlNum'.	Integer See Clause 6.6.3.5.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
stVal	Status data value.	EnumDA See Clause 5.7.4.5.2.2.1.	[1]
q	Quality of the data value 'stVal'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change of the data value 'stVal' or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
stSeld	See 'SPC.stSeld'.	Boolean See Clause 6.6.3.3.	[0..1]
subVal	Value used to substitute 'stVal'.	EnumDA See Clause 5.7.4.5.2.2.1.	[0..1]
ctlVal	Service parameter that determines the control activity, containing the new value for 'stVal'.	EnumDA See Clause 5.7.4.5.2.2.1.	[1]

5.7.4.5.3.1.16. SPC (Controllable single point) (Class)

This common data class is used to represent single point controls.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7-3\CommonDataClasses\CDCControl\SPC.*

Table 5.219 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
origin	Information related to the originator of the last change of the controllable data value.	Originator See Clause 5.7.4.5.3.2.3.	[0..1]
ctlNum	Control sequence number of the control service when the change of the status data value has been caused by a control. The only thing that the server shall do with 'ctlNum' is to include it in the responses to the control model and in the reports about a status change that is caused by a command.	Integer See Clause 6.6.3.5.	[0..1]
stVal	Status data value.	Boolean See Clause 6.6.3.3.	[0..1]
q	Quality of the data value 'stVal'.	Quality See Clause 5.7.4.5.3.2.6.	[0..1]
t	Timestamp of the last change of the data value 'stVal' or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
stSeld	True means that the controllable data is in the status "selected".	Boolean See Clause 6.6.3.3.	[0..1]
subVal	Value used to substitute 'stVal'.	Boolean See Clause 6.6.3.3.	[0..1]
pulseConfig	Used to configure the output pulse generated with the command, if applicable.	PulseConfig See Clause 5.7.4.5.3.2.5.	[0..1]
ctlVal	Service parameter that determines the control activity, containing the new value for 'stVal' ('false' for off, 'true' for on).	Boolean See Clause 6.6.3.3.	[1]

5.7.4.5.3.1.17. Elements Defined in the CDCDescription Model

The following elements are defined in this model.

5.7.4.5.3.1.18. DPL (Device name plate) (Class)

This common data class is used to identify entities like primary equipment or physical devices.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850-7_3(CommonDataClasses)\CDCDescription\DPL*.

Table 5.220 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
vendor	Name of the vendor.	String See Clause 6.6.3.7.	[1]
hwRev	Hardware revision.	String See Clause 6.6.3.7.	[0..1]
swRev	Software revision.	String See Clause 6.6.3.7.	[0..1]
serNum	Serial number.	String See Clause 6.6.3.7.	[0..1]
model	Vendor specific product name.	String See Clause 6.6.3.7.	[0..1]
location	Location where the equipment is installed.	String See Clause 6.6.3.7.	[0..1]
name	The name of the IED (if DPL is used in the context of a LPHD) or of a device like a circuit breaker (if used for the data EEName).	String See Clause 6.6.3.7.	[0..1]
owner	Owner of the device.	String See Clause 6.6.3.7.	[0..1]
ePSName	Name of electric power system the device is connected to.	String See Clause 6.6.3.7.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
primeOper	Primary operator of the device.	String See Clause 6.6.3.7.	[0..1]
secondOper	Secondary operator of the device.	String See Clause 6.6.3.7.	[0..1]
latitude	Geographical position of device in WGS84 coordinates – latitude.	Real See Clause 6.6.3.6.	[0..1]
longitude	Geographical position of device in WGS84 coordinates – longitude.	Real See Clause 6.6.3.6.	[0..1]
altitude	Geographical position of device in WGS84 coordinates – altitude.	Real See Clause 6.6.3.6.	[0..1]
mrID	Master resource ID – unique identification of an asset or device.	GlobalId See Clause 6.6.3.4.	[0..1]

5.7.4.5.3.1.19. LPL (Logical node name plate) (Class)

This common data class is used for nameplate information of logical nodes.

UML Element Location: *Model\UML Elements from External Sources\IEC61850_7_3\CommonDataClasses\CDCDescription\LPL*.

Table 5.221 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
vendor	Name of the vendor.	String See Clause 6.6.3.7.	[1]
swRev	Software revision.	String See Clause 6.6.3.7.	[1]
configRev	Uniquely identifies the configuration of a logical device instance. 'LLN0.NamPlt.configRev' has to be changed at least on any semantic change of the data model of the logical device, related to the client functionality. How this is detected and performed is left to the user. Also the semantics of 'configRev' for other logical nodes is left to the user. For further details see as well Annex C.	String See Clause 6.6.3.7.	[0..1]
paramRev	Uniquely identifies the parameter revision of a logical device or logical node instance. 'paramRev' has to be changed at least on any change of a parameter (FC= SE or FC=SP) within this logical device or logical node. How this is detected and performed is left to the user. For further details see as well Annex C. The value change of 'paramRev' shall be done as follows: - if the parameter change is done in the IED only through communication services or through the local HMI, this value shall be increased by one; - if the parameter change is done in the configuration file, this value shall be increased by 10 000.	Integer See Clause 6.6.3.5.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
valRev	Uniquely identifies the revision of the preconfiguration of configuration values (FC= CF) in a logical device or logical node instance through a SCL file. 'valRev' has to be changed at least on any change of preconfigured values within an SCL file for this logical device or logical node. How this is detected and performed is left to the user. For further details see as well Annex C. The value change of 'valRev' shall be done according to the same rules as for 'paramRev'.	Integer See Clause 6.6.3.5.	[0..1]
IdNs	Logical device name space. Scope of the standard specified with the attribute is LN class definition (CDC definition by reference). For details see IEC 61850-7-1.	String See Clause 6.6.3.7.	[0..1]
InNs	Logical node name space. Scope of the standard specified with the attribute is LN class definition (CDC definition by reference). For details see IEC 61850-7-1.	String See Clause 6.6.3.7.	[0..1]

5.7.4.5.3.1.20. Elements Defined in the CDCStatusInfo Model

The following elements are defined in this model.

5.7.4.5.3.1.21. BCR (Binary counter reading) (Class)

This common data class is used to represent binary counter reading.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\BCR.*

Table 5.222 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
actVal	Binary counter status represented as an integer value.	Integer See Clause 6.6.3.5.	[1]
frVal	Frozen binary counter status represented as an integer value.	Integer See Clause 6.6.3.5.	[0..1]
frTm	Time of the last counter freeze.	UTCDateTime See Clause 6.6.1.11.	[0..1]
q	Quality of the data values 'actVal', 'frVal'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change of the data value 'actVal' or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
units	Unit for value calculated as [value = 'actVal * pulsQty'], [value = 'frVal * pulsQty'].	Unit See Clause 5.7.4.5.3.2.9.	[0..1]
pulsQty	Magnitude of the counted value per count. Used together with 'actVal' and 'frVal' to calculate the value: [value = 'actVal * pulsQty'], [value = 'frVal * pulsQty'].	Real See Clause 6.6.3.6.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
frEna	Controls the freeze process. If true, freezing shall occur as specified in 'strTm', 'frPd' and 'frRs'; otherwise, no freezing shall occur.	Boolean See Clause 6.6.3.3.	[0..1]
strTm	Starting time of the freeze process. If the current time is later than 'strTm', the first freeze shall occur at the expiration of the next freeze interval 'frPd', computed from 'strTm' setting.	UTCDateTime See Clause 6.6.1.11.	[0..1]
frPd	Time interval between freeze operations [ms]. If value is 0, only a single freeze is performed at the time indicated in 'strTm'.	Integer See Clause 6.6.3.5.	[0..1]
frRs	If true, the counter is to be automatically reset to zero after each freezing process.	Boolean See Clause 6.6.3.3.	[0..1]

5.7.4.5.3.1.22. ENS (Enumerated status) (Abstract Class)

This common data class is used to represent integer status values with the value restricted to those in an enumeration.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\ENS.*

Table 5.223 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
stVal	Status data value.	EnumDA See Clause 5.7.4.5.2.2.1.	[1]
q	Quality of the data value 'stVal'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change of the data value 'stVal' or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
subVal	Value used to substitute 'stVal'.	EnumDA See Clause 5.7.4.5.2.2.1.	[0..1]

5.7.4.5.3.1.23. INS (Integer status) (Class)

This common data class is used to represent integer status values.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\INS.*

Table 5.224 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
stVal	Status data value.	Integer See Clause 6.6.3.5.	[1]
q	Quality of the data value 'stVal'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change of the data value 'stVal' or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
subVal	Value used to substitute 'stVal'.	Integer See Clause 6.6.3.5.	[0..1]
units	Unit for 'stVal', 'subVal'.	Unit See Clause 5.7.4.5.3.2.9.	[0..1]

5.7.4.5.3.1.24. SPS (Single point status) (Class)

This common data class is used to represent single point status values.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusInfo\SPS.*

Table 5.225 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
stVal	Status data value.	Boolean See Clause 6.6.3.3.	[1]
q	Quality of the data value 'stVal'.	Quality See Clause 5.7.4.5.3.2.6.	[1]
t	Timestamp of the last change of the data value 'stVal' or quality 'q'.	UTCDateTime See Clause 6.6.1.11.	[1]
subVal	Value used to substitute 'stVal'.	Boolean See Clause 6.6.3.3.	[0..1]

5.7.4.5.3.1.25. Elements Defined in the CDCStatusSet Model

The following elements are defined in this model.

5.7.4.5.3.1.26. CUG (Currency setting group) (Class)

This common data class is used to represent currency settings.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\CUG.*

Table 5.226 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
cur	The value of the currency setting.	CurrencyKind See Clause 6.6.2.1.	[0..1]

5.7.4.5.3.1.27. ENG (Enumerated status setting) (Abstract Class)

This common data class is used to represent integer settings with the value restricted to those in an enumeration.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\ENG.*

Table 5.227 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
setVal	The value of the status setting.	EnumDA See Clause 5.7.4.5.2.2.1.	[0..1]

5.7.4.5.3.1.28. ING (Integer status setting) (Class)

This common data class is used to represent integer settings.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\ING.*

Table 5.228 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
setVal	The value of the status setting.	Integer See Clause 6.6.3.5.	[0..1]
minVal	Minimum setting for 'setVal'.	Integer See Clause 6.6.3.5.	[0..1]
maxVal	Maximum setting for 'setVal'.	Integer See Clause 6.6.3.5.	[0..1]
stepSize	Step between the individual values of 'setVal'.	Integer See Clause 6.6.3.5.	[0..1]
units	Unit for 'setVal', 'minVal', 'maxVal', 'stepSize'.	Unit See Clause 5.7.4.5.3.2.9.	[0..1]

5.7.4.5.3.1.29. ORG (Object reference setting) (Class)

This common data class is used to represent object reference settings, such as object reference to the logical node of which the statistical data have been calculated.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\ORG.*

Table 5.229 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
setSrcRef	The value of the object reference setting used to reference for example a logical node instance or a data object instance.	ObjectReference See Clause 5.7.4.5.2.3.1.	[1]
setTstRef	The value of the test object reference setting, used as alternative to 'setSrcRef' when 'tstEna'=true for testing purpose.	ObjectReference See Clause 5.7.4.5.2.3.1.	[0..1]
setSrcCB	The value of the object reference to the control block, indicating from where the object referred to with 'setSrcRef' shall be received.	ObjectReference See Clause 5.7.4.5.2.3.1.	[0..1]
setTstCB	The value of the object reference to the control block indicating from where the object referred to with 'setTstRef' shall be received, used when 'tstEna'=true for testing purposes.	ObjectReference See Clause 5.7.4.5.2.3.1.	[0..1]
intAddr	Manufacturer-specific internal address.	String See Clause 6.6.3.7.	[0..1]
tstEna	Switch between original data source (as defined with 'setSrcRef' and 'setSrcCB') for a reference and test data source (as defined with 'setTstRef' and 'setTstCB').	Boolean See Clause 6.6.3.3.	[0..1]
purpose	Description of the purpose of the object reference.	String See Clause 6.6.3.7.	[0..1]

5.7.4.5.3.1.30. SPG (Single point setting) (Class)

This common data class is used to represent single point settings.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCStatusSet\SPG.*

Table 5.230 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
setVal	The value of the status setting (false is off, true is on).	Boolean See Clause 6.6.3.3.	[0..1]

5.7.4.5.3.2. Elements Defined in the ConstructedDAs Model

The following elements are defined in this model.

5.7.4.5.3.2.1. AnalogueValue (Analogue value) (Class)

This type is used to represent primary process measured value, pVal. An analogue value may be represented by applying an appropriate conversion to a basic type integer 'i' or to a floating point 'f'.

At least one of the attributes shall be used. If both 'i' and 'f' exist, the application in the server shall insure that both values remain consistent, i.e., that the pVal produced after appropriate conversion is the same within acceptable error. The latest value set by the communication service shall be used to update the other value. As an example, if 'xxx.f' is written, the application shall update 'xxx.i' accordingly.

NOTE The reason for both integer and floating point representation is so that IEDs without floating point capabilities are enabled to support analogue values. In this case, the scaleFactor and offset may be exchanged offline between clients and servers.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\AnalogueValue*.

Table 5.231 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
i	Integer representation of the measured value. The formula to convert between 'i' and the process value (pVal) shall be: $pVal = ('i' * ScaledValueConfig.scaleFactor) + ScaledValueConfig.offset$.	Integer See Clause 6.6.3.5.	[0..1]
f	Floating point representation of the measured value. The formula to convert between 'f' and the process value (pVal) shall be: $pVal = f * 10^{Unit.multiplier}$.	Real See Clause 6.6.3.6.	[0..1]

5.7.4.5.3.2.2. DetailQual (Detailed quality) «packed» (Class)

This type contains reason for an 'invalid' or 'questionable' value. See statechart diagram 'Quality.validity'.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\DetailQual*.

Table 5.232 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
overflow	If true, the value of the attribute to which the quality has been associated is beyond the capability of being represented properly (used for measurand information only). EXAMPLE A measured value may exceed the range that may be represented by the selected data type, for example the data type is a 16-bit unsigned integer and the value exceeds 65535.	Boolean See Clause 6.6.3.3.	[1]
outOfRange	If true, the attribute to which the quality has been associated is beyond a predefined range of values. The server shall decide if validity shall be set to invalid or questionable (used for measurand information only). EXAMPLE A measured value may exceed a predefined range, however the selected data type can still represent the value, for example the data type is a 16-bit unsigned integer, the predefined range is 0 to 40000, if the value is between 40001 and 65535 it is considered to be out of range.	Boolean See Clause 6.6.3.3.	[1]
badReference	If true, the value may not be a correct value due to a reference being out of calibration. The server shall decide if validity shall be set to invalid or questionable (used for measurand information and binary counter information only).	Boolean See Clause 6.6.3.3.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
oscillatory	To prevent overloading of event driven communication channels, it is desirable to detect and suppress oscillating (fast changing) binary inputs. If a signal changes in a defined time (t_{osc}) twice in the same direction (from 0 to 1 or from 1 to 0), then it shall be defined as an oscillation and this attribute shall be set to true. If a configured numbers of transient changes is detected, they shall be suppressed. In this time, the 'Quality.validity' shall be set to 'questionable'. If the signal is still in the oscillating state after the defined number of changes, the value shall be left in the state it was in when this flag was set. In this case, the 'Quality.validity' shall be changed from 'questionable' to 'invalid' and kept so as long as the signal is oscillating. If the configuration is such that all transient changes should be suppressed, the 'Quality.validity' shall be set immediately to 'invalid' and this flag to true (used for status information only).	Boolean See Clause 6.6.3.3.	[1]
failure	If true, a supervision function has detected an internal or external failure.	Boolean See Clause 6.6.3.3.	[1]
oldData	If true, an update is not made during a specific time interval. The value may be an old value that may have changed in the meantime. This specific time interval may be defined by an allowed-age attribute. NOTE "Fail silent" errors, where the equipment stops sending data, will cause setting this flag to true. In this case, the last received information was correct.	Boolean See Clause 6.6.3.3.	[1]
inconsistent	If true, an evaluation function has detected an inconsistency.	Boolean See Clause 6.6.3.3.	[1]
inaccurate	If true, the value does not meet the stated accuracy of the source. EXAMPLE The measured value of power factor may be noisy (inaccurate) when the current is very small.	Boolean See Clause 6.6.3.3.	[1]

5.7.4.5.3.2.3. Originator (Originator) (Class)

This type is used to describe the originator of the last change of a controllable data.

UML Element Location: *Model\UML Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedData\Originator*.

Table 5.233 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
orCat	Category indicates who/what caused the change of a controllable process value. If the initiator of a change is not known, this attribute shall be set to 'process'.	OriginatorCategoryKind See Clause 5.7.4.5.3.3.5.	[1]
orIdent	Identification of the originator of the value change. If the initiator of a change is not known, this value shall be set to null.	GlobalId See Clause 6.6.3.4.	[1]

5.7.4.5.3.2.4. Point (Point definition) (Class)

This type is used to represent points in a two or three dimensional coordinate system.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\Point.*

Table 5.234 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
xVal	X-axis value.	Real See Clause 6.6.3.6.	[1]
yVal	Y-axis value.	Real See Clause 6.6.3.6.	[1]
zVal	Z-axis value.	Real See Clause 6.6.3.6.	[0..1]

5.7.4.5.3.2.5. PulseConfig (Pulse configuration) (Class)

As the result of receiving an 'operate' service request, a pulsed output may be generated to the on or off input of a switching device. This type is used to configure that output pulse. If 'cmdQual=pulse', the duration of the pulse shall be defined with 'onDur', 'offDur' and 'numPls'. If 'cmdQual=persistent', the output stays in the state indicated in the 'operate' service request.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\PulseConfig.*

Table 5.235 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
cmdQual	Kind of control output.	ControlOutputKind See Clause 5.7.4.5.3.3.2.	[1]
onDur	Duration of the pulse [ms]. If set to 0, the duration is defined locally.	Integer See Clause 6.6.3.5.	[1]
offDur	Duration between two pulses [ms]. If set to 0, the duration is defined locally.	Integer See Clause 6.6.3.5.	[1]
numPls	Number of pulses that are generated.	Integer See Clause 6.6.3.5.	[1]

5.7.4.5.3.2.6. Quality (Quality) (Class)

Quality contains data that describe the quality of the data from the server. Quality of the data is also related to the mode of a logical node. Further details can be found in IEC 61850-7-4. The different quality attributes are *not* independent.

The default value shall be applied if the functionality of the related attribute is not supported. The mapping may specify to exclude the attribute from the message if it is not supported or if the default value applies.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\Quality*.

Table 5.236 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
validity	Validity of the value, as condensed information for the client. In case this value is not 'good', the reasons may be found in the 'detailQual'. See statechart diagram 'Quality.validity'.	ValidityKind See Clause 5.7.4.5.3.2.12.	[1]
detailQual	Describes reason in case 'validity' is not 'good'. See statechart diagram 'Quality.validity'.	DetailQual See Clause 5.7.4.5.3.2.2.	[1]
source	Defines the source of a value. NOTE 1 Substitution may be done locally or via the communication services. In the second case, specific attributes with a FC=SV are used. NOTE 2 There are various means to clear a substitution. As an example, a substitution that was done following an invalid condition may be cleared automatically if the invalid condition is cleared. However, this is a local issue and therefore not in the scope of this standard.	SourceKind See Clause 5.7.4.5.3.2.11.	[1]
test	If true, the value is a test value and not to be used for operational purposes. The processing of the test quality in the client shall be as described in IEC 61850-7-4. This bit shall be completely independent from the other bits within the quality descriptor.	Boolean See Clause 6.6.3.3.	[1]
operatorBlocked	If true, further update of the value has been blocked by an operator. The value shall be the information that was acquired before blocking. If this flag is set, then the 'detailQual.oldData' shall also be set. The operator shall use the data attribute 'CDC.blkEna' to block the update of the value. NOTE Both an operator as well as an automatic function may freeze communication updating as well as input updating. In both cases, 'detailQual.oldData' will be set. If the blocking is done by an operator, then this flag is set additionally, and an operator activity is required to clear the condition. EXAMPLE An operator may freeze the update of an input, to save the old value before the auxiliary supply is switched off.	Boolean See Clause 6.6.3.3.	[1]

5.7.4.5.3.2.7. RangeConfig (Range configuration) (Class)

This type is used to configure the limits that define the range of a measured value.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\RangeConfig*.

Table 5.237 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
hhLim	Setting for 'CDC.range' transition between 'high' and 'high-high'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[1]
hLim	Setting for 'CDC.range' transition between 'normal' and 'high'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[1]
lLim	Setting for 'CDC.range' transition between 'normal' and 'low'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[1]
lLim	Setting for 'CDC.range' transition between 'low' and 'low-low'.	Analogue Value See Clause 5.7.4.5.3.2.1.	[1]
min	The minimum process measurement for which 'AnalogueValue.[i,f]' is considered within process limits. If the value is lower, quality shall be set accordingly ('Quality.detailQual.outOfRange'=true => 'Quality.validity'='questionable'; see statechart diagram 'Quality.validity').	Analogue Value See Clause 5.7.4.5.3.2.1.	[1]
max	The maximum process measurement for which 'AnalogueValue.[i,f]' is considered within process limits. If the value is higher, quality shall be set accordingly ('Quality.detailQual.outOfRange'=true => 'Quality.validity'='questionable'; see statechart diagram 'Quality.validity').	Analogue Value See Clause 5.7.4.5.3.2.1.	[1]
limDb	When present, the value used to introduce a hysteresis in the calculation of 'range'. When a high/low limit has been crossed, 'range' is immediately set to the higher/lower value. However, 'range' is only set back to the lower/higher value when the value of the high limit minus/low limit plus 'limDb' has been crossed. The value shall represent the percentage between 'max' and 'min' in units of 0.001 %.	Integer See Clause 6.6.3.5.	[0..1]

5.7.4.5.3.2.8. ScaledValueConfig (Configuration of analogue value) (Class)

This type is used to configure the integer value representation of the analogue value. See AnalogueValue.i.

NOTE If a server does not support transmission of floating point values and needs to apply conversion, the client may retrieve these values from the SCL file.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\ScaledValueConfig.*

Table 5.238 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
scaleFactor	Scaling factor for integer representation of analogue value.	Real See Clause 6.6.3.6.	[1]
offset	Offset for integer representation of analogue value.	Real See Clause 6.6.3.6.	[1]

5.7.4.5.3.2.9. Unit (Unit definition) (Class)

This type is used to represent unit and multiplier information.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\Unit.*

Table 5.239 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
multiplier	Unit multiplier.	SiScaleCodeType See Clause 6.6.2.3.	[0..1]
SIUnit	SI unit of measure.	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.7.4.5.3.2.10. Vector (Vector definition) (Class)

This type is used to represent a complex value (phasor), with magnitude and angle.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\Vector.*

Table 5.240 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
mag	Magnitude of the complex value.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[1]
ang	Angle of the complex value [deg]; angle reference is defined in the context where this type is used.	AnalogueValue See Clause 5.7.4.5.3.2.1.	[0..1]

5.7.4.5.3.2.11. SourceKind (Source) (Enumeration)

Source of a value.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\SourceKind*.

Table 5.241 – SourceKind Enumerated Values

Enumerated Value	Definition
process	The value is provided by an input function from the process I/O or is calculated from some application function.
substituted	The value is provided by an operator input or by an automatic source.

5.7.4.5.3.2.12. ValidityKind (Validity) (Enumeration)

This type is used as validity of a value. See statechart diagram 'Quality.validity'.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\ConstructedDAs\ValidityKind*.

Table 5.242 – ValidityKind Enumerated Values

Enumerated Value	Definition
good	Supervision function has detected no abnormal condition of either the acquisition function or the information source.
invalid	Supervision function has detected an abnormal condition of the acquisition function or the information source (missing or non-operating updating devices). The value shall not be defined under this condition. It shall be used to indicate to the client that the value may be incorrect and shall not be used. EXAMPLE If an input unit detects an oscillation of one input it will mark the related information as invalid.
questionable	Supervision function has detected any abnormal behaviour. However, the value could still be valid. It is client's responsibility to determine whether the values should be used.

5.7.4.5.3.3. Elements Defined in the DAEnums Model

The following elements are defined in this model.

5.7.4.5.3.3.1. AngleReferenceKind (Enumeration)

An enumeration defining the quantity that is used as reference for the phase angles.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums\AngleReferenceKind*.

Table 5.243 – AngleReferenceKind Enumerated Values

Enumerated Value	Definition
Va	Angle reference is V phase a.
Vb	Angle reference is V phase b.
Vc	Angle reference is V phase c.
Aa	Angle reference is Amps phase a.
Ab	Angle reference is Amps phase b.
Ac	Angle reference is Amps phase c.

Enumerated Value	Definition
Vab	Angle reference is Volts phase a-b.
Vbc	Angle reference is Volts phase b-c.
Vca	Angle reference is Volts phase c-a.
Vother	Angle reference is Volts other.
Aother	Angle reference is Amps other.
Synchrophasor	Reference of the angle is as defined in IEEE C37.118, clause 4.2.

5.7.4.5.3.3.2. ControlOutputKind (Enumeration)

An enumeration defining the kind of control output provided.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums\ControlOutputKind.*

Table 5.244 - ControlOutputKind Enumerated Values

Enumerated Value	Definition
pulse	Control output is a pulse output.
persistent	Control output is a persistent output.

5.7.4.5.3.3.3. CtlModelKind (Enumeration)

An enumeration defining the kind of control mechanism provided.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums\CtlModelKind.*

Table 5.245 - CtlModelKind Enumerated Values

Enumerated Value	Definition
status-only	The object is not controllable, only the services that apply to a status object are supported. The attribute 'ctlVal' does not exist.
direct-with-normal-security	Direct control with normal security according to IEC 61850-7-2.
sbo-with-normal-security	SBO (select before operate) control with normal security according to IEC 61850-7-2.
direct-with-enhanced-security	Direct control with enhanced security according to IEC 61850-7-2.
sbo-with-enhanced-security	SBO control with enhanced security according to IEC 61850-7-2.

5.7.4.5.3.3.4. HvReferenceKind (Enumeration)

Reference for harmonic values.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums\HvReferenceKind.*

Table 5.246 – HvReferenceKind Enumerated Values

Enumerated Value	Definition
fundamental	Fundamental frequency.
rms	RMS frequency.
absolute	Absolute frequency.

5.7.4.5.3.3.5. OriginatorCategoryKind (Enumeration)

An enumeration specifying the category of the originator that caused a change of a value.

UML Element Location: *Model\UML Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums\OriginatorCategoryKind.*

Table 5.247 – OriginatorCategoryKind Enumerated Values

Enumerated Value	Definition
not-supported	Originator category not supported.
bay-control	Control operation issued from an operator using a client located at bay level.
station-control	Control operation issued from an operator using a client located at station level.
remote-control	Control operation from a remote operator outside the substation (for example network control center).
automatic-bay	Control operation issued from an automatic function at bay level.
automatic-station	Control operation issued from an automatic function at station level.
automatic-remote	Control operation issued from an automatic function outside of the substation.
maintenance	Control operation issued from a maintenance/service tool.
process	Status change occurred without control action (for example external trip of a circuit breaker or failure inside the breaker).

5.7.4.5.3.3.6. RangeKind (Enumeration)

An enumeration that indicates the relationship of a measured value against configured settings.

UML Element Location: *Model\UML Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums\RangeKind.*

Table 5.248 – RangeKind Enumerated Values

Enumerated Value	Definition
normal	Range of value is normal.
high	Range of value is high.
low	Range of value is low.
high-high	Range of value exceeds high-high limit.
low-low	Range of value is lower than low-low limit.

5.7.4.5.3.3.7. SIUnitKind (Enumeration)

An enumeration of SI units, derived from ISO 1000.

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_3\DAEnums\SIUnitKind.*

Table 5.249 – SI Unit Kind Enumerated Values

Enumerated Value	Definition
none	Dimensionless
m	Length
kg	Mass
s	Time
A	Current
K	Temperature
mol	Amount of substance
cd	Luminous intensity
deg	Plane angle
rad	Plane angle
sr	Solid angle
Gy	Absorbed dose (J/kg)
Bq	Activity (1/s)
°C	Relative temperature
Sv	Dose equivalent (J/kg)
F	Electric capacitance
C	Electric charge (A S)
S	Electric conductance (A/V)
H	Electric inductance (Wb/A)
V	Electric potential (W/A)
ohm	Electric resistance (V/A)
J	Energy (N m)
N	Force (kg m / s ²)
Hz	Frequency (1/s)
lx	Illuminance (Lm/m ²)
Lm	Luminous flux (cd sr)
Wb	Magnetic flux (V s)
T	Magnetic flux density (Wb / m ²)
W	Power (J/s)
Pa	Pressure (N / m ²)
m ²	Area
m ³	Volume
m/s	Velocity
m/s ²	Acceleration
m ³ /s	Volumetric flow rate
m/m ³	Fuel efficiency

Enumerated Value	Definition
M	Moment of mass (kg m)
kg/m ³	Density
m ² /s	Viscosity
W/m K	Thermal conductivity
J/K	Heat capacity
ppm	Concentration
1/s	Rotational speed
rad/s	Angular velocity
W/m ²	Insolation
J/m ²	Insolation energy
S/m	Electric conductivity
K/s	Temperature change rate
Pa/s	Pressure change rate
J/kg K	Specific heat
VA	Apparent power
Watts	Real power (I ² R)
VAr	Reactive power (V I sinTheta)
phi	Phase angle
cos(phi)	Power factor
Vs	Volt second (Ws/A)
V ²	Volt squared (W ² /A ²)
As	Amp second
A ²	Amp squared
A ² t	Amps squared time
VAh	Apparent energy
Wh	Real energy
VArh	Reactive energy
V/Hz	Magnetic flux
Hz/s	Rate of change of frequency
char	Number of characters
char/s	Baud
kgm ²	Turbine inertia
dB	Sound pressure level
J/Wh	Heat rate
W/s	Ramp rate
l/s	Flow rate
dBm	Power level relative to 1mW

5.7.4.5.4. Elements Defined in the IEC61850_7_4 Model

The following elements are defined in this model.

5.7.4.5.4.1. Elements Defined in the DerivedCDCs Model

The following elements are defined in this model.

5.7.4.5.4.1.1. ENCBehaviourMode (Mode behaviour controllable integer status CDC) (Class)

Possible values for 'stVal', 'ctlVal', 'subVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: Mod.

Parent Class: ENG (See Clause 5.7.4.5.3.1.15.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENCBehaviourMode.*

5.7.4.5.4.1.2. ENGCalcIntervalType (Calculation interval type) (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcIntvTyp, ClcRftTyp.

Parent Class: ENG (See Clause 5.7.4.5.3.1.27.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGCalcIntervalType.*

5.7.4.5.4.1.3. ENGCalcMethod (Calculation method) (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcMth.

Parent Class: ENG (See Clause 5.7.4.5.3.1.27.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGCalcMethod.*

5.7.4.5.4.1.4. ENGCalcMode (Calculation mode integer status setting CDC) (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcMod.

Parent Class: ENG (See Clause 5.7.4.5.3.1.27.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\DerivedCDCs\ENGCalcMode.*

5.7.4.5.4.1.5. ENGPfSign (Power factor sign integer status setting CDC) (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: PFSign.

Parent Class: ENG (See Clause 5.7.4.5.3.1.27.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4DerivedCDCs\ENGpfSign.*

5.7.4.5.4.1.6. ENGSTotalCalcMethod (Total apparent power calculation method integer status setting CDC) (Class)

Possible values for 'setVal', 'minVal', 'maxVal' are restricted by enumeration. Used for: ClcTotVA.

Parent Class: ENG (See Clause 5.7.4.5.3.1.27.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4DerivedCDCs\ENGSTotalCalcMethod.*

5.7.4.5.4.1.7. ENSAdjustment (Adjustment integer status CDC) (Class)

Possible values for 'stVal', 'subVal' are restricted by enumeration. Used for: AdjSt.

Parent Class: ENS (See Clause 5.7.4.5.3.1.22.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4DerivedCDCs\ENSAdjustment.*

5.7.4.5.4.1.8. ENSBehaviourMode (Mode behaviour integer status CDC) (Class)

Possible values for 'stVal', 'subVal' are restricted by enumeration. Used for: Beh, Mod.

Parent Class: ENS (See Clause 5.7.4.5.3.1.22.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4DerivedCDCs\ENSBehaviourMode.*

5.7.4.5.4.1.9. ENSHealth (Health state integer status CDC) (Class)

Possible values for 'stVal', 'subVal' are restricted by enumeration. Used for: EEHealth, Health, PhyHealth, MechHealth.

Parent Class: ENS (See Clause 5.7.4.5.3.1.22.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4DerivedCDCs\ENSHealth.*

5.7.4.5.4.1.10. SPCTransient (Transient controllable single point CDC) (Class)

An SPC whose 'stVal' is transient.

Not explicitly defined in the standard.

Transient Data - the status of data with this designation is momentary and must be logged or reported to provide evidence of their momentary state. The transient property of data only applies to boolean process data attributes (FC=ST) of that data. Transient data is identical to normal data, except that for the process state change from TRUE to FALSE no event is generated for reporting and for logging.

Parent Class: SPC (See Clause 5.7.4.5.3.1.16.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850_7_4\DerivedCDCs\SPCTransient.*

5.7.4.5.4.1.11. SPSTransient ((Transient) Single point status CDC) (Class)

An SPS whose 'stVal' is transient.
Not explicitly defined in the standard.

Transient Data - the status of data with this designation is momentary and must be logged or reported to provide evidence of their momentary state. The transient property of data only applies to boolean process data attributes (FC=ST) of that data. Transient data is identical to normal data, except that for the process state change from TRUE to FALSE no event may be generated for reporting and for logging.

Parent Class: SPS (See Clause 5.7.4.5.3.1.24.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850_7_4\DerivedCDCs\SPSTransient.*

5.7.4.5.4.2. Elements Defined in the LogicalNodeEnums Model

The following elements are defined in this model.

5.7.4.5.4.2.1. AdjustmentKind (Enumeration)

An enumeration of an adjustment status.

UML Element Location: *Model\Model Elements from External Sources\IEC61850_7_4\LogicalNodeEnums\AdjustmentKind.*

Table 5.250 - AdjustmentKind Enumerated Values

Enumerated Value	Definition
Completed	Adjustment has completed.
Cancelled	Adjustment has been cancelled.
New adjustments	Adjustment has accepted new adjustment.
Under way	Adjustment is under way.

5.7.4.5.4.2.2. BehaviourModeKind (Enumeration)

Switching between the modes (Mod.stVal) should only happen as a result of an operator command to the data object Mod. Mod and Beh are always accessible by the services. The communication services for the data object Mod do not care about the status of the Beh of the LN.

Logical nodes should process receiving data according to their quality information:

- processed as valid means that the application should react according to the quality and the behaviour of the LN.
- processes as invalid means the application should react as if the quality of the data had been invalid.
- processed as blocked means that the application should decide how to react, besides no process-related action based on the value is performed.
- statements "processed" and "not processed" don't belong to communication services and therefore no quality bit can be evaluated.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\BehaviourModeKind.*

Table 5.251 – BehaviourModeKind Enumerated Values

Enumerated Value	Definition
on	Normal enabled state. All communication services work and get updated values. • Function behind LN: on • Output to the process (switchgear) via a non-IEC 61850 link (for example wire, typical for X..., Y... and GGIO LNs): yes • Output of FC ST, MX (issued independently from Beh): value and q are relevant • Response to normal command from client: a+ (pos. ack.) • Response to test command from client: a- (neg. ack.) • Incoming data with q='normal': processed as valid • Incoming data with q='operatorBlocked': processed as blocked • Incoming data with q='test': processed as valid • Incoming data with q='test'+operatorBlocked': processed as invalid • Incoming data with q='invalid': processed as invalid • Non-IEC 61850 binary (relay, contact) inputs and analogue (instrument transformer) inputs: processed
on-blocked	Process is passively supervised. No output data (digital by relays or analogue setting) will be issued to the process. All communication services work and get updated values. Data objects will be transmitted with quality 'operatorBlocked'. • Function behind LN: on • Output to the process (switchgear) via a non-IEC 61850 link (for example wire, typical for X..., Y... and GGIO LNs): no • Output of FC ST, MX (issued independently from Beh): value is relevant, q='operatorBlocked' • Response to normal command from client: a- (neg. ack.) • Response to test command from client: a- (neg. ack.) • Incoming data with q='normal': processed as valid • Incoming data with q='operatorBlocked': processed as blocked • Incoming data with q='test': processed as invalid • Incoming data with q='test'+operatorBlocked': processed as invalid • Incoming data with q='invalid': processed as invalid • Non-IEC 61850 binary (relay, contact) inputs and analogue (instrument transformer) inputs: processed
test	Function is operated but results are indicated as test results. All communication services work and get updated values. Data objects will be transmitted with quality 'test'. Control commands with quality 'test' will be accepted only by LNs in 'test' or 'test/blocked' mode. Processed as valid means that the application should react in the manner what is foreseen for 'test'. • Function behind LN: on • Output to the process (switchgear) via a non-IEC 61850 link (for example wire, typical for X..., Y... and GGIO LNs): yes • Output of FC ST, MX (issued independently from Beh): value is relevant, q='test' • Response to normal command from client: a- (neg. ack.)

Enumerated Value	Definition
	• Response to test command from client: a+ (pos. ack.) • Incoming data with q='normal': processed as valid • Incoming data with q='operatorBlocked': processed as blocked • Incoming data with q='test': processed as valid • Incoming data with q='test'+operatorBlocked': processed as blocked • Incoming data with q='invalid': processed as invalid • Non-IEC 61850 binary (relay, contact) inputs and analogue (instrument transformer) inputs: processed
test/blocked	Function is operated in test mode, but with no impact to the process. No output data (digital by relays or analogue setting) will be issued to the process. All communication services work and get updated values. Data objects will be transmitted with quality 'test'. Control commands with quality test will be accepted only by LNs in 'test' or 'test/blocked' mode. • Function behind LN: off • Output to the process (switchgear) via a non-IEC 61850 link (for example wire, typical for X..., Y... and GGIO LNs): no • Output of FC ST, MX (issued independently from Beh): value is relevant, q='test'+operatorBlocked' • Response to normal command from client: a- (neg. ack.) • Response to test command from client: a+ (pos. ack.) • Incoming data with q='normal': processed as valid • Incoming data with q='operatorBlocked': processed as blocked • Incoming data with q='test': processed as valid • Incoming data with q='test'+operatorBlocked': processed as blocked • Incoming data with q='invalid': processed as invalid • Non-IEC 61850 binary (relay, contact) inputs and analogue (instrument transformer) inputs: processed
off	Function is inactive but shows its configuration capability. No process output is possible. No control command should be acknowledged (negative response). Only the data object Mod and Beh should be accessible by the services. • Function behind LN: off • Output to the process (switchgear) via a non-IEC 61850 link (for example wire, typical for X..., Y... and GGIO LNs): no • Output of FC ST, MX (issued independently from Beh): value is irrelevant, q='invalid' • Response to normal command from client: a- (neg. ack.) • Response to test command from client: a- (neg. ack.) • Incoming data with q='normal': not processed • Incoming data with q='operatorBlocked': not processed • Incoming data with q='test': not processed • Incoming data with q='test'+operatorBlocked': not processed • Incoming data with q='invalid': not processed

Enumerated Value	Definition
	Non-IEC 61850 binary (relay, contact) inputs and analogue (instrument transformer) inputs: not processed

5.7.4.5.4.2.3. CalcIntervalTypeKind (Enumeration)

An enumeration of a calculation interval frequencies.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\CalcIntervalTypeKind.*

Table 5.252 - CalcIntervalTypeKind Enumerated Values

Enumerated Value	Definition
MS	Calculation interval is millisecond.
PER_CYCLE	Calculation interval is per cycle.
CYCLE	Calculation interval is cycle.
DAY	Calculation interval is one day.
WEEK	Calculation interval is one week.
MONTH	Calculation interval is one month.
YEAR	Calculation interval is one year.
EXTERNAL	Calculation interval is externally defined.

5.7.4.5.4.2.4. CalcMethodKind (Enumeration)

An enumeration of calculation methods.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\CalcMethodKind.*

Table 5.253 - CalcMethodKind Enumerated Values

Enumerated Value	Definition
UNSPECIFIED	Calculation of the analogue values is unspecified.
TRUE_RMS	All analogue values are true r.m.s. values.
PEAK_FUNDAMENTAL	All analogue values are peak fundamental values.
RMS_FUNDAMENTAL	All analogue values are r.m.s. fundamental values.
MIN	All analogue values are minimum values.
MAX	All analogue values are maximum values.
AVG	All analogue values are average values.
SDV	All analogue values are standard deviation values.
PREDICTION	All analogue values are long term changes over time.
RATE	All analogue values are actual changes over time calculated with the actual and previous value.

5.7.4.5.4.2.5. CalcModeKind (Enumeration)

An enumeration of calculation modes.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\CalcModeKind.*

Table 5.254 - CalcModeKind Enumerated Values

Enumerated Value	Definition
TOTAL	Total time from the first start of the device/application until the current time.
PERIOD	Periodic time cycle.
SLIDING	Sliding window from now predefined window backwards.

5.7.4.5.4.2.6. HealthKind (Enumeration)

An enumeration of health status.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\HealthKind.*

Table 5.255 - HealthKind Enumerated Values

Enumerated Value	Definition
Ok	No problems, normal operation ("green").
Warning	Minor problems, but in safe operating mode ("yellow"). The exact meaning is a local issue, depending on the dedicated function/device.
Alarm	Severe problem, no operation possible ("red").

5.7.4.5.4.2.7. PfSignKind (Enumeration)

An enumeration of power factor sign kind.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodeEnums\PfSignKind.*

Table 5.256 - PfSignKind Enumerated Values

Enumerated Value	Definition
Active Power	Real power.
Lead/Lag	Lead/lag.

5.7.4.5.4.3. Elements Defined in the LogicalNodes Model

The following elements are defined in this model.

5.7.4.5.4.3.1. Elements Defined in the LNGroupF Model

The following elements are defined in this model.

5.7.4.5.4.3.2. DERComplexOutputRamping (Ramp function LN) (Class)

This logical node models a generic ramp. It is required due to the fact that the data object attributes of the ASG common data class do not contain all of the information required to achieve a full ramping function with divergent up and down trends (61850 LogicalNode FRMP).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupF\DERComplexOutputRamping.*

Table 5.257 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
AdjSt	'AdjSt.stVal' is the state of adjustment process.	ENSAdjustment See Clause 5.7.4.5.4.1.7.	[0..1]
Out	'Out.mag.f' is the analogue output of this function.	MV See Clause 5.7.4.5.3.1.6.	[1]
ErrTerm	'ErrTerm.mag.f' is the control loop termination error value, in case the function cannot be fulfilled because of external reasons. The value is a difference between set point and actual value.	MV See Clause 5.7.4.5.3.1.6.	[0..1]
RmpUp	'RmpUp.setMag.f' is the ramping rate on an upward trend.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
RmpDn	'RmpDn.setMag.f' is the ramping rate on a downward trend.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
StepPs	'StepPs.setMag.f' is the step size when turning from negative to positive direction.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]
StepNg	'StepNg.setMag.f' is the step size when turning from positive to negative direction.	ASG See Clause 5.7.4.5.3.1.10.	[0..1]

5.7.4.5.4.3.3. Elements Defined in the LNGroupL Model

The following elements are defined in this model.

5.7.4.5.4.3.4. DomainLN (Domain) (Abstract Class)

Abstract type, holding attributes common to all of all the domain logical nodes (corresponds to Common LN).

UML Element Location: *Model\UML Model Elements from External Sources\IEC61850\IEC61850_7_4LogicalNodes\LNGroup\LNDomainLN.*

Table 5.258 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
NameP/It	Name plate of the logical node.	LPL See Clause 5.7.4.5.3.1.19.	[0..1]
Beh	'DomainLN.Beh.stVal' is a read-only value, describing the behaviour of a domain logical node. It depends on the current operating mode of the logical node ('DomainLN.Mod.stVal'), and the current operating mode of the logical device that contains it ('LLN0.Mod.stVal'). Processing of the quality status (q) of the received data is the prerequisite for correct interpretation of 'DomainLN.Beh.stVal'. See statechart diagram 'DomainLN.behStateMachine'.	ENSBbehaviourMode See Clause 5.7.4.5.4.1.8.	[1]
Health	'Health.stVal' reflects the state of the logical node related hardware and software. More detailed information related to the source of the problem may be provided by specific attribute of the logical node.	ENSHHealth See Clause 5.7.4.5.4.1.9.	[0..1]
Blk	If 'Blk.stVal=true', this function has been dynamically blocked.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
Mod	'DomainLN.Mod.cttVal' is used by operator to initiate the change in the operating mode of the domain logical node. The resulting operating mode is 'DomainLN.Mod.stVal'. Processing of the quality status (q) of the received data is the prerequisite for correct interpretation of 'DomainLN.Mod.stVal'. See statechart diagram 'DomainLN.behStateMachine'.	ENCBbehaviourMode See Clause 5.7.4.5.4.1.1.	[0..1]
CmdBlk	'CmdBlk.cttVal=true' initiates blocking of control sequences and action triggers of all controllable data objects in this function. The resulting status is 'CmdBlk.stVal'. NOTE This data object may be used only in case the logical node does not already contain BlkOpn/BlkCls available (i.e., logical nodes with controllable data objects, other than XCBR, XSWI, YPSH and KVLV).	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
InRef	'InRef[i].setRef' is reference to the data object <i>i</i> bound to this input.	ORG See Clause 5.7.4.5.3.1.29.	[0..*]
BlkRef	'BlkRef[i].setRef' shows the receiving of dynamically blocking signal <i>i</i> .	ORG See Clause 5.7.4.5.3.1.29.	[0..*]

5.7.4.5.4.3.5. LLN0 (Logical device LN) (Class)

This logical node contains common information for logical devices (LDs).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\UML Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupLLN0*.

Table 5.259 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
namPlt	Name plate of the logical device.	LPL See Clause 5.7.4.5.3.1.19.	[0..1]
Beh	Behaviour of the Logical Node. It can be either on, blocked, test, test/blocked, or off.	ENSBhaviourMode See Clause 5.7.4.5.4.1.8.	[1]
Health	'Health.stVal' reflects the state of the logical device related hardware and software. It is derived from the worst (most critical) value of the health attribute of all the domain logical nodes contained in the logical device: 'Health.stVal' = max('DomainLN[i].Health.stVal').	ENSHHealth See Clause 5.7.4.5.4.1.9.	[0..1]
Blk	If 'Blk.stVal=true', the function of the logical device has been dynamically blocked.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
Mod	'Mod.ctlVal' is used by operator to initiate the change in the operating mode of the logical device, which subsequently influences behaviour of the contained domain logical nodes ('DomainLN.Beh.stVal'). The resulting operating mode is 'DomainLN.Mod.stVal'. Processing of the quality status (q) of the received data is the prerequisite for correct interpretation of 'LLN0.Mod.stVal'. See statechart diagram 'DomainLN.behStateMachine'.	ENCBhaviourMode See Clause 5.7.4.5.4.1.1.	[0..1]
CmdBlk	'CmdBlk.ctlVal=true' initiates blocking of control sequences and action triggers of all controllable data objects in the logical device. The resulting status is 'CmdBlk.stVal'.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
InRef	'InRef[i].setRef' is reference to the data object <i>i</i> bound to this input.	ORG See Clause 5.7.4.5.3.1.29.	[0..*]
BlkRef	'BlkRef[i].setRef' shows the receiving of dynamically blocking signal <i>i</i> .	ORG See Clause 5.7.4.5.3.1.29.	[0..*]
OpTmh	'OpTmh.stVal' is the operation time [h], of a physical device (???) since start of the operation.	INS See Clause 5.7.4.5.3.1.23.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
LockKey	If 'LockKey.stVal=true', the operation of the complete logical device has been switched (from remote) to local. This changeover is always done locally with a physical key or toggle switch, which may have a set of contacts from which the position can be read.	SPS See Clause 5.7.4.5.3.1.24.	[1]
Loc	If 'Loc.stVal=true', the control behaviour is allowed at this level. At bay level, "local" means operation from the bay unit and "remote" means operation from a station unit. At process level, "local" means direct operation on the process device (for example on a circuit breaker) and "remote" means operation from a bay unit. If in a logical device 'LLN0.Loc.stVal' is in contradiction to the any contained 'DomainLN.Loc', "local" is always dominant. (See also Annex B)	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
LocSta	'LocSta.ctfVal=true' switches control authority to station level and disables control from remote. 'LocSta.ctfVal=false' does the opposite, i.e., it switches control authority from local to remote. The resulting status is 'LocSta.stVal'.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
Diag	'Diag.ctfVal=true' initiates running the diagnostics. The resulting status is 'Diag.stVal'.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
LEDRs	'LEDRs.ctfVal=true' initiates the resetting of all light emitting diodes. The resulting status is 'LEDRs.stVal'.	SPCTransient See Clause 5.7.4.5.4.1.10.	[0..1]
GrRef	'GrRef.setRef' is the reference to a higher level logical device.	ORG See Clause 5.7.4.5.3.1.29.	[0..1]
MltLev	If 'MltLev.setVal=true', authority control from multiple levels is allowed, otherwise no other control level is allowed (see Annex B).	SPG See Clause 5.7.4.5.3.1.30.	[0..1]

5.7.4.5.4.3.6. LPHD (Physical device LN) (Class)

This logical node contains common information for physical devices (IEDs).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\UML Elements from External Sources\IEC61850\IEC61850_7_4LogicalNodes\LNGroupLPHD*.

Table 5.260 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
PhyNam	Physical device (IED) name plate.	DPL See Clause 5.7.4.5.3.1.18.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
PhyHealth	'PhyHealth.stVal' reflects the state of the IED related hardware and software.	ENSHHealth See Clause 5.7.4.5.4.1.9.	[1]
OutOv	If 'OutOv.stVal=true', an output communications buffer overflow occurred. Consequently, the annunciations important for the communication may be lost. A general interrogation is recommended or an integrity scan is started automatically.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
Proxy	If 'Proxy.stVal=true', the IED is a proxy.	SPS See Clause 5.7.4.5.3.1.24.	[1]
InOv	If 'InOv.stVal=true', an input communications buffer overflow occurred. Consequently, the annunciations important for the communication may be lost. A general interrogation is recommended or an integrity scan is started automatically.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
NumPwrUp	'NumPwrUp.stVal' is the number of power up operations of the IED since the last reset.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
WrmStr	'WrmStr.stVal' is the number of warm start operations of the IED since the last reset.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
WacTrg	'WacTrg.stVal' is the number of times the watchdog circuit has reset the IED since the counter reset.	INS See Clause 5.7.4.5.3.1.23.	[0..1]
PwrUp	If 'PwrUp.stVal=true', an IED power up has been detected.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
PwrDn	If 'PwrDn.stVal=true', an IED power down has been detected.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
PwrSupAlm	If 'PwrSupAlm.stVal=true', an alarm from the external power supply occurred. It may be an external contact. It refers always to the local power supply of the IED, and not to the health (EEHealth) of the complete external supply system.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
Sim	'Sim.ctlVal=true' causes the IED to use simulated GOOSE or sampled measurand values instead of original ones since they are first received.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
RsStat	'RsStat.ctlVal=true' initiates the resetting of the IED security statistics.	SPCTransient See Clause 5.7.4.5.4.1.10.	[0..1]

5.7.4.5.4.3.7. Elements Defined in the LNGroupM Model

The following elements are defined in this model.

5.7.4.5.4.3.8. DERMeasurement (Measurement) (Class)

This logical node is used for calculation of currents, voltages, powers and impedances in a three-phase system. The main use is for operative applications (61850 LogicalNode MMXU).

Parent Class: DomainLN (See Clause 5.7.4.5.4.3.4.)

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_4\LogicalNodes\LNGroupM\DERMeasurement.*

Table 5.261 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
ClcExp	If 'ClcExp.stVal=true', the calculation period of a statistical logical node has expired.	SPSTransient See Clause 5.7.4.5.4.1.11.	[0..1]
NamPlt	Name plate of the logical node.	LPL See Clause 5.7.4.5.3.1.19.	[0..1]
TotW	'TotW.mag.f' is the total three-phase real power [W].	MV See Clause 5.7.4.5.3.1.6.	[0..1]
Beh	'DomainLN.Beh.stVal' is a read-only value, describing the behaviour of a domain logical node. It depends on the current operating mode of the logical node ('DomainLN.Mod.stVal'), and the current operating mode of the logical device that contains it ('LLN0.Mod.stVal'). Processing of the quality status (q) of the received data is the prerequisite for correct interpretation of 'DomainLN.Beh.stVal'. See statechart diagram 'DomainLN.behStateMachine'.	ENSBehaviourMode See Clause 5.7.4.5.4.1.8.	[1]
ClcStr	'ClcStr.ctfVal=true' enables the start calculation of statistical data, either immediately or at 'operTm' of the control model (if set). Resulting status is 'ClcStr.stVal'.	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
TotVAr	'TotVAr.mag.f' is the total three-phase reactive power [VAr].	MV See Clause 5.7.4.5.3.1.6.	[0..1]
ClcMth	'ClcMth.setVal' is the kind of statistical calculation, specifying how the data attributes that represent analogue values have been calculated. The calculation method shall be the same for all data objects of the logical node instance.	ENGCalcMethod See Clause 5.7.4.5.4.1.3.	[0..1]
Health	'Health.stVal' reflects the state of the logical node related hardware and software. More detailed information related to the source of the problem may be provided by specific attribute of the logical node.	ENSHealth See Clause 5.7.4.5.4.1.9.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
TotVA	'TotVA.mag.f' is the total three-phase apparent power [VA].	MV See Clause 5.7.4.5.3.1.6.	[0..1]
Blk	If 'Blk.stVal=true', this function has been dynamically blocked.	SPS See Clause 5.7.4.5.3.1.24.	[0..1]
ClcMod	'ClcMod.setVal' is the calculation mode.	ENGCalcMode See Clause 5.7.4.5.4.1.4.	[0..1]
TotPF	'TotPF.mag.f' is the average three-phase power factor.	MV See Clause 5.7.4.5.3.1.6.	[0..1]
ClcIntvTyp	'ClcIntvTyp.setVal' is the calculation interval type.	ENGCalcIntervalType See Clause 5.7.4.5.4.1.2.	[0..1]
Hz	'Hz.mag.f' is the frequency [Hz].	MV See Clause 5.7.4.5.3.1.6.	[0..1]
Mod	'DomainLN.Mod.ctVal' is used by operator to initiate the change in the operating mode of the domain logical node. The resulting operating mode is 'DomainLN.Mod.stVal'. Processing of the quality status (q) of the received data is the prerequisite for correct interpretation of 'DomainLN.Mod.stVal'. See statechart diagram 'DomainLN.behStateMachine'.	ENCBehaviourMode See Clause 5.7.4.5.4.1.1.	[0..1]
ClcIntvPer	'ClcIntvPer.setVal' is the number of units to consider to calculate the calculation interval duration, in case 'ClcIntvTyp.setVal' other than 'external'.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
CmdBlk	'CmdBlk.ctVal=true' initiates blocking of control sequences and action triggers of all controllable data objects in this function. The resulting status is 'CmdBlk.stVal'. NOTE This data object may be used only in case the logical node does not already contain BlkOpn/BlkCls available (i.e., logical nodes with controllable data objects, other than XCBR, XSWI, YPSH and KVLV).	SPC See Clause 5.7.4.5.3.1.16.	[0..1]
PPV	'PPV' holds the phase to phase voltage phasors [V], as follows: 'PPV.[phsAB.phsBC.phsCA].cVal.[mag.ang].f' are phase voltage phasors; 'PPV.angRef.stVal' is the quantity used as reference for the phase angles of the 3 phasors above.	DEL See Clause 5.7.4.5.3.1.3.	[0..1]
InRef	'InRef[i].setRef' is reference to the data object <i>i</i> bound to this input.	ORG See Clause 5.7.4.5.3.1.29.	[0..*]

Attribute Name	Description	Attribute Type	Multiplicity
NumSubIntv	'NumSubIntv.setVal' is the number of sub-intervals contained in a calculation period interval duration.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
PhV	'PhV.[phsA,phsB,phsC].cVal' hold phase to ground (line) voltage phasors [V]. For structure, see 'A'.	WYE See Clause 5.7.4.5.3.1.8.	[0..1]
A	'A' holds the currents [A] for three phases, as follows: 'A.[phsA,phsB,phsC].cVal.[mag,ang].f' are phase current phasors; 'A.neut.cVal.[mag,ang].f' is neutral current phasor; 'A.net.cVal.[mag,ang].f' is net current phasor (phsA+phsB+phsC+neut); 'A.res.cVal.[mag,ang].f' is residual current phasor (phsA+phsB+phsC); 'A.angRef.stVal' is the quantity used as reference for the phase angles of the 6 phasors above.	WYE See Clause 5.7.4.5.3.1.8.	[0..1]
BlkRef	'BlkRef[i].setRef' shows the receiving of-dynamically blocking signal <i>i</i> .	ORG See Clause 5.7.4.5.3.1.29.	[0..*]
ClcRfTyp	'ClcRfTyp.setVal' is the kind of refresh interval.	ENGCalcIntervalType See Clause 5.7.4.5.4.1.2.	[0..1]
ClcRfPer	'ClcRfPer.setVal' is the number of units to consider to calculate the refreshment interval duration in case 'ClcIntvTyp.setVal' is other than 'external'.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
W	'W.[phsA,phsB,phsC].cVal' hold the real powers [W] per phase. For structure, see 'A'.	WYE See Clause 5.7.4.5.3.1.8.	[0..1]
ClcSrc	'ClcSrc.setRef' is the object reference to the logical node whose analogue data attributes are used to calculate the values contained in this logical node instance.	ORG See Clause 5.7.4.5.3.1.29.	[0..1]
VAr	'VAr.[phsA,phsB,phsC].cVal' hold the reactive powers per phase [VAr]. For structure, see 'A'.	WYE See Clause 5.7.4.5.3.1.8.	[0..1]
ClcNxTmms	'ClcNxTmms.setVal' is the remaining time [ms] up to the end of the current calculation interval.	ING See Clause 5.7.4.5.3.1.28.	[0..1]
VA	'VA.[phsA,phsB,phsC].cVal' hold the apparent powers per phase [VA]. For structure, see 'A'.	WYE See Clause 5.7.4.5.3.1.8.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
InSyn	'InSyn.setRef' is the object reference to the source of the external synchronization signal for the calculation interval.	ORG See Clause 5.7.4.5.3.1.29.	[0..1]
PF	'PF.[phsA,phsB,phsC].cVal' hold the phase to ground power factor phasors. For structure, see 'A'.	WYE See Clause 5.7.4.5.3.1.8.	[0..1]
Z	'Z.[phsA,phsB,phsC].cVal' hold the impedances per phase. For structure, see 'A'.	WYE See Clause 5.7.4.5.3.1.8.	[0..1]
AvAPhs	'AvAPhs.mag.f' is the arithmetic average of the magnitude of current of the 3 phases: average(Ia,Ib,Ic).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
AvPPVPhs	'AvPPVPhs.mag.f' is the arithmetic average of the magnitude of phase to phase voltage of the 3 phases: average(PPVa, PPVb, PPVc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
AvPhVPhs	'AvPhVPhs.mag.f' is the arithmetic average of the magnitude of phase to reference voltage of the 3 phases: average(PhVa, PhVb, PhVc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
AvWPhs	'AvWPhs.mag.f' is the arithmetic average of the magnitude of active power of the 3 phases: average(Wa, Wb, Wc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
AvVAPhs	'AvVAPhs.mag.f' is the arithmetic average of the magnitude of apparent power of the 3 phases: average(VAa, VAb, VAc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
AvVArPhs	'AvVArPhs.mag.f' is the arithmetic average of the magnitude of reactive power of the 3 phases: average(VAra, VArb, VArc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
AvPFPPhs	'AvPFPPhs.mag.f' is the arithmetic average of the magnitude of power factor of the 3 phases.: average(PFa, PFb, PFc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
AvZPhs	'AvZPhs.mag.f' is the arithmetic average of the magnitude of impedance of the 3 phases.: average(Za, Zb, Zc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MaxAPhs	'MaxAPhs.mag.f' is the maximum magnitude of current of the 3 phases: max(Ia,Ib,Ic).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MaxPPVPhs	'MaxPPVPhs.mag.f' is the maximum magnitude of phase to phase voltage of the 3 phases: max(PPVa, PPVb, PPVc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
MaxPhVPhs	'MaxPhVPhs.mag.f' is the maximum magnitude of phase to reference voltage of the 3 phases: max(PhVa, PhVb, PhVc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MaxWPhs	'MaxWPhs.mag.f' is the maximum magnitude of active power of the 3 phases: max(Wa, Wb, Wc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MaxVAPhs	'MaxVAPhs.mag.f' is the maximum magnitude of apparent power of the 3 phases: max(VAa, VAb, VAc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MaxVArPhs	'MaxVArPhs.mag.f' is the maximum magnitude of rective power of the 3 phases: max(VArA, VArb, VArc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MaxPFPhs	'MaxPFPhs.mag.f' is the maximum magnitude of power factor of the 3 phases.: max(PFa, PFb, PFc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MaxZPhs	'MaxZPhs.mag.f' is the maximum magnitude of impedance of the 3 phases.: max(Za, Zb, Zc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MinAPhs	'MinAPhs.mag.f' is the minimum magnitude of current of the 3 phases: min(Ia, Ib, Ic).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MinPPVPhs	'MinPPVPhs.mag.f' is the minimum magnitude of phase to phase voltage of the 3 phases: min(PPVa, PPVb, PPVc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MinPhVPhs	'MinPhVPhs.mag.f' is the minimum magnitude of phase to reference voltage of the 3 phases: min(PhVa, PhVb, PhVc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MinWPhs	'MinWPhs.mag.f' is the minimum magnitude of active power of the 3 phases: min(Wa, Wb, Wc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MinVAPhs	'MinVAPhs.mag.f' is the minimum magnitude of apparent power of the 3 phases: min(VAa, VAb, VAc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MinVArPhs	'MinVArPhs.mag.f' is the minimum magnitude of rective power of the 3 phases: min(VArA, VArb, VArc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
MinPFPhs	'MinPFPhs.mag.f' is the minimum magnitude of power factor of the 3 phases.: min(PFa, PFb, PFc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
MinZPhs	'MinZPhs.mag.f' is the minimum magnitude of impedance of the 3 phases.: min(Za, Zb, Zc).	MV See Clause 5.7.4.5.3.1.6.	[0..1]
ClcTotVA	'ClcTotVA.setVal' is the calculation method used for total apparent power 'TotVA'.	ENGSTotalCalcMethod See Clause 5.7.4.5.4.1.6.	[0..1]
PFSign	'PFSign.setVal' is the sign convention for power factor 'PF' (and reactive power 'VAR').	ENGPFsign See Clause 5.7.4.5.4.1.5.	[0..1]

5.7.5. iec_cim_naesb_eui_model_20101111Update_20120802

This package, derived from NAESB REQ.18/WEQ.19, contains information elements used in the Meter Component, Energy Manager Component, FSGIM measurements, and exchange of energy usage information with external parties.

UML Package Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802*

5.7.5.1. NAESB PAP10 EUI

This is the NAESB REQ.18/WEQ.19 PAP10 Energy Usage Information Model. It was ratified in October 2010 and updated this year 2012.

UML Package Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI*

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

5.7.5.2. Elements Defined in the iec_cim_naesb_eui_model_20101111Update_20120802 Model

The following elements are defined in this model.

5.7.5.2.1. Elements Defined in the NAESEB PAP10 EUI Model

The following elements are defined in this model.

5.7.5.2.1.1. BaseCIM_CombinedVersion (Class)

The combined version denotes the versions of the subpackages that have been combined into the total CIM model. This is a convenience instead of having to look at each subpackage.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\BaseCIM_CombinedVersion.*

Table 5.262 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
date	Form is YYYY-MM-DD for example for January 5, 2009 it is 2009-01-05.	UTCDateTime See Clause 6.6.1.11.	[1]
version	Form is IEC61970CIMXXvYY_IEC61968CIMXXvYY combined where XX is the major CIM package version and the YY is the minor version, and different packages could have different major and minor versions. For example IEC61970CIM13v18_IEC61968CIM10v16 combined. Additional packages might be added in the future.	String See Clause 6.6.3.7.	[1]

5.7.5.2.1.2. Customer (Class)

Organization receiving services from ServiceSupplier.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Customer.*

Table 5.263 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.264 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	CustomerAgreements	All agreements of this customer.	CustomerAgreement See Clause 5.7.5.2.1.3.	[0..*]
Association	ServiceSuppliers	ServiceSuppliers associated with a Customer.	ServiceSupplier See Clause 5.7.5.2.1.23.	[1..*]
Association	UsagePoints	UsagePoints associated with a specific customer	UsagePoint See Clause 5.7.5.2.1.26.	[0..*]

5.7.5.2.1.3. CustomerAgreement (Class)

Agreement between the customer and the ServiceSupplier to pay for service at a specific service location. It provides for the recording of certain billing information about the type of service provided at the service location and is used during charge creation to determine the type of service.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\CustomerAgreement.*

Table 5.265 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.266 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	Customer	Customer for this agreement.	Customer See Clause 5.7.5.2.1.2.	[1]
Association	CustomerAuthorisations	CustomerAuthorisations provided by the Customer.	CustomerAuthorisation See Clause 5.7.5.2.1.4.	[0..*]
Association	ServiceDeliveryPoints	All service delivery points regulated by this customer agreement.	ServiceDeliveryPoint See Clause 5.7.5.2.1.22.	[0..*]

5.7.5.2.1.4. CustomerAuthorisation (Class)

Holds an authorization for access to specific user-private data granted to a third party service provider.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI Customer Authorisation.*

Table 5.267 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
authorizationServer	Contains the URI link to the authorization endpoint associated with this authorization.	AnyURI See Clause 6.6.3.2.	[0..1]
authorizedPeriod	Restricts access to requests or subscriptions within this date time interval.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]
validityInterval	Date and time interval this agreement is valid (from going into effect to termination).	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]
accessToken	Contains the access token associated with this authorization.	String See Clause 6.6.3.7.	[0..1]
publishedPeriod	Restricts access to the objects within the associated resource that were published within this date time interval.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]
resource	Contains the identifier of the resource, same as was specified in OAuth "scope".	AnyURI See Clause 6.6.3.2.	[0..1]
status	The status of this authorization. 0 - Revoked 1 - Active	Integer See Clause 6.6.3.5.	[0..1]

This element has the following connections to other elements:

Table 5.268 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	CustomerAgreements	CustomerAgreements represented by a CustomerAuthorisation.	CustomerAgreement See Clause 5.7.5.2.1.3.	[0..*]

5.7.5.2.1.5. ElectricPowerQualitySummary (Class)

A summary of power quality events. This information represents a summary of power quality information typically required by energy management systems. It is not intended to satisfy the detailed requirements of power quality monitoring. All values are as defined by measurementProtocol during the period and period. The standards typically also give ranges of allowed values. See referenced standards for definition, measurement protocol and period.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI Electric Power Quality Summary.*

Table 5.269 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
flickerPst	A measurement of long term Rapid Voltage Change	Real See Clause 6.6.3.6.	[0..1]
flickerPst	flickerPst is a value measured over 10 minutes that characterizes the likelihood that the voltage fluctuations would result in perceptible light flicker. A value of 1.0 is designed to represent the level that 50% of people would perceive flicker in a 60 watt incandescent bulb.	Real See Clause 6.6.3.6.	[0..1]
harmonicVoltage	A measurement of the Harmonic Voltage during the period. For DC, distortion is with respect to a signal of zero Hz.	Real See Clause 6.6.3.6.	[0..1]
longInterruptions	A count of Long Interruption events (as defined by measurementProtocol) during the summary interval period.	Integer See Clause 6.6.3.5.	[0..1]
mainsVoltage	A measurement of the Mains [Signaling] Voltage during the summary interval period.	Real See Clause 6.6.3.6.	[0..1]
measurementProtocol	A reference to the source used as the measurement protocol definition. e.g., "IEEE1519-2009", "EN50160"	String See Clause 6.6.3.7.	[0..1]
powerFrequency	A measurement of the power frequency during the summary interval period.	Real See Clause 6.6.3.6.	[0..1]
rapidVoltageChanges	A count of Rapid Voltage Change events during the summary interval period	Integer See Clause 6.6.3.5.	[0..1]
shortInterruptions	A count of Short Interruption events during the summary interval period	Integer See Clause 6.6.3.5.	[0..1]
summaryInterval	The relevant time reference for the ElectricPowerQualitySummary. This time interval may not be the same as any associated EMIntervalData time interval. For example, many of the statistics in the summary are running counts. The summaryInterval defines when the counts began.	UTCDateTimeInterval See Clause 6.6.1.4.	[1]
supplyVoltageDips	A count of Supply Voltage Dip events during the summary interval period	Integer See Clause 6.6.3.5.	[0..1]
supplyVoltageImbalance	A count of Supply Voltage Imbalance events during the summary interval period	Integer See Clause 6.6.3.5.	[0..1]
supplyVoltageVariations	A count of Supply Voltage Variations during the summary interval period	Integer See Clause 6.6.3.5.	[0..1]
tempOvervoltage	A count of Temporary Overvoltage events (as defined by measurementProtocol) during the summary interval period	Integer See Clause 6.6.3.5.	[0..1]

This element has the following connections to other elements:

Table 5.270 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	UsagePoint	UsagePoint for this ElectricPowerQualitySummary.	UsagePoint See Clause 5.7.5.2.1.26.	[1]

5.7.5.2.1.6. EndDeviceAsset (Class)

EndDeviceAsset that performs one or more end device functions. One type of EndDeviceAsset is a MeterAsset which can perform metering, load management, connect/disconnect, accounting functions, etc. Some EndDeviceAssets, such as ones monitoring and controlling air conditioner, refrigerator, pool pumps may be connected to a MeterAsset. All EndDeviceAssets may have communication capability defined by the associated ComFunction(s). An EndDeviceAsset may be owned by a consumer, an energy service provider, utility or otherwise.

There may be a related end device function that identifies a sensor or control point within a metering application or communications systems (e.g., water, gas, electricity). Some devices may use an optical port that conforms to the ANSI C12.18 standard for communications.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_nesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\EndDeviceAsset.*

Table 5.271 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.272 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	UsagePoint	Usage point to which this end device asset belongs.	UsagePoint See Clause 5.7.5.2.1.26.	[0..1]

5.7.5.2.1.7. EnergyUsageInformation (Class)

A collection of customer energy usage information. This class is a container, and has no attributes.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_nesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\EnergyUsageInformation.*
This element has the following connections to other elements:

Table 5.273 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	ServiceSuppliers	ServiceSuppliers associated with this collection of EnergyUsageInformation.	ServiceSupplier See Clause 5.7.5.2.1.23.	[0..*]
Association	UsagePoints	UsagePoints in this collection of EnergyUsageInformation.	UsagePoint See Clause 5.7.5.2.1.26.	[0..*]

5.7.5.2.1.8. IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB P4P10 EUIIdentifiedObject.*

Table 5.274 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
aliasName	The aliasName is free text human readable name of the object alternative to IdentifiedObject.name. It may be non unique and may not correlate to a naming hierarchy.	String See Clause 6.6.3.7.	[0..1]
mRID	Master resource identifier issued by a model authority. The mRID is globally unique within an exchange context. The specific type and encoding of mRID is dependent on the context in which it is found by the adopting profiles.	GlobalId See Clause 6.6.3.4.	[0..1]
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.275 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	Names	All names of this identified object.	Name See Clause 5.7.5.2.1.14.	[0..*]

5.7.5.2.1.9. IntervalBlock (Class)

Time sequence of Readings of the same ReadingType.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUIIntervalBlock*.

This element has the following connections to other elements:

Table 5.276 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	IntervalReadings	Interval reading contained in this block.	IntervalReading See Clause 5.7.5.2.1.10.	[0..*]
Association	MeterReading	Meter reading containing this interval block.	MeterReading See Clause 5.7.5.2.1.12.	[0..1]
Association	ReadingType	Type information for interval reading values contained in this block.	ReadingType See Clause 5.7.5.2.1.20.	[1]

5.7.5.2.1.10. IntervalReading (Class)

Data captured over a specific interval of time. If not specified, the duration is the intervalLength of the associated ReadingType, where the full definition of the units of measure is located.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUIIntervalReading*.

Table 5.277 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
cost	The cost associated with this reading for this interval.	Real See Clause 6.6.3.6.	[0..1]
interval	The time interval associated with the reading.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]
value	Value of this interval reading.	Real See Clause 6.6.3.6.	[0..1]

This element has the following connections to other elements:

Table 5.278 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	IntervalBlocks	All blocks containing this interval reading.	IntervalBlock See Clause 5.7.5.2.1.9.	[0..*]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	MeterReading	MeterReading associated with these IntervalReadings.	MeterReading See Clause 5.7.5.2.1.12.	[0..1]
Association	ReadingQualities	Used only if quality of this interval reading value is different than 'Good'.	ReadingQuality See Clause 5.7.5.2.1.19.	[0..*]

5.7.5.2.1.11. LineItem (Class)

Details on an amount line, with rounding, date and note.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\LineItem.*

Table 5.279 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
amount	Amount for this line item.	Real See Clause 6.6.3.6.	[0..1]
dateTime	Date and time when this line was created in the application process.	UTCDateTime See Clause 6.6.1.11.	[0..1]
measurement	Optional relevant measurement for line item.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
note	Free format note relevant to this line.	String See Clause 6.6.3.7.	[0..1]
rounding	Totalized monetary value of all errors due to process rounding or truncating that is not reflected in 'amount'.	Real See Clause 6.6.3.6.	[0..1]

5.7.5.2.1.12. MeterReading (Class)

Set of values obtained from the meter.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\MeterReading.*

Table 5.280 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]
valuesInterval	Date and time interval of the data items contained within this meter reading.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]

This element has the following connections to other elements:

Table 5.281 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	IntervalBlocks	All interval blocks contained in this meter reading.	IntervalBlock See Clause 5.7.5.2.1.9.	[0..*]
Association	IntervalReadings	IntervalReadings of this MeterReading.	IntervalReading See Clause 5.7.5.2.1.10.	[0..*]
Association	Readings	All reading values contained within this meter reading.	Reading See Clause 5.7.5.2.1.18.	[0..*]
Association	ReadingType	ReadingType of this MeterReading.	ReadingType See Clause 5.7.5.2.1.20.	[0..1]
Association	UsagePoint	UsagePoint for this MeterReading.	UsagePoint See Clause 5.7.5.2.1.26.	[1]

5.7.5.2.1.13. NAESEB_EUI_Version (Class)

The version of this model.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB_PAPI0 EUI\NAESB_EUI_Version.*

Table 5.282 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
date	Form is YYYY-MM-DD for example for January 5, 2009 it is 2009-01-05. 20120802	UTCDateTime See Clause 6.6.1.11.	[1]
version	naesb_eui_vxx.yy where xx is the major version and yy is the minor version.	String See Clause 6.6.3.7.	[1]

5.7.5.2.1.14. Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB_PAPI0 EUI\Name.*

Table 5.283 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	Any free text that name the object.	String See Clause 6.6.3.7.	[1]

This element has the following connections to other elements:

Table 5.284 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	IdentifiedObject	Identified object that this name designates.	IdentifiedObject See Clause 5.7.5.2.1.8.	[1]
Association	NameType	Type of this name.	NameType See Clause 5.7.5.2.1.15.	[1]

5.7.5.2.1.15. NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EU\NameType.*

Table 5.285 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	Name of the name type.	String See Clause 6.6.3.7.	[1]

This element has the following connections to other elements:

Table 5.286 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	Names	All names of this type.	Name See Clause 5.7.5.2.1.14.	[0..*]
Association	NameTypeAuthority	Authority responsible for managing names of this type.	NameTypeAuthority See Clause 5.7.5.2.1.16.	[0..1]

5.7.5.2.1.16. NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EU\NameTypeAuthority.*

Table 5.287 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	Name of the name type authority.	String See Clause 6.6.3.7.	[1]

This element has the following connections to other elements:

Table 5.288 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	NameTypes	All name types managed by this authority.	NameType See Clause 5.7.5.2.1.15.	[0..*]

5.7.5.2.1.17. PositionPoint (Class)

Set of spatial coordinates that determine a point, defined in coordinate system specified in 'Location.CoordinateSystem', or "WGS 84" if not specified otherwise. Use a single position point instance to describe a point-oriented location. Use a sequence of position points to describe a line-oriented object (physical location of non-point oriented objects like cables or lines), or area of an object (like a substation or a geographical zone - in this case, have first and last position point with the same values).

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eni_model_20101111Update_20120802NAESB PAPI0 EU\PositionPoint.*

Table 5.289 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
xPosition	X axis position.	String See Clause 6.6.3.7.	[0..1]
yPosition	Y axis position.	String See Clause 6.6.3.7.	[0..1]
zPosition	(if applicable) Z axis position.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.290 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	UsagePoint	UsagePoint at this PositionPoint (location).	UsagePoint See Clause 5.7.5.2.1.26.	[0..1]

5.7.5.2.1.18. Reading (Class)

Specific value measured by a meter or other asset. Each Reading is associated with a specific ReadingType.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\Reading.*

Table 5.291 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
cost	Cost in a currency	Real See Clause 6.6.3.6.	[0..1]
timeStamp	The time when the value was last updated	UTCDateTime See Clause 6.6.1.11.	[0..1]
value	Value of this reading.	Real See Clause 6.6.3.6.	[0..1]

This element has the following connections to other elements:

Table 5.292 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	MeterReadings	All meter readings (sets of values) containing this reading value.	MeterReading See Clause 5.7.5.2.1.12.	[0..*]
Association	ReadingQualities	Used only if quality of this reading value is different than 'Good'.	ReadingQuality See Clause 5.7.5.2.1.19.	[0..*]
Association	ReadingType	Type information for this reading value.	ReadingType See Clause 5.7.5.2.1.20.	[1]

5.7.5.2.1.19. ReadingQuality (Class)

Quality of a specific reading value or interval reading value. Note that more than one quality may be applicable to a given reading. Typically not used unless problems or unusual conditions occur (i.e., quality for each reading is assumed to be 'good' unless stated otherwise in associated reading quality).

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\ReadingQuality.*

Table 5.293 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
quality	Quality, to be specified if different than ReadingType.defaultQuality.	QualityOfReading See Clause 6.6.2.5.	[0..1]

This element has the following connections to other elements:

Table 5.294 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	IntervalReading	Interval reading value to which this quality applies.	IntervalReading See Clause 5.7.5.2.1.10.	[1]
Association	Reading	Reading value to which this quality applies.	Reading See Clause 5.7.5.2.1.18.	[0..1]

5.7.5.2.1.20. ReadingType (Class)

Detailed description for a type of a reading value. Values in attributes allow for the description of metadata about a measurement or measurements.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111\Update_20120802\NAESB PAPI0 EUI\ReadingType.*

Table 5.295 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
accumulation	Accumulation behavior of a reading over time, usually 'measuringPeriod', to be used with individual endpoints (as opposed to 'macroPeriod' and 'aggregate' that are used to describe aggregations of data from individual endpoints).	AccumulationKind See Clause 5.7.5.2.1.35.	[0..1]
aggregate	Salient attribute of the reading data aggregated from individual endpoints. This is mainly used to define a mathematical operation carried out over 'macroPeriod', but may also be used to describe an attribute of the data when the 'macroPeriod' is not defined.	DataQualifierKind See Clause 5.7.5.2.1.37.	[0..1]
argument	Argument used to introduce numbers into the unit of measure description where they are needed (e.g., 4 where the measure needs an argument such as CEMI(n=4)). Most arguments used in practice however will be integers (i.e., 'denominator'=1). Value 0 in 'numerator' and 'denominator' means not applicable.	RationalNumber See Clause 5.7.5.2.1.30.	[0..1]
commodity	Commodity being measured.	CommodityKind See Clause 5.7.5.2.1.36.	[0..1]
consumptionTier	In case of common flat-rate pricing for power, in which all purchases are at a given rate, 'consumptionTier'=0. Otherwise, the value indicates the consumption tier, which can be used in conjunction with time of use or critical peak period pricing. Consumption tier pricing refers to the method of billing in which a certain "block" of energy is purchased/sold at one price, after which the next block of energy is purchased at another price, and so on, all throughout a defined period. At the start of the defined period, consumption is initially zero, and any usage is measured against the first consumption tier ('consumptionTier'=1). If this block of energy is consumed before the end of the period, energy consumption moves to be reconed against the second consumption tier ('consumptionTier'=2), and so on. At the end of the defined period, the consumption accumulator is reset, and usage within the 'consumptionTier'=1 restarts.	Integer See Clause 6.6.3.5.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
epp	Critical peak period bucket the reading value is attributed to. Value 0 means not applicable. Even though critical peak period is usually considered a specialized form of time of use, this attribute is defined explicitly for flexibility.	Integer See Clause 6.6.3.5.	[0..1]
currency	Metering-specific currency.	CurrencyKind See Clause 6.6.2.1.	[0..1]
defaultQuality	The default quality of readings. It may be overridden for specific measurements in Reading or IntervalReading classes.	QualityOfReading See Clause 6.6.2.5.	[0..1]
flowDirection	Flow direction for a reading where the direction of flow of the commodity is important (for electricity measurements this includes current, energy, power, and demand).	FlowDirectionKind See Clause 5.7.5.2.1.31.	[0..1]
interharmonic	Indication of a "harmonic" or "interharmonic" basis for the measurement. Value 0 in 'numerator' and 'denominator' means not applicable.	ReadingInterharmonic See Clause 5.7.5.2.1.29.	[0..1]
intervalLength	(if incremental reading value) Length of increment interval.	Duration See Clause 6.6.1.9.	[0..1]
macroPeriod	Time period of interest that reflects how the reading is viewed or captured over a long period of time.	MacroPeriodKind See Clause 5.7.5.2.1.38.	[0..1]
measurementKind	Identifies 'what' is being measured, as refinement of 'commodity'. When combined with 'unit', it provides detail to the unit of measure. For example, 'energy' with a unit of measure of 'kWh' indicates to the user that active energy is being measured, while with 'kVAh' or 'kVArh', it indicates apparent energy and reactive energy, respectively. 'power' can be combined in a similar way with various power units of measure: Distortion power ('distortionVoltAmperes') with 'kVA' is different from 'power' with 'kVA'.	MeasurementKind See Clause 5.7.5.2.1.32.	[0..1]
measuringPeriod	Time attribute inherent or fundamental to the reading value (as opposed to 'macroPeriod' that supplies an "adjective" to describe aspects of a time period with regard to the measurement). It refers to the way the value was originally measured and not to the frequency at which it is reported or presented. For example, an hourly interval of consumption data would have value 'hourly' as an attribute. However in the case of an hourly sampled voltage value, the meterReadings schema would carry the 'hourly' interval size information. It is common for meters to report demand in a form that is measured over the course of a portion of an hour, while enterprise applications however commonly assume the demand (in kW or kVAr) normalised to 1 hour. The system that receives readings directly from the meter therefore must perform this transformation before publishing readings for use by the other enterprise systems. The scalar used is chosen based on the block size (not any sub-interval size).	TimeAttributeKind See Clause 5.7.5.2.1.41.	[0..1]
multiplier	Metering-specific multiplier.	SiScaleCodeType See Clause 6.6.2.3.	[0..1]
phases	Metric-specific phase code.	PhaseCodeKind See Clause 5.7.5.2.1.40.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
tou	Time of use bucket the reading value is attributed to. Value 0 means not applicable.	Integer See Clause 6.6.3.5.	[0..1]
unit	Metering-specific unit.	UnitSymbolKind See Clause 6.6.2.6.	[0..1]

This element has the following connections to other elements:

Table 5.296 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	IntervalBlocks	All blocks containing interval reading values with this type information.	IntervalBlock See Clause 5.7.5.2.1.9.	[0..*]
Association	MeterReadings	MeterReadings with this ReadingType.	MeterReading See Clause 5.7.5.2.1.12.	[0..*]
Association	Readings	All reading values with this type information.	Reading See Clause 5.7.5.2.1.18.	[0..*]
Association	ServiceSupplier	ServiceSupplier able to use ReadingType.	ServiceSupplier See Clause 5.7.5.2.1.23.	[0..1]

5.7.5.2.1.21. ServiceCategory (Class)

Category of service provided to the customer.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EU\ServiceCategory.*

Table 5.297 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
kind	Kind of service.	ServiceKind See Clause 5.7.5.2.1.33.	[0..1]

This element has the following connections to other elements:

Table 5.298 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	UsagePoint	All service delivery points that deliver this category of service.	UsagePoint See Clause 5.7.5.2.1.26.	[0..*]

5.7.5.2.1.22. ServiceDeliveryPoint (Class)

Logical point on the network where the ownership of the service changes hands. It is one of potentially many service points within a service location, delivering service in accordance with a customer agreement. Used at the place where a meter may be installed.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111\Update_20120802\NAESB PAPI0 EUI\ServiceDeliveryPoint.*

Table 5.299 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.300 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	CustomerAgreement	Customer agreement regulating this service delivery point.	CustomerAgreement See Clause 5.7.5.2.1.3.	[0..1]
Association	TariffProfile	TariffProfile governing this ServiceDeliveryPoint.	TariffProfile See Clause 5.7.5.2.1.25.	[0..1]
Association	UsagePoints	UsagePoints at this ServiceDeliveryPoint	UsagePoint See Clause 5.7.5.2.1.26.	[0..*]

5.7.5.2.1.23. ServiceSupplier (Class)

Organization that provides services to customers.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111\Update_20120802\NAESB PAPI0 EUI\ServiceSupplier.*

Table 5.301 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
kind	Kind of supplier.	SupplierKind See Clause 5.7.5.2.1.34.	[0..1]
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.302 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	Customers	Customers supported by a ServiceSupplier.	Customer See Clause 5.7.5.2.1.2.	[0..*]
Association	EnergyUsageInformation	Collection of EnergyUsageInformation supported by this ServiceSupplier.	EnergyUsageInformation See Clause 5.7.5.2.1.7.	[0..*]
Association	ReadingTypes	ReadingTypes supported by ServiceSupplier.	ReadingType See Clause 5.7.5.2.1.20.	[0..*]

5.7.5.2.1.24. SummaryMeasurement (Class)

An aggregated summary measurement reading.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\SummaryMeasurement.*

Table 5.303 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
powerOfTenMultiplier	The multiplier part of the unit of measure, e.g., "kilo" (k)	SiScaleCodeType See Clause 6.6.2.3.	[0..1]
readingTypeRef	Reference to a full ReadingType that describes the measurement. Note only one of readingTypeRef or unit + powerOfTenMultiplier are used in any single instance of SummaryMeasurement.	AnyURI See Clause 6.6.3.2.	[0..1]
timestamp	The date and time (if needed) of the summary measurement.	UTCDateTime See Clause 6.6.1.11.	[0..1]
unit	The units of the reading, e.g., "Wh".	UnitSymbolKind See Clause 6.6.2.6.	[0..1]
value	The value of the summary measurement.	Real See Clause 6.6.3.6.	[0..1]

5.7.5.2.1.25. TariffProfile (Class)

A schedule of charges; structure associated with Tariff that allows the definition of complex tariff structures such as step and time of use.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\TariffProfile.*

Table 5.304 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
name	The name is any free human readable and possibly non unique text naming the object.	String See Clause 6.6.3.7.	[0..1]

This element has the following connections to other elements:

Table 5.305 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	ServiceDeliveryPoints	ServiceDeliveryPoints governed by a TariffProfile.	ServiceDeliveryPoint See Clause 5.7.5.2.1.22.	[0..*]

5.7.5.2.1.26. UsagePoint (Class)

Logical point on a network at which consumption or production is either physically measured (e.g., metered) or estimated (e.g., unmetered street lights).

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUIUsagePoint.*

Table 5.306 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
description	A human readable description of the object.	String See Clause 6.6.3.7.	[0..1]
isVirtual	Is used to indicate that the UsagePoint is virtual (that is, not a real physical measurement). This may be the result of a computation or estimation.	Boolean See Clause 6.6.3.3.	[1]
name	The name is any free human readable and possibly non-unique text naming the object.	String See Clause 6.6.3.7.	[0..1]
roleFlags	The set of roles pertinent to this UsagePoint	RoleFlags See Clause 5.7.5.2.1.28.	[1]
status	Status of this UsagePoint: 0 - Off 1 - On	Integer See Clause 6.6.3.5.	[1]

This element has the following connections to other elements:

Table 5.307 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	Customer	Customer associated with this UsagePoint	Customer See Clause 5.7.5.2.1.2.	[0..1]
Association	ElectricPowerQualitySummaries	ElectricPowerQualitySummaries at this UsagePoint.	ElectricPowerQualitySummary See Clause 5.7.5.2.1.5.	[0..*]
Association	EndDeviceAssets	All end device assets at this usage point.	EndDeviceAsset See Clause 5.7.5.2.1.6.	[0..*]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	EnergyUsageInformation	EnergyUsageInformation collection that this UsagePoint is part of.	EnergyUsageInformation	[0..*]
Association	MeterReadings	MeterReadings available at this UsagePoint.	MeterReading See Clause 5.7.5.2.1.7.	[0..*]
Association	PositionPoint	PositionPoint describing location of this UsagePoint.	PositionPoint See Clause 5.7.5.2.1.12.	[0..1]
Association	ServiceCategory	Service category delivered by this service delivery point.	ServiceCategory See Clause 5.7.5.2.1.17.	[0..1]
Association	ServiceDeliveryPoints	ServiceDeliveryPoints associated with this UsagePoint.	ServiceDeliveryPoint See Clause 5.7.5.2.1.21.	[0..*]
Association	UsageSummaries	UsageSummaries at this UsagePoint.	UsageSummary See Clause 5.7.5.2.1.22.	[0..*]
			UsageSummary See Clause 5.7.5.2.1.27.	[0..*]

5.7.5.2.1.27. UsageSummary (Class)

Summary of usage for a billing period.

Note: This model was extended using the work of OpenADE 2015 for ESPI. These extensions are proposed for the next revision of the NAESB standard REQ.21. Also extended are SummaryMeasurement and LineDetail.

Parent Class: IdentifiedObject (See Clause 5.7.5.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\UsageSummary.*

Table 5.308 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
billingPeriod	The billing period to which the included measurements apply. May also be an off-bill arbitrary period.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]
billLastPeriod	The amount of the bill for the referenced billingPeriod in the currency attribute of this UsageSummary (e.g., 840 = USD, US dollar).	Real See Clause 6.6.3.6.	[0..1]
billToDate	If the summary contains data from a current period beyond the end of the referenced billingPeriod, the bill amount as of the date the summary is generated, in the currency attribute of this UsageSummary (e.g., 840 = USD, US dollar).	Real See Clause 6.6.3.6.	[0..1]
commodity	The commodity for this summary report.	CommodityKind See Clause 5.7.5.2.1.36.	[0..1]
costAdditionalLastPeriod	Additional charges from the for the referenced billingPeriod, in the currency attribute of this UsageSummary (e.g., 840 = USD, US dollar).	Real See Clause 6.6.3.6.	[0..*]

Attribute Name	Description	Attribute Type	Multiplicity
costAdditionalLastPeriodDetail	Additional charges from the for the referenced billingPeriod which in total add up to costAdditionalLastPeriod, in the currency attribute of this UsageSummary (e.g., 840 = USD, US dollar).	LineItem See Clause 5.7.5.2.1.11.	[0..*]
currency	The ISO 4217 code indicating the currency applicable to the bill amounts in the summary. See list at http://www.unece.org/cefact/recommendations/rec09/rec09_ecetrd203.pdf	CurrencyKind See Clause 6.6.2.1.	[0..1]
currentBillingPeriodOverallConsumption	The total consumption for the billing period.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
currentDayLastYearNetConsumption	The amount of energy consumed one year ago interpreted as same day of week same week of year (see ISO 8601).	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
currentDayNetConsumption	Net consumption for the current day (delivered - received).	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
currentDayOverallConsumption	Overall energy consumption for the current day.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
overallConsumptionLastPeriod	The amount of energy consumed in for the referenced billingPeriod.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
peakDemand	If the summary contains data from a current period beyond the end of the referenced billingPeriod, peak demand recorded for the current period.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
previousDayLastYearOverallConsumption	If the summary contains data from a current period beyond the end of the referenced billingPeriod, the amount of energy consumed on the previous day one year ago interpreted as same day of week same week of year (see ISO 8601) based on the day of the statusTimestamp.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
previousDayNetConsumption	If the summary contains data from a current period beyond the end of the referenced billingPeriod, net consumption for the previous day relative to the day of the statusTimestamp.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
previousDayOverallConsumption	If the summary contains data from a current period beyond the end of the referenced billingPeriod, the total consumption for the previous day based on the day of the statusTimestamp.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
qualityOfReading	Indication of the quality of the summary readings	QualityOfReading See Clause 6.6.2.5.	[0..1]
ratchetDemand	If the summary contains data from a current period beyond the end of the referenced billingPeriod, the current ratchet demand value for the ratchet demand over the ratchetDemandPeriod.	SummaryMeasurement See Clause 5.7.5.2.1.24.	[0..1]
ratchetDemandPeriod	The period over which the ratchet demand applies.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]
statusTimeStamp	Date/Time status of this UsageSummary.	UTCDateTime See Clause 6.6.1.11.	[1]

This element has the following connections to other elements:

Table 5.309 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	UsagePoint	UsagePoint for this UsageSummary.	UsagePoint See Clause 5.7.5.2.1.26.	[0..1]

5.7.5.2.1.28. RoleFlags (Class)

Describes a set of specific communication or physical attributes that the associated UsagePoint has. The term RoleFlags in this specific case pertains to the application-specific nature of the UsagePoint. More than one of these flags can be true for any UsagePoint.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EU\RoleFlags*.

Table 5.310 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
isDC	Is direct current (DC) rather than alternating current (AC)	Boolean See Clause 6.6.3.3.	[1]
isDER	Is a DER point	Boolean See Clause 6.6.3.3.	[1]
isMirror	Represents a copy or mirror of an original UsagePoint	Boolean See Clause 6.6.3.3.	[1]
isOneway	Device is a one-way communications device that can only transmit its information and cannot receive.	Boolean See Clause 6.6.3.3.	[1]
isPEV	Is a PEV Usage Point	Boolean See Clause 6.6.3.3.	[1]
isPremiseAggregationPoint	This UsagePoint aggregates the contributions of other UsagePoints in the premise	Boolean See Clause 6.6.3.3.	[1]
isRevenueQuality	Is revenue quality.	Boolean See Clause 6.6.3.3.	[1]
isVirtual	Is used to indicate that the UsagePoint is virtual (that is, not a real physical measurement). This may be the result of a computation or estimation. Note that this was added to IEC 61968-9 2nd ed model of a UsagePoint which occurred after its adoption by NAESB. It has since been added to Green Button and will be added to the next release of "NAESB Business Practices and Information Models to Support Priority Action Plan 10 – Standardized Energy Usage Information standards, Revision 1.1, 2012.	Boolean See Clause 6.6.3.3.	[1]

5.7.5.2.1.29. ReadingInterharmonic «Compound» (Class)

Interharmonics are represented as a rational number 'numerator' / 'denominator', and harmonics are represented using the same mechanism and identified by 'denominator'=1.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EU\ReadingInterharmonic*.

Table 5.311 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
numerator	Interharmonic numerator. Value 0 means not applicable. Value 1 is used in combination with 'denominator'=2 to represent interharmonic 1/2, and with 'denominator'=1 it represents fundamental frequency. Finally, values greater than 1 indicate the harmonic of that order (e.g., 'numerator'=5 is the fifth harmonic).	Integer See Clause 6.6.3.5.	[0..1]
denominator	Interharmonic denominator. Value 0 means not applicable. Value 2 is used in combination with 'numerator'=1 to represent interharmonic 1/2. Finally, value 1 indicates the harmonic of the order specified with 'numerator'.	Integer See Clause 6.6.3.5.	[0..1]

5.7.5.2.1.30. RationalNumber «Compound» (Class)

Rational number = 'numerator' / 'denominator'.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\RationalNumber*.

Table 5.312 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
numerator	Numerator.	Integer See Clause 6.6.3.5.	[0..1]
denominator	Denominator. Value 1 indicates the number is a simple integer.	Integer See Clause 6.6.3.5.	[0..1]

5.7.5.2.1.31. FlowDirectionKind (Enumeration)

This value identifies the direction of flow of what is being measured. Specific indications are supported, especially for electrical measurements that have complex values.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\FlowDirectionKind*.

Table 5.313 – FlowDirectionKind Enumerated Values

Enumerated Value	Definition
none	Not Applicable (N/A)
forward	"Delivered," or "Imported" as defined 61968-2. Forward Active Energy is a positive kWh value as one would naturally expect to find as energy is supplied by the utility and consumed at the service. Forward Reactive Energy is a positive VARh value as one would naturally expect to find in the presence of inductive loading.

Enumerated Value	Definition
	In polyphase metering, the forward energy register is incremented when the sum of the phase energies is greater than zero:
lagging	$Energy_A + Energy_B + Energy_C > 0$ Typically used to describe that a power factor is lagging the reference value. Note 1: When used to describe VA, "lagging" describes a form of measurement where reactive power is considered in all four quadrants, but real power is considered only in quadrants I and IV. Note 2: When used to describe power factor, the term "lagging" implies that the PF is negative. The term "lagging" in this case takes the place of the negative sign. If a signed PF value is to be passed by the data producer, then the direction of flow enumeration zero (none) should be used in order to avoid the possibility of creating an expression that employs a double negative. The data consumer should be able to tell from the sign of the data if the PF is leading or lagging. This principle is analogous to the concept that "Reverse" energy is an implied negative value, and to publish a negative reverse value would be ambiguous. Note 3: Lagging power factors typically indicate inductive loading.
leading	Typically used to describe that a power factor is leading the reference value. Note: Leading power factors typically indicate capacitive loading.
net	$ Forward - Reverse $, See 61968-2.
	Note: In some systems, the value passed as a "net" value could become negative. In other systems the value passed as a "net" value is always a positive number, and rolls-over and rolls-under as needed.
q1plusQ2	Reactive positive quadrants. (The term "lagging" is preferred.)
q1plusQ3	Quadrants 1 and 3
q1plusQ4	Quadrants 1 and 4 usually represent forward active energy
q1minusQ4	Quadrant1 minus Quadrant4
q2plusQ3	Quadrants 2 and 3 usually represent reverse active energy
q2plusQ4	Quadrants 2 and 4
q2minusQ3	Quadrant2 minus Quadrant3
q3plusQ4	Reactive negative quadrants. (The term "leading" is preferred.)
q3minusQ2	Quadrant3 minus Quadrant2
quadrant1	Quadrant1 only
quadrant2	Quadrant2 only
quadrant3	Quadrant3 only
quadrant4	Quadrant4 only
reverse	Reverse Active Energy is equivalent to "Received," or "Exported" as defined in 61968-2.
	Reverse Active Energy is a positive kWh value as one would expect to find when energy is backfed by the service onto the utility network.
	Reverse Reactive Energy is a positive VARh value as one would expect to find in the presence of capacitive loading and a leading Power Factor.

Enumerated Value	Definition
	In polyphase metering, the reverse energy register is incremented when the sum of the phase energies is less than zero: $Energy_A + Energy_B + Energy_C < 0$ Note: The value passed as a reverse value is always a positive value. It is understood by the label "reverse" that it represents negative flow.
total	$ Forward + Reverse $. See 61968-2. The sum of the commodity in all quadrants Q1+Q2+Q3+Q4. In polyphase metering, the total energy register is incremented when the absolute value of the sum of the phase energies is greater than zero: $ Energy_A + Energy_B + Energy_C > 0$
totalByPhase	In polyphase metering, the total by phase energy register is incremented when the sum of the absolute values of the phase energies is greater than zero: $ Energy_A + Energy_B + Energy_C > 0$ In single phase metering, the formulas for "Total" and "Total by phase" collapse to the same expression. For communication purposes however, the "Total" enumeration should be used with single phase meter data.

5.7.5.2.1.32. MeasurementKind (Enumeration)

Identifies 'what' is being measured, as refinement of 'commodity'. When combined with 'unit', it provides detail to the unit of measure. For example, 'energy' with a unit of measure of 'kWh' indicates to the user that active energy is being measured, while with 'kVAh' or 'kVAh', it indicates apparent energy and reactive energy, respectively. 'power' can be combined in a similar way with various power units of measure: Distortion power ('distortion VoltAmperes') with 'kVA' is different from 'power' with 'kVA'.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUIMeasurementKind*.

Table 5.314 - MeasurementKind Enumerated Values

Enumerated Value	Definition
none	Not Applicable
apparentPowerFactor	apparentPowerFactor
currency	Funds
current	Current
currentAngle	Current angle
currentImbalance	Current imbalance
date	Date

Enumerated Value	Definition
demand	Demand
distance	Distance
distortionVoltAmperes	Distortion volt-amperes
energization	Energization
energy	Energy
energizationLoadSide	Energization load side
fan	Fan
frequency	Frequency
ieeee1366ASAI	IEEEE1366 Average Service Availability Index (ASAI)
ieeee1366ASIDI	IEEEE1366 Average System Interruption Duration Index (ASIDI)
ieeee1366ASIFI	IEEEE1366 Average System Interruption Frequency Index (ASIFI)
ieeee1366CAIDI	IEEEE1366 Customer Average Interruption Duration Index (CAIDI)
ieeee1366CAIFI	IEEEE1366 Customer Average Interruption Frequency Index (CAIFI)
ieeee1366CEMIn	IEEEE1366 Customers Experiencing Multiple Interruptions (CEMIN)
ieeee1366CEMSMIn	IEEEE 1366 Customers Experiencing Multiple Sustained Interruption and Momentary Interruption Events (CEMSMIN)
ieeee1366CTAIDI	IEEEE 1366 Customers Experiencing Multiple Sustained Interruption and Momentary Interruption Events (CEMSMIN)
ieeee1366MAIFI	IEEEE 1366 Momentary Average Interruption Frequency Index (MAIFI)
ieeee1366MAIFle	IEEEE 1366 Momentary Average Interruption Event Frequency Index (MAIFIE)
ieeee1366SAIDI	IEEEE 1366 System average interruption duration index (SAIDI)
ieeee1366SAIFI	IEEEE 1366 System average interruption frequency index (SAIFI)
lineLosses	Line Losses
losses	Losses
negativeSequence	Negative Sequence
phasorPowerFactor	Phasor Power Factor
phasorReactivePower	Phasor Reactive Power
positiveSequence	Positive Sequence
power	Power
powerFactor	Power Factor
quantityPower	Quantity Power
sag	Sag
swell	Swell
switchPosition	Switch Position
tapPosition	Tap Position
tariffRate	Tariff Rate
temperature	Temperature
totalHarmonicDistortion	Total Harmonic Distortion (THD)
transformerLosses	Transformer Losses
unipedeVoltageDip10to15	Unipede Voltage Dip 10 to 15

Enumerated Value	Definition
unipedeVoltageDip15to30	Unipede Voltage Dip 15 to 30
unipedeVoltageDip30to60	Unipede Voltage Dip 30 to 60
unipedeVoltageDip60to90	Unipede Voltage Dip 60 to 90
unipedeVoltageDip90to100	Unipede Voltage Dip 90 to 100
voltage	Voltage
voltageAngle	Voltage Angle
voltageExcursion	Voltage Excursion
voltageImbalance	Voltage Imbalance
volume	Indicates fluid volume
zeroFlowDuration	Zero Flow Duration
zeroSequence	Zero Sequence
distortionPowerFactor	Distortion power factor
frequencyExcursion	Frequency excursion. Usually expressed as a "count".
applicationContext	Application context.
apTitle	Ap Title
assetNumber	Asset number.
bandwidth	Bandwidth
batteryVoltage	Battery voltage
broadcastAddress	Broadcast address
deviceAddressType1	Device address type 1
deviceAddressType2	Device address type 2
deviceAddressType3	Device address type 3
deviceAddressType4	Device address type 4
deviceClass	Device class
electronicSerialNumber	Electronic serial number
endDeviceID	End device ID
groupAddressType1	Group address type 1
groupAddressType2	Group address type 2
groupAddressType3	Group address type 3
groupAddressType4	Group address type 4
ipAddress	IP Address
macAddress	MAC Address
mfgAssignedConfigurationID	Manufacturer Assigned Configuration ID
mfgAssignedPhysicalSerialNumber	Manufacturer Assigned Physical Serial Number
mfgAssignedProductNumber	Manufacturer Assigned Product Number
mfgAssignedUniqueCommunicationAddress	Manufacturer Assigned Unique Communication Address

Enumerated Value	Definition
multiCastAddress	Multicast Address
oneWayAddress	One Way Address
signalStrength	Signal Strength
twoWayAddress	Two Way Address
signaltoNoiseRatio	Signal to Noise Ratio
alarm	Alarm
batteryCarryover	Battery carryover
dataOverflowAlarm	Data overflow alarm
demandLimit	Demand limit
demandReset	Demand resets. Usually expressed as a count as part of a billing cycle.
diagnostic	Diagnostic
emergencyLimit	Emergency limit
encoderTamper	Encoder tamper
ieee1366MomentaryInterruption	IEEE 1366 Momentary Interruption
ieee1366MomentaryInterruptionEvent	IEEE 1366 Momentary Interruption Event
ieee1366SustainedInterruption	IEEE 1366 Sustained Interruption
interruptionBehaviour	Interruption Behavior
inversionTamper	Inversion Tamper
loadInterrupt	Load Interrupt
loadShed	Load Shed
maintenance	Maintenance
physicalTamper	Physical Tamper
powerLossTamper	Power Loss Tamper
powerOutage	Power Outage
powerQuality	Power Quality
powerRestoration	Power Restoration
programmed	Programmed
pushbutton	Push Button
relayActivation	Relay Activation
relayCycle	Relay Cycle
removalTamper	Removal Tamper
reprogrammingTamper	Reprogramming Tamper
reverseRotationTamper	Reverse Rotation Tamper
switchArmed	Switch Armed
switchDisabled	Switch Disabled
tamper	Tamper
watchdogTimeout	Watchdog Timeout

Enumerated Value	Definition
billLastPeriod	Customer's bill for the previous billing period (Currency)
billToDate	Customer's bill, as known thus far within the present billing period (Currency)
billCarryover	Customer's bill carryover.
connectionFee	Monthly fee for connection to commodity.
audibleVolume	Sound
volumetricFlow	Volumetric Flow

5.7.5.2.1.33. ServiceKind (Enumeration)

Kind of service.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\ServiceKind.*

Table 5.315 - ServiceKind Enumerated Values

Enumerated Value	Definition
electricity	Electricity
gas	Gas
water	Water
time	Time
heat	Heat
refuse	Refuse
sewerage	Sewerage
rates	Rates
tvLicence	TV Licence
internet	Internet
other	Other
cold	Includes chilled water and ice

5.7.5.2.1.34. SupplierKind (Enumeration)

Kind of supplier.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\SupplierKind.*

Table 5.316 - SupplierKind Enumerated Values

Enumerated Value	Definition
utility	Utility
retailer	Retailer

Enumerated Value	Definition
other	Other
district	District
intermediary	Intermediary
local	Local
microgrid	Microgrid

5.7.5.2.1.35. AccumulationKind (Enumeration)

This attribute provides metadata about how the measurements described are made. These enumerated values refer to the common present and historic methods by which physical measurements are made in revenue-quality metering instruments and are based on that terminology.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUIAccumulationKind.*

Table 5.317 - AccumulationKind Enumerated Values

Enumerated Value	Definition
none	Not Applicable, or implied by the unit of measure.
bulkQuantity	A value from a register which represents the bulk quantity of a commodity. This quantity is computed as the integral of the commodity usage rate. This value is typically used as the basis for the dial reading at the meter, and as a result, will roll over upon reaching a maximum dial value. Note 1: With the metering system, the roll-over behavior typically implies a roll-under behavior so that the value presented is always a positive value (e.g., unsigned integer or positive decimal.) However, when communicating data between enterprise applications a negative value might occur in a case such as net metering. Note 2: A BulkQuantity refers primarily to the dial reading and not the consumption over a specific period of time.
continuousCumulative	The sum of the previous billing period values and the present period value. Note: "ContinuousCumulative" is commonly used in conjunction with "demand." The "ContinuousCumulative Demand" would be the cumulative sum of the previous billing period maximum demand values (as occurring with each demand reset) summed with the present period maximum demand value (which has yet to be reset.)
cumulative	The sum of the previous billing period values. Note: "Cumulative" is commonly used in conjunction with "demand." Each demand reset causes the maximum demand value for the present billing period (since the last demand reset) to accumulate as an accumulative total of all maximum demands. So instead of "zeroing" the demand register, a demand reset has the affect of adding the present maximum demand to this accumulating total.
deltaData	The difference between the value at the end of the prescribed interval and the beginning of the interval. This is used for incremental interval data. Note: One common application would be for load profile data, another use might be to report the number of events within an interval (such as the number of equipment energizations within the specified period of time.)
indicating	As if a needle is swung out on the meter face to a value to indicate the current value. (Note: An "indicating" value is typically measured over hundreds of milliseconds or greater, or may imply a "pusher" mechanism to capture a value. Compare this to "instantaneous" which is measured over a shorter period of time.)
summation	A form of accumulation which is selective with respect to time.

Enumerated Value	Definition
timeDelay	Note : "Summation" could be considered a specialization of "Bulk Quantity" according to the rules of inheritance where "Summation" selectively accumulates pulses over a timing pattern, and "BulkQuantity" accumulates pulses all of the time.
instantaneous	A form of computation which introduces a time delay characteristic to the data value
latchingQuantity	Typically measured over the fastest period of time allowed by the definition of the metric (usually milliseconds or tens of milliseconds.) (Note: "Instantaneous" was moved to attribute #3 in 61968-9Ed2 from attribute #1 in 61968-9Ed1.) When this description is applied to a metered value, it implies that the value is a time-independent cumulative quantity much a BulkQuantity, except that it latches upon the maximum value upon reaching that value. Any additional accumulation (positive or negative) is discarded until a reset occurs. Note: A LatchingQuantity may also occur in the downward direction – upon reaching a minimum value. The terms "maximum" or "minimum" will usually be included as an attribute when this type of accumulation behaviour is present. When this description is applied to an encoded value (UOM= "Code"), it implies that the value has one or more bits which are latching. The condition that caused the bit to be set may have long since evaporated. In either case, the timestamp that accompanies the value may not coincide with the moment the value was initially set. In both cases a system will need to perform an operation to clear the latched value.
boundedQuantity	A time-independent cumulative quantity such as BulkQuantity or a LatchingQuantity, except that the accumulation stops at the maximum or minimum values. When the maximum is reached, any additional positive accumulation is discarded, but negative accumulation may be accepted (thus lowering the counter.) Likewise, when the negative bound is reached, any additional negative accumulation is discarded, but positive accumulation is accepted (thus increasing the counter.)

5.7.5.2.1.36. CommodityKind (Enumeration)

Identifies the commodity being measured.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUICommodityKind.*

Table 5.318 – CommodityKind Enumerated Values

Enumerated Value	Definition
none	Not Applicable
electricity SecondaryMetered	All types of metered quantities. This type of reading comes from the meter and represents a "secondary" metered value.
electricity PrimaryMeter	It is possible for a meter to be outfitted with an external VT and/or CT. The meter might not be aware of these devices, and the display not compensate for their presence. Ultimately, when these scalars are applied, the value that represents the service value is called the "primary metered" value. The "index" in sub-category 3 mirrors those of sub-category 0.
communication	A measurement of the communication infrastructure itself.
air	Air
insulativeGas	Insulative gas (SF ₆ is found separately below.)
insulativeOil	Insulative Oil
naturalGas	Natural Gas
propane	Propane C ₃ H ₈

Enumerated Value	Definition
potableWater	Drinkable water
steam	Water in steam form, usually used for heating.
wasteWater	Waste Water (Sewerage)
heatingFluid	This fluid is likely in liquid form. It is not necessarily water or water based. The warm fluid returns cooler than when it was sent. The heat conveyed may be metered.
coolingFluid	The cool fluid returns warmer than when it was sent. The heat conveyed may be metered.
nonpotableWater	Reclaimed water possibly used for irrigation but not sufficiently treated to be considered safe for drinking.
nox	Nitrous Oxides NO _x
so2	Sulfur Dioxide SO ₂
ch4	Methane CH ₄
co2	Carbon Dioxide CO ₂
carbon	Carbon
pfc	Perfluorocarbons PFC
sf6	Sulfurhexafluoride SF ₆
tvLicence	Television
internet	Internet service
refuse	trash
h2	Hydrogen, H ₂
c2h2	Acetylene, C ₂ H ₂
c2h4	Ethylene, C ₂ H ₄
c2h6	Ethane, C ₂ H ₆
co	Carbon monoxide, CO
o2	Oxygen, O ₂
dissolvedCombustibleGas	Dissolved Combustible Gas (A combination of combustible gasses such as H ₂ , CH ₄ , C ₂ H ₂ , C ₂ H ₄ , C ₂ H ₆ , and/or CO in some mixture.)
co2e	Carbon Dioxide CO ₂ Equivalent
lead	Lead, Pb
mercury	Mercury, Hg
ozone	Ozone, O ₃
pm10	Particulate matter whose maximum size is 10 µm.
pm25	Particulate matter whose maximum size is 2.5 µm.
sox	Sulfur Oxides, SO _x

5.7.5.2.1.37. DataQualifierKind (Enumeration)

This value qualifies the measurement as to what it represents -- for example a minimum or maximum value or a nominal or nameplate value.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\DataQualifierKind.*

Table 5.319 – DataQualifierKind Enumerated Values

Enumerated Value	Definition
high	Typically used to identify the high volume flow port of a compound water meter.
low	Typically used to identify the low volume flow port of a compound water meter.
none	Not Applicable
average	The value represents an average
excess	The value represents an amount over which a threshold was exceeded.
highThreshold	The value represents a programmed threshold.
lowThreshold	The value represents a programmed threshold.
maximum	The highest value observed
minimum	The smallest value observed
nominal	The value represents nominal or nameplate values
normal	The value represents typical operating values
secondMaximum	The second highest value observed
secondMinimum	The second smallest value observed
thirdMaximum	The third highest value observed
fourthMaximum	The fourth highest value observed
fifthMaximum	The fifth highest value observed
sum	The accumulated sum

5.7.5.2.1.38. MacroPeriodKind (Enumeration)

Time period of interest that reflects how the reading is viewed or captured over a long period of time.

UML Element Location: *Model\UML Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\MacroPeriodKind.*

Table 5.320 – MacroPeriodKind Enumerated Values

Enumerated Value	Definition
none	Not Applicable
billingPeriod	Captured during the billing period starting at midnight of the first day of the billing period (as defined by the billing cycle day). If during the current billing period, it specifies a period from the start of the current billing period until "now".
daily	Daily Period starting at midnight. If for the current day, this specifies the time from midnight to "now".
monthly	Monthly period starting at midnight on the first day of the month. If within the current month, this specifies the period from the start of the month until "now".
seasonal	A season of time spanning multiple months. E.g., "Summer," "Spring," "Fall," and "Winter" based cycle. If within the current season, it specifies the period from the start of the current season until "now."
weekly	Weekly period starting at midnight on the first day of the week and ending the instant before midnight the last day of the week. If within the current week, it specifies the period from the start of the week until "now."
specifiedPeriod	The period defined by the start time and end time of the MeterReading.valuesInterval associated with the containing ReadingType.

5.7.5.2.1.39. MeasuringPeriodKind (Enumeration)

Time attribute inherent or fundamental to the reading value (as opposed to 'macroPeriod' that supplies an "adjective" to describe aspects of a time period with regard to the measurement). It refers to the way the value was originally measured and not to the frequency at which it is reported or presented. For example, an hourly interval of consumption data would have value 'hourly' as an attribute. However in the case of an hourly sampled voltage value, the meterReadings schema would carry the 'hourly' interval size information.

It is common for meters to report demand in a form that is measured over the course of an hour, while enterprise applications however commonly assume the demand (in kW or kVAr) normalized to 1 hour. The system that receives readings directly from the meter therefore performs this transformation before publishing readings for use by the other enterprise systems. The scalar used is chosen based on the block size (not any sub-interval size).

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\MeasuringPeriodKind.*

Table 5.321 - MeasuringPeriodKind Enumerated Values

Enumerated Value	Definition
none	Not Applicable
tenMinute	10-minute
fifteenMinute	15-minute
oneMinute	1-minute
twentyfourHour	24-hour
thirtyMinute	30-minute
fiveMinute	5-minute
sixtyMinute	60-minute
twoMinute	2-minute
threeMinute	3-minute
present	Within the present period of time
previous	Shifted within the previous monthly cycle and data set
twentyMinute	20-minute interval
fixedBlock60Min	60-minute Fixed Block
fixedBlock30Min	30-minute Fixed Block
fixedBlock20Min	20-minute Fixed Block
fixedBlock15Min	15-minute Fixed Block
fixedBlock10Min	10-minute Fixed Block
fixedBlock5Min	5-minute Fixed Block
fixedBlock1Min	1-minute Fixed Block
rollingBlock60MinIntvl30MinS	60-minute Rolling Block with 30-minute sub-intervals
rollingBlock60MinIntvl20MinS	60-minute Rolling Block with 20-minute sub-intervals
rollingBlock60MinIntvl15MinS	60-minute Rolling Block with 15-minute sub-intervals
rollingBlock60MinIntvl12MinS	60-minute Rolling Block with 12-minute sub-intervals

Enumerated Value	Definition
ubIntvl	
rollingBlock60MinIntvl10MinS	60-minute Rolling Block with 10-minute sub-intervals
ubIntvl	
rollingBlock60MinIntvl6MinSu	60-minute Rolling Block with 6-minute sub-intervals
bIntvl	
rollingBlock60MinIntvl5MinSu	60-minute Rolling Block with 5-minute sub-intervals
bIntvl	
rollingBlock60MinIntvl4MinSu	60-minute Rolling Block with 4-minute sub-intervals
bIntvl	
rollingBlock30MinIntvl15MinS	30-minute Rolling Block with 15-minute sub-intervals
ubIntvl	
rollingBlock30MinIntvl10MinS	30-minute Rolling Block with 10-minute sub-intervals
ubIntvl	
rollingBlock30MinIntvl6MinSu	30-minute Rolling Block with 6-minute sub-intervals
bIntvl	
rollingBlock30MinIntvl5MinSu	30-minute Rolling Block with 5-minute sub-intervals
bIntvl	
rollingBlock30MinIntvl3MinSu	30-minute Rolling Block with 3-minute sub-intervals
bIntvl	
rollingBlock30MinIntvl2MinSu	30-minute Rolling Block with 2-minute sub-intervals
bIntvl	
rollingBlock15MinIntvl5MinSu	15-minute Rolling Block with 5-minute sub-intervals
bIntvl	
rollingBlock15MinIntvl3MinSu	15-minute Rolling Block with 3-minute sub-intervals
bIntvl	
rollingBlock15MinIntvl1MinSu	15-minute Rolling Block with 1-minute sub-intervals
bIntvl	
rollingBlock10MinIntvl5MinSu	10-minute Rolling Block with 5-minute sub-intervals
bIntvl	
rollingBlock10MinIntvl2MinSu	10-minute Rolling Block with 2-minute sub-intervals
bIntvl	
rollingBlock10MinIntvl1MinSu	10-minute Rolling Block with 1-minute sub-intervals
bIntvl	
rollingBlock5MinIntvl1MinSubIntvl	5-minute Rolling Block with 1-minute sub-intervals

5.7.5.2.1.40. PhaseCodeKind (Enumeration)

Enumeration of phase identifiers. Allows designation of phases for both transmission and distribution equipment, circuits and loads.

Residential and small commercial loads are often served from single-phase, or split-phase, secondary circuits. Phases 1 and 2 refer to hot wires that are 180 degrees out of phase, while N refers to the neutral wire. Through single-phase transformer connections, these secondary circuits may be served from one or two of the primary phases A, B, and C. For three-phase loads, use the A, B, C phase codes instead of s12N.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\PhaseCodeKind.*

Table 5.322 - PhaseCodeKind Enumerated Values

Enumerated Value	Definition
none	Not applicable to any phase
phaseN	Neutral
phaseNtoGnd	Neutral to ground
phaseC	Phase C
phaseCtoN	Phase C to Neutral
phaseCtoAv	Phase C current or voltage relative to Phase A voltage
phaseCAtoN	CA to Neutral
phaseB	Phase B
phaseBtoN	Phase B to Neutral
phaseBtoC	Phase B to C
phaseBtoAv	Phase B current or voltage relative to Phase A voltage
phaseBCtoN	BC to Neutral
phaseA	Phase A
phaseAtoN	Phase A to Neutral
phaseAtoB	Phase A to B
phaseAtoAv	Phase A current relative to Phase A voltage
phaseABtoN	AB to Neutral
phasesABC	Involving all phases
phaseABCtoN	ABC to Neutral
s2	Phase S2
s2N	Phase S2 to Neutral
s1	Phase S1
s1N	Phase S1 to Neutral
s12	Phase S1 to S2
s12N	Phase S1, S2 to Neutral
threeWireWye	Three wire Wye
fourWireWye	Four wire Wye
threeWireDelta	Three wire Delta
fourWireDelta	Four wire Delta

Enumerated Value	Definition
fourWireHLDelta	Four wire High-leg Delta
fourWireOpenDelta	Four wire Open Delta
networked	Networked meter

5.7.5.2.1.41. TimeAttributeKind (Enumeration)

This value identifies the time period during which this measurement was performed. For example if averaged (see DataQualifierKind) what period is averaged. Other complex time periods for how a measurement is computed are identified.

UML Element Location: *Model\UMLModel Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAPI0 EUI\TimeAttributeKind.*

Table 5.323 - TimeAttributeKind Enumerated Values

Enumerated Value	Definition
twelveMinute	12-minute
twoHour	2-hour
fourHour	4-hour
sixHour	6-hour
twelveHour	12-hour
specifiedInterval	The interval length is described in RationalNumber.numerator in seconds. Attribute RationalNumber.denominator should be '1' for whole seconds.
specifiedFixedBlock	The fixed block duration is described in RationalNumber.numerator in seconds. RationalNumber.denominator should be '1' for whole seconds.
specifiedRollingBlock	The rolling block size is described by RationalNumber.numerator in seconds, and the sub-interval size by RationalNumber.denominator in seconds.
none	Not Applicable
tenMinute	10-minute
fifteenMinute	15-minute
oneMinute	1-minute
twentyfourHour	24-hour
thirtyMinute	30-minute
fiveMinute	5-minute
sixtyMinute	60-minute
twoMinute	2-minute
threeMinute	3-minute
present	Within the present period of time
previous	Shifted within the previous monthly cycle and data set
twentyMinute	20-minute interval
fixedBlock60Min	60-minute Fixed Block

Enumerated Value	Definition
fixedBlock30Min	30-minute Fixed Block
fixedBlock20Min	20-minute Fixed Block
fixedBlock15Min	15-minute Fixed Block
fixedBlock10Min	10-minute Fixed Block
fixedBlock5Min	5-minute Fixed Block
fixedBlock1Min	1-minute Fixed Block
rollingBlock60MinIntvl30MinS ubIntvl	60-minute Rolling Block with 30-minute sub-intervals
rollingBlock60MinIntvl20MinS ubIntvl	60-minute Rolling Block with 20-minute sub-intervals
rollingBlock60MinIntvl15MinS ubIntvl	60-minute Rolling Block with 15-minute sub-intervals
rollingBlock60MinIntvl12MinS ubIntvl	60-minute Rolling Block with 12-minute sub-intervals
rollingBlock60MinIntvl10MinS ubIntvl	60-minute Rolling Block with 10-minute sub-intervals
rollingBlock60MinIntvl6MinSu bIntvl	60-minute Rolling Block with 6-minute sub-intervals
rollingBlock60MinIntvl5MinSu bIntvl	60-minute Rolling Block with 5-minute sub-intervals
rollingBlock60MinIntvl4MinSu bIntvl	60-minute Rolling Block with 4-minute sub-intervals
rollingBlock30MinIntvl15MinS ubIntvl	30-minute Rolling Block with 15-minute sub-intervals
rollingBlock30MinIntvl10MinS ubIntvl	30-minute Rolling Block with 10-minute sub-intervals
rollingBlock30MinIntvl6MinSu bIntvl	30-minute Rolling Block with 6-minute sub-intervals
rollingBlock30MinIntvl5MinSu bIntvl	30-minute Rolling Block with 5-minute sub-intervals
rollingBlock30MinIntvl3MinSu bIntvl	30-minute Rolling Block with 3-minute sub-intervals
rollingBlock15MinIntvl5MinSu bIntvl	15-minute Rolling Block with 5-minute sub-intervals
rollingBlock15MinIntvl3MinSu bIntvl	15-minute Rolling Block with 3-minute sub-intervals

Enumerated Value	Definition
rollingBlock15MinIntvl1MinSubIntvl	15-minute Rolling Block with 1-minute sub-intervals
rollingBlock10MinIntvl5MinSubIntvl	10-minute Rolling Block with 5-minute sub-intervals
rollingBlock10MinIntvl2MinSubIntvl	10-minute Rolling Block with 2-minute sub-intervals
rollingBlock10MinIntvl1MinSubIntvl	10-minute Rolling Block with 1-minute sub-intervals
rollingBlock5MinIntvl1MinSubIntvl	5-minute Rolling Block with 1-minute sub-intervals

5.7.6. WXXM

This package, derived from the Weather Exchange Model (WXXM), contains information elements used in the Energy Manager Component to represent weather observations, hazards, and forecasts.

FSGIM has imported this UML Model, Version 1.1.1., and tailored it as follows:

1. A subset of classes needed for FSGIMWeather were selected.
2. Primitive data types were normalized to FSGIM common primitive types.
3. The inheritance hierarchy was abbreviated while preserving all inherited attributes and relationships. Local versions of deeply nested inheritance were simplified by hand-assembling equivalent classes.
4. Enumerations were coerced to the enumeration style and stereotypes of FSGIM.

These modifications were designed to capture the desired subset of functionality, improve consistency, and preserve all semantics of the base WXXM standard. In each case, the FSGIM class retains the name of the WXXM class and additionally, the XMI identifier of the original to provide traceability back to the original class.

UML Package Location: *Model\Model Elements from External Sources\WXXM*

5.7.6.1. AVWX

Aviation-Specific Weather Layer of WXXM.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX*

5.7.6.1.1. AVWX_AreaOfInterestWx

Area of interest for aviation-specific weather classes.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_AreaOfInterestWx*

5.7.6.1.2. AVWX_DataTypes

Data types specific to aviation weather.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_DataTypes*

5.7.6.1.3. AVWX_Features

Features specific to aviation weather.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Features*

5.7.6.1.4. AVWX_Measures

Measurements specific to aviation weather.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Measures*

5.7.6.1.5. AVWX_Phenom_Hazards

Phenom hazards specific to aviation weather.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards*

5.7.6.1.6. AVWX_Phenom_TropicalCyclone

Tropical cyclone classes.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_TropicalCyclone*

5.7.6.1.7. AVWX_Phenom_Volcanic

Volcanic phenomenon.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Volcanic*

5.7.6.1.8. AVWX_Phenom_Wind

Wind phenomenon.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Wind*

5.7.6.1.9. ICAO_Annex_3_Base

Basic classes from the International Civil Aviation Organization (ICAO), Annex 3 - Meteorological Services for International Air Navigation.

UML Package Location: *Model\Model Elements from External Sources\WXXM\AVWX\ICAO_Annex_3_Base*

5.7.6.2. WX

General Weather Layer classes.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX*

5.7.6.2.1. Measures

Basic measurements for weather observations.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\Measures*

5.7.6.2.2. WX_Base

Weather schema that builds upon a GML 3.2 base.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Base*

5.7.6.2.3. WX_Cloud

Description of cloud phenomena.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Cloud*

5.7.6.2.4. WX_Coverage

Descriptions of coverage.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Coverage*

5.7.6.2.5. WX_DataTypes

Data types used in weather descriptions.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes*

5.7.6.2.6. WX_Features

Defines basic area of interest for a weather observation or report including specific details.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Features*

5.7.6.2.7. WX_Forecast

Defines a basic weather forecast.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Forecast*

5.7.6.2.8. WX_Measures

Additional core weather measurements.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures*

5.7.6.2.9. WX_MotionVector

Definition of the motion vector of moving weather phenomena.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_MotionVector*

5.7.6.2.10. WX_Observation

Weather-specific description of OGC Observation and Measurement standards.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Observation*

5.7.6.2.11. WX_PhenomBase

Base description of a weather phenomenon.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase*

5.7.6.2.12. WX_Process

Describes the process used to produce weather observations and reports.

UML Package Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Process*

5.7.6.3. CommonUsagePackages

Collection of utilities that have not attained formal governance status, though in some cases they are on a standardization track.

These are recommended best practice within the HollowWorld modelling framework.

UML Package Location: *Model\Model Elements from External Sources\WXXM\CommonUsagePackages*

5.7.6.3.1. ISO 19136 Annex D.3

Classes corresponding to types and elements provided by GML that do not simply implement existing ISO 19100 classes from the Harmonized Model.

See ISO 19136 Annex D.3.

UML Package Location: *Model\Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3*

5.7.6.4. ISO Standard Packages

ISO 19100 Harmonized Model 2006-06-14 version

- Converted from Rose to EA by John Herring
- some elements mistakenly dropped in conversion.
- 19103/Records package patched by Simon Cox, reinstating Schema and Record
- 19109 package patched by Simon Cox, reinstating GF_ *AssociationType classes
- Package Client/Supplier diagrams moved to new package to reduce clutter in Project browser

SJDC 2007-02-14

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages*

5.7.6.4.1. ISO 03166 Country Codes

ISO standard enumeration for country codes.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 03166 Country Codes*

5.7.6.4.2. ISO 19139 Metadata

Iso standard for implementation of geographic information metadata in XML schema from ISO 19115.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19139 Metadata*

5.7.6.4.2.1. Cultural and linguistic adaptability

Defines the cultural locale for a location including language, country, and character set.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19139 Metadata\Cultural and linguistic adaptability*

5.7.6.4.3. ISO 00639 Human Language

ISO standard enumeration for language codes.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 00639 Human Language*

5.7.6.4.4. ISO 19107 Spatial Schema

ISO standard types for spatial schema.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19107 Spatial Schema*

5.7.6.4.4.1. Geometry

Geometry classes.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19107 Spatial Schema\ Geometry*

5.7.6.4.4.1.1. Coordinate geometry

Coordinate geometry classes.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19107 Spatial Schema\Geometry\Coordinate geometry*

5.7.6.4.5. ISO 19111 Spatial Referencing by Coordinates

ISO standard coordinate reference system.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19111 Spatial Referencing by Coordinates*

5.7.6.4.5.1. Coordinate Reference Systems

Coordinate reference system for geographic locations.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19111 Spatial Referencing by Coordinates\Coordinate Reference Systems*

5.7.6.4.6. ISO 19115 Metadata

ISO standard for various geographic information metadata.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata*

5.7.6.4.6.1. Citation and responsible party information

Classes for citation and responsible party information.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information*

5.7.6.4.6.2. Data quality information

Data quality metadata.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Data quality information*

5.7.6.4.6.3. Extent information

Metadata about spatial, vertical, and temporal extent.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Extent information*

5.7.6.4.6.4. Identification information

Enumeration of the character coding standard used in the resource.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Identification information*

5.7.6.4.6.5. Maintenance information

This package supports the provision of metadata related to the scope and frequency of maintenance for a resource or of metadata about a resource in a single class,

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Maintenance information*

5.7.6.4.6.6. Metadata entity set information

Description of the base metadata class.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Metadata entity set information*

5.7.6.4.6.7. Reference system information

Generic identifier class used to associate a reference code with an identification authority.

UML Package Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Reference system information*

5.7.6.5. SWE

This package contains components associated with the Open Geospatial Consortium Sensor Web Enablement initiative

UML Package Location: *Model\Model Elements from External Sources\WXXM\SWE*

5.7.6.5.1. OM1_Observation

The Observation Schema

Documented in OGC 07-022r1

UML Package Location: *Model\Model Elements from External Sources\WXXM\SWE\OM1_Observation*

5.7.6.5.1.1. observation

The basic Observation model.

UML Package Location: *Model\Model Elements from External Sources\WXXM\SWE\OM1_Observation\observation*

5.7.6.5.2. OM2_Sampling

A model for sampling features.

These feature-types are associated with making observations at locations that are considered in some way to be representative of a domain feature. Sampling features embody a sampling strategy that is suitable for the observation procedure and the observed property. Sampling features are artefacts of the observation process rather than the inherent domain semantics. Similar sampling strategies are used across a wide range of application domains so a common sampling features schema may be described.

Instances of sampling features may act as the proximate feature-of-interest for observations [O&M], particularly early in the value-adding chain.

Documented in OGC 07-002r3

UML Package Location: *Model\Model Elements from External Sources\WXXM\SWE\OM2_Sampling*

5.7.6.5.2.1. samplingBase

Sampling features are feature types that are used primarily for making observations.

UML Package Location: *Model\Model Elements from External Sources\WXXM\SWE\OM2_Sampling\samplingBase*

5.7.6.5.3. SensorML

Stubs for SensorML.

UML Package Location: *Model\Model Elements from External Sources\WXXM\SWE\SensorML*

5.7.6.5.3.1. process

SensorML process descriptor.

UML Package Location: *Model\Model Elements from External Sources\WXXM\SWE\SensorML\process*

5.7.6.6. Elements Defined in the WXXM Model

The following elements are defined in this model.

5.7.6.6.1. Elements Defined in the AVWX Model

The following elements are defined in this model.

5.7.6.6.1.1. Elements Defined in the AVWX_AreaOfInterestWx Model

The following elements are defined in this model.

5.7.6.6.1.1.1. AerodromeWx «FeatureType» (Class)

Area of interest for weather-related observations and forecasts around an aerodrome.

The WXXM Profile has removed sea Wx, qfe, qnh, and the override of isDefinedFor in this class. These attributes do not have meaning within FSGIM because they are particular to measurements made specifically in support of aircraft.

Parent Class: AviationWx (See Clause 5.7.6.6.1.1.2.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_AreaOfInterestWx\AerodromeWx.*

Table 5.324 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
cavok	Ceiling and Visibility are OK. A term used in aviation to indicate that (1) the higher of there are no clouds below 5000 feet above aerodrome level, or, the minimum sector altitude; and no cumulonimbus or towering cumulus; (2) visibility is at least 10 kilometres (6 statute miles); and (3) no current or forecast significant weather such as precipitation, thunderstorms, shallow fog or low drifting snow.	Boolean See Clause 6.6.3.3.	[1]

This element has the following connections to other elements:

Table 5.325 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	cloudCondition	Cloud condition.	CloudCondition See Clause 5.7.6.6.2.3.1.	[0..*]
Association	colourState	Aviation Colour Code is present if there is a nearby volcano state.	AviationColourCode See Clause 5.7.6.6.1.2.1.	[0..1]
Association	wxCondition	Weather condition description.	WxCondition See Clause 5.7.6.6.2.1.1.5.	[0..*]

5.7.6.6.1.1.2. AviationWx «FeatureType» (Class)

Generic area of interest for weather-related observations and forecasts. Contains properties relating to general weather conditions.

Parent Class: AbstractWxFeature (See Clause 5.7.6.6.2.2.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_AreaOfInterestWx\AviationWx.*

Table 5.326 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
airPressure	The air pressure.	Pressure See Clause 5.7.6.6.2.1.7.	[1]
airTemperature	The air temperature.	AirTemperature See Clause 5.7.6.6.2.8.1.	[1]
dewpointTemperature	The dew point temperature.	AirTemperature See Clause 5.7.6.6.2.8.1.	[1]
horizontalVisibility	Horizontal visibility.	HorizontalVisibility See Clause 5.7.6.6.1.1.3.	[1]
isDefinedFor	Provides a means to explicitly link to the corresponding feature.	AbstractFeature See Clause 5.7.6.6.3.1.1.	[1]
verticalVisibility	Vertical visibility.	VerticalVisibilityDistance	[1]
verticalWindGust	Vertical wind gust.	WindGust See Clause 5.7.6.6.2.8.9.	[1]
windDirection	Wind direction.	WindDirection See Clause 5.7.6.6.2.8.8.	[1]
windGust	Wind gust.	WindGust See Clause 5.7.6.6.2.8.9.	[1]
windSpeed	Wind speed.	WindSpeed See Clause 5.7.6.6.2.8.10.	[1]

5.7.6.6.1.1.3. **HorizontalVisibility** «FeatureType» (Class)
Horizontal visibility.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_AreaOfInterestWX\HorizontalVisibility.*

Table 5.327 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
minimumVisibility	Minimum visibility.	HorizontalVisibilityDis tance See Clause 5.7.6.6.2.8.5.	[1]
prevailingVisibility	The prevailing visibility.	HorizontalVisibilityDis tance See Clause 5.7.6.6.2.8.5.	[1]
unidirectionalVisibility	True if visibility is unidirectional.	Boolean See Clause 6.6.3.3.	[1]

This element has the following connections to other elements:

Table 5.328 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	directionMinimum	Direction minimum.	Compass8 See Clause 5.7.6.6.2.5.6.	[0..1]

5.7.6.6.1.1.4. **SurfaceDeposit** «FeatureType» (Class)
Surface deposit.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_AreaOfInterestWX\SurfaceDeposit.*

Table 5.329 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
contaminationExtent	Extent of contamination.	Percentage See Clause 5.7.6.6.2.1.6.	[1]
depositDepth	Depth of deposit.	Depth See Clause 5.7.6.6.2.8.3.	[1]

This element has the following connections to other elements:

Table 5.330 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	surfaceDepositType	Surface deposit type	SurfaceDepositTypes See Clause 5.7.6.6.1.1.6.	[0..1]

5.7.6.6.1.1.5. SeaState (Enumeration)

Enumeration for state of the sea descriptions.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_AreaOfInterestWX\SeaState.*

Table 5.331 – SeaState Enumerated Values

Enumerated Value	Definition
CALM GLASSY	Sea state is calm glassy.
CALM RIPPLED	Sea state is calm rippled.
SMOOTH WAVELETS	Sea state is smooth wavelets.
SLIGHT	Sea state is slight.
MODERATE	Sea state is moderate.
ROUGH	Sea state is rough.
VERY ROUGH	Sea state is very rough.
HIGH	Sea state is high.
VERY HIGH	Sea state is very high.
PHENOMENAL	Sea state is phenomenal.

5.7.6.6.1.1.6. SurfaceDepositTypes (Enumeration)

Enumeration for types of surface contamination.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_AreaOfInterestWX_SurfaceDepositTypes.*

Table 5.332 - SurfaceDepositTypes Enumerated Values

Enumerated Value	Definition
CLEAR AND DRY	Surface deposit type is clear and dry.
DAMP	Surface deposit type is damp.
WET AND WATER PATCHES	Surface deposit type is wet and water patches.
RIME AND FROST	Surface deposit type is rime and frost.
DRY SNOW	Surface deposit type is dry snow.
WET SNOW	Surface deposit type is wet snow.
SLUSH	Surface deposit type is slush.
ICE	Surface deposit type is ice.
COMPACTED SNOW	Surface deposit type is compacted snow.
FROZEN RUTS	Surface deposit type is frozen ruts.
NOT REPORTED	Surface deposit type is not reported.

5.7.6.6.1.2. Elements Defined in the AVWX_DataTypes Model

The following elements are defined in this model.

5.7.6.6.1.2.1. AviationColourCode (Enumeration)

Enumeration for aviation colour codes. A system of four colour codes used to indicate conditions of volcanic ash that may be important to the safety of aircraft (or conditions on the ground).

- GREEN - volcano is normal non-erupting state
- YELLOW - volcano is experiencing signs of unrest
- ORANGE - volcano is exhibiting heightened unrest with increased likelihood of eruption
- RED - eruption is forecast to be immanent

Detailed descriptions from: <http://www.wovo.org/aviation-colour-codes.html>

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_DataTypes\AviationColourCode.*

Table 5.333 – AviationColourCode Enumerated Values

Enumerated Value	Definition
ORANGE	Volcano is exhibiting heightened unrest with increased likelihood of eruption. or, Volcanic eruption is underway with no or minor ash emission. [specify ash-plume height if possible]
RED	Eruption is forecast to be imminent with significant emission of ash into the atmosphere likely. or, Eruption is underway with significant emission of ash into the atmosphere. [specify ash-plume height if possible]
GREEN	Volcano is in normal, non-eruptive state. or, after a change from a higher level: Volcanic activity considered to have ceased, and volcano reverted to its normal, non-eruptive state.
YELLOW	Volcano is experiencing signs of elevated unrest above known background levels. or, after a change from higher level: Volcanic activity has decreased significantly but continues to be closely monitored for possible renewed increase.
NOT GIVEN	Volcanic activity is not given.
UNKNOWN	Volcanic activity is unknown.
NIL	Volcanic activity is nil.

5.7.6.6.1.2.2. ContaminationExtent (Enumeration)

Enumeration for contamination extent ranges.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_DataTypes\ContaminationExtent.*

Table 5.334 – ContaminationExtent Enumerated Values

Enumerated Value	Definition
<10%	Contamination extent <10%.
11-25%	Contamination extent 11-25%.
26-50%	Contamination extent 26-50%.
51-100%	Contamination extent 51-100%.
NOT REPORTED	Contamination not reported.

5.7.6.6.1.3. Elements Defined in the AVWX_Features Model

The following elements are defined in this model.

5.7.6.6.1.3.1. Aerodrome «FeatureType» (Class)
Stand-in class for AIXM equivalent (Airport/Heliport).

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXXM\AVWX\AVWX_Features\Aerodrome*.

This element has the following connections to other elements:

Table 5.335 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	position	Position of feature.	DirectPosition See Clause 5.7.6.6.4.4.3.	[0..1]

5.7.6.6.1.3.2. Sea «FeatureType» (Class)
Sea feature.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXXM\AVWX\AVWX_Features\Sea*.

5.7.6.6.1.3.3. Volcano «FeatureType» (Class)
Volcano feature.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXXM\AVWX\AVWX_Features\Volcano*.

Table 5.336 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
summitElevation	Summit elevation.	VerticalDistance See Clause 5.7.6.6.2.8.6.	[1]
iavceiIdentifier	The International Association for Volcanology and Chemistry of Earth's Interior (IAVCEI), The World Organization of Volcano Observatories (WOVO), and the Global Volcano Model (GVM) have sanctioned GVP to assign official names and numbers to the world's volcanoes.	String See Clause 6.6.3.7.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
position	The position of the volcano.	DirectPosition See Clause 5.7.6.6.4.4.3.	[0..1]

This element has the following connections to other elements:

Table 5.337 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	position	Position of feature.	DirectPosition See Clause 5.7.6.6.4.4.3.	[0..1]

5.7.6.6.1.4. Elements Defined in the AVWX_Measures Model

The following elements are defined in this model.

5.7.6.6.1.4.1. QPressure «FeatureType» (Class)

Restricts Pressure to an integer value between 500 and 1100 (inclusive).

Parent Class: Pressure (See Clause 5.7.6.6.2.1.7.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Measures\QPressure.*

5.7.6.6.1.5. Elements Defined in the AVWX_Phenom_Hazards Model

The following elements are defined in this model.

5.7.6.6.1.5.1. Convection «FeatureType» (Class)

Convection hazard.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\Convection.*

This element has the following connections to other elements:

Table 5.338 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	growth	Expected growth.	Growth See Clause 5.7.6.6.2.5.8.	[0..1]
Association	intensity	Intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Expected movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.5.2. Icing «FeatureType» (Class)

Icing hazard.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXXM\AVWX\AVWX_Phenom_Hazards\Icing.*

Table 5.339 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
accretionRate	Rate of accretion of ice.	Real See Clause 6.6.3.6.	[0..1]

This element has the following connections to other elements:

Table 5.340 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	movement	Movement description.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]
Association	type	Icing types.	IcingTypes See Clause 5.7.6.6.1.5.7.	[0..1]

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

5.7.6.6.1.5.3. MountainObscuration «FeatureType» (Class)
Mountain obscuration hazard.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\MountainObscuration*.
This element has the following connections to other elements:

Table 5.341 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.5.4. MountainWave «FeatureType» (Class)
Mountain wave hazard.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\MountainWave*.
This element has the following connections to other elements:

Table 5.342 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	downdraft	Speed of downdraft.	Speed See Clause 5.7.6.6.2.1.8.	[0..1]
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	intensity	Intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.5.5. Turbulence «FeatureType» (Class)
Turbulence hazard.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2-1(1.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\Turbulence.*

Table 5.343 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
edr	Eddy dissipation rate.	Real See Clause 6.6.3.6.	[0..1]

This element has the following connections to other elements:

Table 5.344 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]
Association	type	Turbulence type.	TurbulenceTypes See Clause 5.7.6.6.1.5.8.	[0..1]

5.7.6.6.1.5.6. WindShear «FeatureType» (Class)
Wind shear hazard.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXXM\A\WX\A\WX_Phenom_Hazards\WindShear.*

Table 5.345 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
level	Level of wind shear.	VerticalDistance See Clause 5.7.6.6.2.8.6.	[1]

This element has the following connections to other elements:

Table 5.346 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.5.7. IcingTypes (Enumeration)
Enumeration for types of icing.

UML Element Location: *Model\UML Model Elements from External Sources\WXXXM\A\WX\A\WX_Phenom_Hazards\IcingTypes.*

Table 5.347 - IcingTypes Enumerated Values

Enumerated Value	Definition
CLEAR	Icing type is clear.
MIXED	Icing type is mixed.
RIME	Icing type is rime.

5.7.6.6.1.5.8. TurbulenceTypes (Enumeration)

Enumeration for types of turbulence.

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Hazards\TurbulenceTypes*.

Table 5.348 - TurbulenceTypes Enumerated Values

Enumerated Value	Definition
CAT	Clear Air Turbulence.
CHOP	Chop air turbulence.

5.7.6.6.1.6. Elements Defined in the AVWX_Phenom_TropicalCyclone Model

The following elements are defined in this model.

5.7.6.6.1.6.1. TropicalCyclone «FeatureType» (Class)

Tropical cyclone hazard.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_TropicalCyclone\TropicalCyclone*.

Table 5.349 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
centralPressure	Central pressure of cyclone.	Pressure See Clause 5.7.6.6.2.1.7.	[1]
maxSurfaceWindSpeed	Maximum surface wind.	WindSpeed See Clause 5.7.6.6.2.8.10.	[1]
centre	The centre position of the cyclone.	DirectPosition See Clause 5.7.6.6.4.4.3.	[0..1]

This element has the following connections to other elements:

Table 5.350 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	centre	Center position of cyclone.	DirectPosition See Clause 5.7.6.6.4.4.3.	[0..1]
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.7. Elements Defined in the AVWX_Phenom_Volcanic Model

The following elements are defined in this model.

5.7.6.6.1.7.1. VolcanicActivity «FeatureType» (Class)
Volcanic activity.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_Volcanic\VolcanicActivity*.

Table 5.351 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
volcano	Volcano hazard.	Volcano See Clause 5.7.6.6.1.3.3.	[1]
eruption	True if volcanic activity is an eruption.	Boolean See Clause 6.6.3.3.	[0..1]

This element has the following connections to other elements:

Table 5.352 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.7.2. VolcanicAshCloud «FeatureType» (Class)

Volcanic ash cloud.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\AVWX_Phenom_VolcanicAshCloud.*

This element has the following connections to other elements:

Table 5.353 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.8. Elements Defined in the AVWX_Phenom_Wind Model

The following elements are defined in this model.

5.7.6.6.1.8.1. WindLayer «FeatureType» (Class)
Wind layer hazard.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXXM\4VWX\4VWX_Phenom_WindWindLayer.*

Table 5.354 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
direction	Direction of wind layer.	WindDirection See Clause 5.7.6.6.2.8.8.	[1]
directionVariesFrom	Direction varies from this limit.	WindDirection See Clause 5.7.6.6.2.8.8.	[1]
directionVariesTo	Direction varies to this limit.	WindDirection See Clause 5.7.6.6.2.8.8.	[1]
height	Height of wind layer.	VerticalDistance See Clause 5.7.6.6.2.8.6.	[1]
position	Position of wind layer.	DirectPosition See Clause 5.7.6.6.4.4.3.	[1]
speed	Speed of wind layer.	WindSpeed See Clause 5.7.6.6.2.8.10.	[1]
speedVariesFrom	Speed varies from this limit.	WindSpeed See Clause 5.7.6.6.2.8.10.	[1]
speedVariesTo	Speed varies to this limit.	WindSpeed See Clause 5.7.6.6.2.8.10.	[1]

This element has the following connections to other elements:

Table 5.355 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.1.9. Elements Defined in the ICAO Annex 3 Base Model

The following elements are defined in this model.

5.7.6.6.1.9.1. AbstractReport «FeatureType» (Abstract Class)

Abstract report base class.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\AVWX\ICAO Annex 3 Base\AbstractReport.*

Table 5.356 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
process	The process related to the distribution or collection of weather reports.	ReportProcess See Clause 5.7.6.6.2.12.1.	[1]
issueTime	When the report was issued.	UTCDateTime See Clause 6.6.1.11.	[1]
rawText	Arbitrary associated text potentially from a reporting station in an older text-based format.	String See Clause 6.6.3.7.	[1]
automated	True if report is automated.	Boolean See Clause 6.6.3.3.	[1]
missing	%%%% True if report source data is missing.	Boolean See Clause 6.6.3.3.	[1]

This element has the following connections to other elements:

Table 5.357 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	appliesTo	Applies to Aerodrome.	Aerodrome See Clause 5.7.6.6.1.3.1.	[0..1]
Association	wxForecast	Weather forecast(s) applies to.	Forecast See Clause 5.7.6.6.2.7.1.	[0..*]
Association	wxObservation	Observation(s) applies to.	Observation See Clause 5.7.6.6.2.10.1.	[1..*]

5.7.6.6.1.9.2. TREND «FeatureType» (Class)
TREND forecast.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\A\WX\ICAO Annex 3_Base\TREND.*

Table 5.358 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
changePeriod	Change period of trend.	Duration See Clause 6.6.1.9.	[1]

This element has the following connections to other elements:

Table 5.359 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	forecast	Sequence of forecasts for this Trend.	Forecast See Clause 5.7.6.6.2.7.1.	[0..*]
Association	trendChangeIndicator	Expected change.	ForecastChangeIndicator See Clause 5.7.6.6.2.5.7.	[0..1]

5.7.6.6.1.9.3. NextAdvisoryText (Enumeration)

Enumeration of next advisory projection.

UML Element Location: *Model\Model Elements from External Sources\WXXM\A\WX\ICAO_Annex_3_Base\NextAdvisoryText.*

Table 5.360 - NextAdvisoryText Enumerated Values

Enumerated Value	Definition
NO LATER THAN	Next advisory: no later than.
WILL BE ISSUED BY	Next advisory: will be issued by.
NO FURTHER ADVISORIES	Next advisory: no further advisories.

5.7.6.6.2. Elements Defined in the WX Model

The following elements are defined in this model.

5.7.6.6.2.1. Elements Defined in the Measures Model

The following elements are defined in this model.

5.7.6.6.2.1.1. AbstractMeasure «Type» (Abstract Class)

Abstract measurement class. Each measurement has a unit of measure, uom. WXXM allows more than one uom -- e.g., F/C/K for temperature. FSGIM only allows one choice of uom for a physical quantity.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\AbstractMeasure.*

Table 5.361 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
value	The value of the measure.	Real See Clause 6.6.3.6.	[1]

This element has the following connections to other elements:

Table 5.362 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	powerOffTenMultiplier	Default is 1.	SiScaleCodeType See Clause 6.6.2.3.	[0..1]
Association	uom	Units of measure for the measurement.	UnitsSymbolKind See Clause 6.6.2.6.	[1]

5.7.6.6.2.1.2. Angle «Type» (Class)

Generic angle class.

Parent Class: AbstractMeasure (See Clause 5.7.6.6.2.1.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\Measures\Angle*.

This element has the following connections to other elements:

Table 5.363 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Unit of measure for angle. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be rad	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.7.6.6.2.1.3. Bearing «Type» (Class)

Bearing is used to represent direction in the 2D coordinate reference system. It is an "angle measured from true north". It is represented as an angle restricted to the range greater than 0 and up to and including 360 degrees which is true north.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\Measures\Bearing*.

Table 5.364 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
maxInclusive	The value of true north, 360.	Integer See Clause 6.6.3.5.	[1]
minInclusive	Must be greater than 0.	Integer See Clause 6.6.3.5.	[1]

5.7.6.6.2.1.4. Distance «Type» (Class)

Generic distance class.

Parent Class: AbstractMeasure (See Clause 5.7.6.6.2.1.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\Measures\Distance*.

This element has the following connections to other elements:

Table 5.365 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Units of measure for distance. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be m	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.7.6.6.2.1.5. Luminance «Type» (Class)

Density of luminous intensity.

Parent Class: AbstractMeasure (See Clause 5.7.6.2.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WXXMeasures\Luminance.*

This element has the following connections to other elements:

Table 5.366 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Unit of measure for luminance. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be sb	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.7.6.6.2.1.6. Percentage «Type» (Class)

Generic percentage class; value is limited to between 0 and 100 inclusive.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WXXMeasures\Percentage.*

Table 5.367 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
maxInclusive	Maximum inclusive value.	Integer See Clause 6.6.3.5.	[1]
minInclusive	Minimum inclusive value.	Integer See Clause 6.6.3.5.	[1]

5.7.6.6.2.1.7. Pressure «Type» (Class)
Generic pressure class.

Parent Class: AbstractMeasure (See Clause 5.7.6.6.2.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Pressure.*

This element has the following connections to other elements:

Table 5.368 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Unit of measure for pressure. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be pa	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.7.6.6.2.1.8. Speed «Type» (Class)
Generic speed class.

Parent Class: AbstractMeasure (See Clause 5.7.6.6.2.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Speed.*

This element has the following connections to other elements:

Table 5.369 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Unit of measure for speed. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be mPerS	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.7.6.6.2.1.9. Temperature «Type» (Class)
Generic temperature class.

Parent Class: AbstractMeasure (See Clause 5.7.6.6.2.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Measures\Temperature*.
This element has the following connections to other elements:

Table 5.370 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	uom	Unit of measure for temperature. (This association redefines the uom association between the AbstractMeasure Class (see Clause 5.7.6.6.2.1.1) and the UnitSymbolKind Enumeration (see Clause 6.6.2.6).) Constraint: must be degK	UnitSymbolKind See Clause 6.6.2.6.	[1]

5.7.6.6.2.2. Elements Defined in the WX_Base Model

The following elements are defined in this model.

5.7.6.6.2.2.1. AbstractWxFeature «FeatureType» (Abstract Class)
Abstract weather feature.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Base\AbstractWxFeature*.

Table 5.371 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
obsOrFstTime	Forecast time for object. For a moving feature that is being tracked (e.g., GustFront), this will often be the time instant corresponding to the forecasted position. For a forecast of conditions for a general area, the time will often be the period of time associated with the forecast. For example, 'between 2 and 3 PM, there will be a wind shift in area X'.	UTCDateTime See Clause 6.6.1.11.	[1]
phenomenon	Phenomenon associated with feature.	AbstractPhenomenon See Clause 5.7.6.6.2.1.1.1.	[0..*]
validTime	Time period over which the feature is considered valid. Most commonly used in conjunction with forecasts. Valid time is logically separate from a forecast time in that it describes a time period within which a forecast is in effect. So a forecast that describes 'Snow predicted between 6 and 10 PM this evening' may have a valid time that starts at the current time, while the forecast period itself runs from 6 to 10.	UTCDateTime See Clause 6.6.1.11.	[1]

This element has the following connections to other elements:

Table 5.372 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	associatedFeature	Associated additional feature.	Sea See Clause 5.7.6.6.1.3.2.	[0..1]
Association	associatedFeature	Associated additional feature.	Turbulence See Clause 5.7.6.6.1.5.5.	[0..1]
Association	associatedFeature	Associated additional feature.	AerodromeWx See Clause 5.7.6.6.1.1.1.	[0..1]
Association	associatedFeature	Associated additional feature.	VolcanicAshCloud See Clause 5.7.6.6.1.7.2.	[0..1]
Association	associatedFeature	Associated additional feature.	AbstractFeature See Clause 5.7.6.6.3.1.1.	[0..1]
Association	associatedFeature	Associated additional feature.	WindLayer See Clause 5.7.6.6.1.8.1.	[0..1]
Association	associatedFeature	Associated additional feature.	SurfaceDeposit See Clause 5.7.6.6.1.1.4.	[0..1]
Association	associatedFeature	Associated additional feature.	Convection See Clause 5.7.6.6.1.5.1.	[0..1]
Association	associatedFeature	Associated additional feature.	HorizontalVisibility See Clause 5.7.6.6.1.1.3.	[0..1]
Association	associatedFeature	Associated additional feature.	Icing See Clause 5.7.6.6.1.5.2.	[0..1]
Association	associatedFeature	Associated additional feature.	AviationWx See Clause 5.7.6.6.1.1.2.	[0..1]
Association	associatedFeature	Associated additional feature.	MountainWave See Clause	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	associatedFeature	Associated additional feature.	5.7.6.6.1.5.4. MountainObscuration See Clause 5.7.6.6.1.5.3.	[0..1]
Association	associatedFeature	Associated additional feature.	WindShear See Clause 5.7.6.6.1.5.6.	[0..1]
Association	associatedFeature	Associated additional feature.	TropicalCyclone See Clause 5.7.6.6.1.6.1.	[0..1]
Association	associatedFeature	Associated additional feature.	VolcanicActivity See Clause 5.7.6.6.1.7.1.	[0..1]
Association	changeIndicator	Expected change.	ForecastChangeIndicator See Clause 5.7.6.6.2.5.7.	[0..1]

5.7.6.6.2.2.2. ExtentOf «Union» (Union)

Defines the extent of a feature.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_Base\ExtentOf*.

Table 5.373 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
boundedBy	Envelope defining the extent of the phenomenon.	GM_Envelope See Clause 5.7.6.6.4.4.4.	[1]

5.7.6.6.2.3. Elements Defined in the WX_Cloud Model

The following elements are defined in this model.

5.7.6.6.2.3.1. CloudCondition «FeatureType» (Class)

Cloud condition.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_Cloud\CloudCondition*.

This element has the following connections to other elements:

Table 5.374 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	cloudAmount	Cloud amount.	CloudAmount See Clause 5.7.6.6.2.3.8.	[0..1]
Association	cloudType	Cloud type.	CloudTypes See Clause 5.7.6.6.2.3.9.	[0..1]

5.7.6.6.2.3.2. Hydrometeor «FeatureType» (Class)

Hydrometeor feature.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_Cloud\Hydrometeor*.

Table 5.375 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
partial	True if hydrometeor has partial coverage.	Boolean See Clause 6.6.3.3.	[0..1]
isFreezing	True if the cloud condensate is freezing.	Boolean See Clause 6.6.3.3.	[0..1]
shallow	True if the hydrometeor is shallow.	Boolean See Clause 6.6.3.3.	[0..1]
patchy	True if the hydrometeor is patchy.	Boolean See Clause 6.6.3.3.	[0..1]

This element has the following connections to other elements:

Table 5.376 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	type	Hydrometeor type.	HydrometeorTypes See Clause 5.7.6.6.2.3.10.	[0..1]

5.7.6.6.2.3.3. Lithometeor «FeatureType» (Class)

Lithometeor feature.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_CloudLithometeor.*

Table 5.377 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
isBlowing	True if lithometeor is blowing.	Boolean See Clause 6.6.3.3.	[0..1]
isLowDrifting	True if lithometeor is low drifting.	Boolean See Clause 6.6.3.3.	[0..1]

This element has the following connections to other elements:

Table 5.378 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	type	Obscuration type.	ObscurationTypes See Clause 5.7.6.6.2.11.6.	[0..1]

5.7.6.6.2.3.4. Precipitation «FeatureType» (Class)

Precipitation feature.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_CloudPrecipitation.*

Table 5.379 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
isFreezing	True if precipitation is freezing.	Boolean See Clause 6.6.3.3.	[0..1]

This element has the following connections to other elements:

Table 5.380 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	type	Precipitation type.	PrecipitationTypes See Clause 5.7.6.6.2.11.8.	[0..1]

5.7.6.6.2.3.5. RadioactiveCloud «FeatureType» (Class)

Radioactive cloud feature.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\Model Elements from External Sources\WX\MWX_Cloud\RadioactiveCloud.*

5.7.6.6.2.3.6. Shower «FeatureType» (Class)

Shower feature.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\Model Elements from External Sources\WX\MWX_Cloud\Shower.*

Table 5.381 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
obscured	True if shower is obscured.	Boolean See Clause 6.6.3.3.	[0..1]
embedded	True if shower is embedded.	Boolean See Clause 6.6.3.3.	[0..1]
widespread	True if shower is widespread.	Boolean See Clause 6.6.3.3.	[0..1]

This element has the following connections to other elements:

Table 5.382 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	contains	Contains precipitation.	Precipitation See Clause 5.7.6.6.2.3.4.	[0..*]

5.7.6.6.2.3.7. Thunderstorm «FeatureType» (Class)
Thunderstorm feature.

Parent Class: VerticallyBoundedPhenomenon (See Clause 5.7.6.6.2.11.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_CloudThunderstorm.*

Table 5.383 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
widespread	True if thunderstorm is widespread.	Boolean See Clause 6.6.3.3.	[0..1]
embedded	True if thunderstorm is embedded.	Boolean See Clause 6.6.3.3.	[0..1]
obscured	True if thunderstorm is obscured.	Boolean See Clause 6.6.3.3.	[0..1]

This element has the following connections to other elements:

Table 5.384 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	contains	Contains precipitation.	Precipitation See Clause 5.7.6.6.2.3.4.	[0..*]

5.7.6.6.2.3.8. CloudAmount (Enumeration)
Enumeration for cloud amounts.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_CloudCloudAmount.*

Table 5.385 - CloudAmount Enumerated Values

Enumerated Value	Definition
CLEAR	Cloud amount is clear.
FEW	Cloud amount is few.
SCATTERED	Cloud amount is scattered.
BROKEN	Cloud amount is broken.
OVERCAST	Cloud amount is overcast.
SCATTERED-BROKEN	Cloud amount is scattered-broken.

5.7.6.6.2.3.9. CloudTypes (Enumeration)

Enumeration for types of cloud.

UML Element Location: *Model\Model Elements from External Sources\WX\WX_Cloud\CloudTypes*.

Table 5.386 - CloudTypes Enumerated Values

Enumerated Value	Definition
ALTOCUMULUS	Cloud type is alto cumulus.
ALTOSTRATUS	Cloud type is alto stratus.
CIRROCUMULUS	Cloud type is cirro cumulus.
CIRROSTRATUS	Cloud type is cirro stratus.
CIRRUS	Cloud type is cirrus.
CUMULONIMBUS	Cloud type is cumulonimbus.
CUMULUS	Cloud type is cumulus.
NIMBOSTRATUS	Cloud type is nimbostratus.
STRATOCUMULUS	Cloud type is stratocumulus.
STRATUS	Cloud type is stratus.
TOWERING CUMULUS	Cloud type is towering cumulus.
OTHER	Cloud type is other.

5.7.6.6.2.3.10. HydrometeorTypes (Enumeration)

Enumeration for types of hydrometeor.

UML Element Location: *Model\Model Elements from External Sources\WX\WX_Cloud\HydrometeorTypes*.

Table 5.387 - HydrometeorTypes Enumerated Values

Enumerated Value	Definition
FOG	Hydrometeor type is fog.
MIST	Hydrometeor type is mist.
UNIDENTIFIED	Hydrometeor type is unidentified.

5.7.6.6.2.4. Elements Defined in the WX_Coverage Model

The following elements are defined in this model.

5.7.6.6.2.4.1. AbstractCoverage «FeatureType» (Class)

This element serves as the head of a substitution group which may contain any coverage whose type is derived from gml:AbstractCoverageType. It may act as a variable in the definition of content models where it is required to permit any coverage to be valid.

UML Element Location: *Model\Model Elements from External Sources\WX\WX_Coverage\AbstractCoverage*.

5.7.6.6.2.4.2. AreaCoverage «FeatureType» (Class)

Defines an area coverage.

Parent Class: DiscreteCoverage (See Clause 5.7.6.6.2.4.3.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Coverage\AreaCoverage.*

This element has the following connections to other elements:

Table 5.388 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extent	Extent of area coverage.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]

5.7.6.6.2.4.3. DiscreteCoverage «FeatureType» (Class)

A discrete coverage consists of a domain set, range set and optionally a coverage function. The domain set consists of either spatial or temporal geometry objects, finite in number. The range set is comprised of a finite number of attribute values each of which is associated to every direct position within any single spatiotemporal object in the domain. In other words, the range values are constant on each spatiotemporal object in the domain. This coverage function maps each element from the coverage domain to an element in its range. The coverageFunction element describes the mapping function.

This element serves as the head of a substitution group which may contain any discrete coverage whose type is derived from gml:DiscreteCoverageType.

Parent Class: AbstractCoverage (See Clause 5.7.6.6.2.4.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\Coverage\DiscreteCoverage.*

5.7.6.6.2.5. Elements Defined in the WX_DataTypes Model

The following elements are defined in this model.

5.7.6.6.2.5.1. PercentageRange «DataType» (Class)

Percentage range.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\DataTypes\PercentageRange.*

Table 5.389 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
maximum	Maximum value of percentage.	Percentage See Clause 5.7.6.6.2.1.6.	[1]
minimum	Minimum value of percentage.	Percentage See Clause 5.7.6.6.2.1.6.	[1]

5.7.6.6.2.5.2. WeatherModifier «Union» (Union)
Weather modifier.

UML Element Location: *Model\WeatherModifier*

Table 5.390 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
weatherIntensity	Intensity of weather.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[1]
weatherProximity	Proximity of weather.	WeatherProximity See Clause 5.7.6.6.2.5.13.	[1]

5.7.6.6.2.5.3. WxCode «SimpleType» (Class)
Restriction of weather codes. This includes notation for weather conditions (rain, snow, etc) like that used in METARs and TAFs

UML Element Location: *Model\WxCode*

5.7.6.6.2.5.4. Compass16 (Enumeration)
Enumeration for compass points, including 12 intercardinal points.

UML Element Location: *Model\Compass16*

Table 5.391 – Compass16 Enumerated Values

Enumerated Value	Definition
N	North.
NNE	North-North-East.
NE	North-East.
ENE	East-North-East.
E	East.
ESE	East-South-East.
SE	South-East.
SSE	South-South-East.
S	South.
SSW	South-South-West.
SW	South-West.
WSW	West-South-West.
W	West.
WNW	West-North-West.
NW	North-West.
NNW	North-North-West.

5.7.6.6.2.5.5. Compass4 (Enumeration)

Enumeration for cardinal compass points.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\Compass4.*

Table 5.392 – Compass4 Enumerated Values

Enumerated Value	Definition
N	North.
S	South.
E	East.
W	West.

5.7.6.6.2.5.6. Compass8 (Enumeration)

Enumeration for 8 compass points, including 4 intercardinal points.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\Compass8.*

Table 5.393 - Compass8 Enumerated Values

Enumerated Value	Definition
N	North.
NE	North-East.
E	East.
SE	South-East.
S	South.
SW	South-West.
W	West.
NW	North-West.

5.7.6.6.2.5.7. ForecastChangeIndicator (Enumeration)

Enumeration for forecast change indicator descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\ForecastChangeIndicator.*

Table 5.394 - ForecastChangeIndicator Enumerated Values

Enumerated Value	Definition
BECOMING	Forecast change indicator is becoming.
NO SIGNIFICANT CHANGE	Forecast change indicator is no significant change.
TEMPORARY	Forecast change indicator is temporary.
FROM	Forecast change indicator is from.

5.7.6.6.2.5.8. Growth (Enumeration)

Enumeration for storm growth (taken from CCFP).

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\Growth.*

Table 5.395 - Growth Enumerated Values

Enumerated Value	Definition
FAST POSITIVE	Storm growth is fast and positive.
POSITIVE	Storm growth is positive.
NO CHANGE	Storm growth is no-change.
NEGATIVE	Storm growth is negative.

5.7.6.6.2.5.9. IntensityChange (Enumeration)

Enumeration for intensity change descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\IntensityChange.*

Table 5.396 - IntensityChange Enumerated Values

Enumerated Value	Definition
INTENSIFY	Weather will intensify.
NO CHANGE	No change.
WEAKEN	Weather phenomena will weaken.

5.7.6.6.2.5.10. PastTendency (Enumeration)

Enumeration for past tendency descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\PastTendency.*

Table 5.397 - PastTendency Enumerated Values

Enumerated Value	Definition
DOWNWARD	Past tendency is downward.
NO CHANGE	Past tendency is no change.
UPWARD	Past tendency is upward.

5.7.6.6.2.5.11. WeatherDescriptor (Enumeration)

Enumeration for weather descriptor.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\WeatherDescriptor.*

Table 5.398 - WeatherDescriptor Enumerated Values

Enumerated Value	Definition
BLOWING	Weather characterized as blowing.
FREEZING	Weather characterized as freezing.
DRIFTING	Weather characterized as drifting.
LOW DRIFTING	Weather characterized as low-drifting.
PARTIAL	Weather characterized as partial.
PATCHES	Weather characterized as pathes.
SHALLOW	Weather characterized as shallow.
SHOWERS	Weather characterized as showers.
THUNDERSTORMS	Weather characterized as thunderstorms.

5.7.6.6.2.5.12. WeatherIntensity (Enumeration)

Enumeration for weather intensity descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\WeatherIntensity*.

Table 5.399 - WeatherIntensity Enumerated Values

Enumerated Value	Definition
NONE	A report of no intensity (no icing present, for example) for a condition.. This is different from missing information
LIGHT	Weather intensity is light.
LIGHT-MODERATE	A light to moderate intensity. This is used by AIR/SIGMETs
MODERATE	Weather intensity is moderate.
MODERATE-SEVERE	A moderate to severe intensity. This is used by AIR/SIGMETs
HEAVY	Weather intensity is heavy.
SEVERE	Weather intensity is severe.

5.7.6.6.2.5.13. WeatherProximity (Enumeration)

Enumeration for weather intensity descriptions.

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_DataTypes\WeatherProximity*.

Table 5.400 - WeatherProximity Enumerated Values

Enumerated Value	Definition
VICINITY	Weather proximity is vicinity.

5.7.6.6.2.6. Elements Defined in the WX_Features Model

The following elements are defined in this model.

5.7.6.6.2.6.1. AreaOfInterest «FeatureType» (Class)

This represents a generic area of interest such as a radar scan area for use in situations where a soft-type feature of interest is preferable.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Features\AreaOfInterest*.

This element has the following connections to other elements:

Table 5.401 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of feature.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]

5.7.6.6.2.7. Elements Defined in the WX_Forecast Model

The following elements are defined in this model.

5.7.6.6.2.7.1. Forecast «FeatureType» (Class)

Generic weather forecast class. Forecast inherits from O&M observation. This is a temporary fix as this inheritance violates Liskov substitution principle (subclass can't be used as a substitute for the super class).

For a Forecast the samplingTime implies the time period associated with the observations used to produce the forecast.

Parent Class: Observation (See Clause 5.7.6.6.5.1.2.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\ForecastForecast.*

Table 5.402 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
confidence	Percentage confidence in forecast.	Percentage See Clause 5.7.6.6.2.1.6.	[1]
confidenceRange	Range for percentage.	PercentageRange See Clause 5.7.6.6.2.5.1.	[1]
forecastAnalysisTime	Forecast analysis time.	UTCDateTime See Clause 6.6.1.11.	[1]
probability	Probability for this forecast.	Percentage See Clause 5.7.6.6.2.1.6.	[1]
validTime	Time period over which forecast result is valid (can incorporate "expiration time").	UTCDateTime See Clause 6.6.1.11.	[1]

This element has the following connections to other elements:

Table 5.403 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	changeIndicator	Expected change.	ForecastChangeIndicator See Clause 5.7.6.6.2.5.7.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	VolcanicAshCloud See Clause 5.7.6.6.1.7.2.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	Sea See Clause 5.7.6.6.1.3.2.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	Iceing See Clause 5.7.6.6.1.5.2.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	SurfaceDeposit See Clause 5.7.6.6.1.1.4.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	HorizontalVisibility See Clause 5.7.6.6.1.1.3.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	relatedObservation	observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained. The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	VolcanicActivity See Clause 5.7.6.6.1.7.1.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	TropicalCyclone See Clause 5.7.6.6.1.6.1.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	AviationWx See Clause 5.7.6.6.1.1.2.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	AerodromeWx See Clause 5.7.6.6.1.1.1.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	Turbulence See Clause 5.7.6.6.1.5.5.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	relatedObservation	forecasts tied to be maintained. The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	WindLayer See Clause 5.7.6.6.1.8.1.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	AbstractFeature See Clause 5.7.6.6.3.1.1.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	Convection See Clause 5.7.6.6.1.5.1.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	MountainObscuration See Clause 5.7.6.6.1.5.3.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation. This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	MountainWave See Clause 5.7.6.6.1.5.4.	[0..1]
Association	relatedObservation	The base observation allows for a forecast to be linked to its corresponding base observation.	WindShear See Clause	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
		This is in support of existing data models where the observations and forecasts are bundled together for convenience. It allows for the linkage between an observation and its associated forecasts tied to be maintained.	5.7.6.6.1.5.6.	

5.7.6.6.2.8. Elements Defined in the WX_Measures Model

The following elements are defined in this model.

5.7.6.6.2.8.1. AirTemperature «FeatureType» (Class)

Air temperature.

Parent Class: Temperature (See Clause 5.7.6.6.2.1.9.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX_Measures\AirTemperature.*

5.7.6.6.2.8.2. CloudHeight «FeatureType» (Class)

Cloud height.

Parent Class: VerticalDistance (See Clause 5.7.6.6.2.8.6.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX_Measures\CloudHeight.*

5.7.6.6.2.8.3. Depth «FeatureType» (Class)

Depth.

Parent Class: VerticalDistance (See Clause 5.7.6.6.2.8.6.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX_Measures\Depth.*

5.7.6.6.2.8.4. HorizontalDistance «FeatureType» (Class)

Horizontal distance.

Parent Class: Distance (See Clause 5.7.6.6.2.1.4.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX_Measures\HorizontalDistance.*

5.7.6.6.2.8.5. HorizontalVisibilityDistance «FeatureType» (Class)

Restricts HorizontalDistance to an integer value greater than or equal to zero.

Parent Class: HorizontalDistance (See Clause 5.7.6.6.2.8.4.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\HorizontalVisibilityDistance.*

5.7.6.6.2.8.6. VerticalDistance «FeatureType» (Class)

Vertical distance.

Parent Class: Distance (See Clause 5.7.6.6.2.1.4.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\VerticalDistance.*

5.7.6.6.2.8.7. VerticalVisibilityDistance «FeatureType» (Class)

Restricts VerticalDistance to an integer value greater than or equal to zero.

Parent Class: VerticalDistance (See Clause 5.7.6.6.2.8.6.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\VerticalVisibilityDistance.*

5.7.6.6.2.8.8. WindDirection «Type» (Class)

WindDirection is a Bearing expressed as an integer.

Parent Class: Bearing (See Clause 5.7.6.6.2.1.3.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\WindDirection.*

5.7.6.6.2.8.9. WindGust «FeatureType» (Class)

Wind gust.

Parent Class: Speed (See Clause 5.7.6.6.2.1.8.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_Measures\WindGust.*

5.7.6.6.2.8.10. WindSpeed «FeatureType» (Class)

Restriction of Speed to an integer value greater than or equal to one.

Parent Class: Speed (See Clause 5.7.6.6.2.1.8.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_Measures\WindSpeed.*

5.7.6.6.2.9. Elements Defined in the WX_MotionVector Model

The following elements are defined in this model:

5.7.6.6.2.9.1. MotionVector «FeatureType» (Class)
Motion vector feature.

Parent Class: AbstractWxFeature (See Clause 5.7.6.6.2.2.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_MotionVector\MotionVector.*

Table 5.404 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
position	The position of the motion vector.	DirectPosition See Clause 5.7.6.6.4.4.3.	[1]

This element has the following connections to other elements:

Table 5.405 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	direction	Direction of motion vector.	Angle See Clause 5.7.6.6.2.1.2.	[1]
Association	position	Position of feature.	DirectPosition See Clause 5.7.6.6.4.4.3.	[1]
Association	speed	Speed of movement.	Speed See Clause 5.7.6.6.2.1.8.	[1]

5.7.6.6.2.10. Elements Defined in the WX_Observation Model

The following elements are defined in this model:

5.7.6.6.2.10.1. Observation «FeatureType» (Class)
Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

Parent Class: Observation (See Clause 5.7.6.6.5.1.2.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_Observation\Observation*.

Table 5.406 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
validTime	Time period over which observation result is valid (can incorporate "expiration time").	UTCDateTime See Clause 6.6.1.11.	[1]

5.7.6.6.2.11. Elements Defined in the WX_PhenomBase Model

The following elements are defined in this model.

5.7.6.6.2.11.1. AbstractPhenomenon «FeatureType» (Abstract Class)

Abstract class intended to act as head of substitution group for all weather phenomena classes.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_PhenomBase\AbstractPhenomenon*.

Table 5.407 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
coverage	Percentage of coverage of the phenomenon.	PercentageRange See Clause 5.7.6.6.2.5.1.	[1]

This element has the following connections to other elements:

Table 5.408 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	extentOf	Extent of the phenomenon.	ExtentOf See Clause 5.7.6.6.2.2.2.	[0..1]
Association	intensity	Weather intensity.	WeatherIntensity See Clause 5.7.6.6.2.5.12.	[0..1]
Association	movement	Movement.	MovementDescription See Clause 5.7.6.6.2.11.2.	[0..*]

5.7.6.6.2.11.2. MovementDescription «FeatureType» (Class)

Movement description feature.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_PhenomBase\MovementDescription.*

Table 5.409 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
directionTowards	Direction feature is moving towards.	Bearing See Clause 5.7.6.6.2.1.3.	[1]
speed	Speed of feature motion.	Speed See Clause 5.7.6.6.2.1.8.	[1]
isStationary	True if movement description is stationary.	Boolean See Clause 6.6.3.3.	[1]

This element has the following connections to other elements:

Table 5.410 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	compassDirection	Compass direction.	Compass16 See Clause 5.7.6.6.2.5.4.	[0..1]

5.7.6.6.2.11.3. OtherPhenomenon «FeatureType» (Class)

Other phenomenon feature.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_PhenomBase\OtherPhenomenon.*

This element has the following connections to other elements:

Table 5.411 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	phenomenonType	Phenomenon type.	OtherPhenomenonTypes See Clause 5.7.6.6.2.11.7.	[0..1]

5.7.6.6.2.11.4. VerticallyBoundedPhenomenon «FeatureType» (Abstract Class)

A weather phenomenon that is vertically bounded with a base and/or top.

Parent Class: AbstractPhenomenon (See Clause 5.7.6.6.2.11.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\VerticallyBoundedPhenomenon.*

Table 5.412 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
base	Base vertical distance of phenomenon.	VerticalDistance See Clause 5.7.6.6.2.8.6.	[1]
top	Top distance of the phenomenon.	VerticalDistance See Clause 5.7.6.6.2.8.6.	[1]

5.7.6.6.2.11.5. WxCondition «Type» (Class)

Weather condition description.

Parent Class: AbstractFeature (See Clause 5.7.6.6.3.1.1.)

UML Element Location: *Model\Model Elements from External Sources\WXXM\WX\WX_PhenomBase\WxCondition.*

Table 5.413 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
weatherModifier	Allows for intensity or proximity attribution of condition.	WeatherModifier See Clause 5.7.6.6.2.5.2.	[1]
wxPhenomenon	Phenomenon associated with condition.	AbstractPhenomenon See Clause 5.7.6.6.2.11.1.	[1]

This element has the following connections to other elements:

Table 5.414 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	weatherDescriptor	Descriptor of weather condition.	WeatherDescriptor See Clause 5.7.6.6.2.5.11.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	wxCODE	Weather codes.	WxCODE See Clause 5.7.6.6.2.5.3.	[0..1]

5.7.6.6.2.11.6. ObscurationTypes (Enumeration)

Enumeration for obscuration types.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_PhenomBase\ObscurationTypes.*

Table 5.415 - ObscurationTypes Enumerated Values

Enumerated Value	Definition
MIST	Obscuration type is mist.
FOG	Obscuration type is fog.
SMOKE	Obscuration type is smoke.
VOLCANIC ASH	Obscuration type is volcanic ash.
DUST	Obscuration type is dust.
SAND	Obscuration type is sand.
HAZE	Obscuration type is haze.
SPRAY	Obscuration type is spray.
UNIDENTIFIED	Obscuration type is unidentified.

5.7.6.6.2.11.7. OtherPhenomenonTypes (Enumeration)

Enumeration for other types of phenomenon.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_PhenomBase\OtherPhenomenonTypes.*

Table 5.416 - OtherPhenomenonTypes Enumerated Values

Enumerated Value	Definition
DUSTWHIRL	Includes SANDWHIRL (dust devil).
SQUALL	Squall.
FUNNEL CLOUD	Includes TORNADO and WATERSPOUT.
DUSTSTORM	Dust storm.
SANDSTORM	Sand storm.

5.7.6.6.2.11.8. PrecipitationTypes (Enumeration)

Enumeration for precipitation types.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_PhenomBase\PrecipitationTypes.*

Table 5.417 - PrecipitationTypes Enumerated Values

Enumerated Value	Definition
DRIZZLE	Precipitation type is drizzle.
RAIN	Precipitation type is rain.
SNOW	Precipitation type is snow.
SNOW_GRAINS	Precipitation type is snow grains.
ICE_CRYSTALS	Precipitation type is ice crystals.
ICE_PELLETS	Precipitation type is ice pellets.
HAIL	Precipitation type is hail.
SMALL_HAIL	Precipitation type is small hail.
UNKNOWN	Precipitation type is unknown.
UNIDENTIFIED	Precipitation type is unidentified (Sharknado).

5.7.6.6.2.12. Elements Defined in the WX_Process Model

The following elements are defined in this model.

5.7.6.6.2.12.1. ReportProcess «FeatureType» (Class)

The process related to the distribution or collection of weather reports

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\WX\WX_Process\ReportProcess.*

5.7.6.6.3. Elements Defined in the CommonUsagePackages Model

The following elements are defined in this model.

5.7.6.6.3.1. Elements Defined in the ISO 19136 Annex D.3 Model

The following elements are defined in this model.

5.7.6.6.3.1.1. AbstractFeature (gml:AbstractFeature) «FeatureType» (Abstract Class)

Abstract class representing all classes stereotyped <<FeatureType>> i.e., classes that realize GF_FeatureType, representing features of all types from all application schemas. This class may be used as a "wildcard" when it is necessary to refer to a feature of any type.

N.B. Any class stereotyped <<FeatureType>> is implicitly substitutable for AnyFeature.

Implemented by gml:AbstractFeature (gml 3.2) or gml:Feature (earlier versions).

Parent Class: AbstractGML (See Clause 5.7.6.6.3.1.2.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractFeature.*

Table 5.418 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
boundedBy	Bounding envelope of feature.	GM_Envelope See Clause 5.7.6.6.4.4.4.	[0..1]
position	The point location of the feature.	DirectPosition See Clause 5.7.6.6.4.4.3.	[0..1]

5.7.6.6.3.1.2. AbstractGML (gml:AbstractGML) «Type» (Abstract Class)

Abstract class representing all classes stereotyped <<ObjectType>> i.e., classes that represent objects with identity that may be externally referenced. Implemented by gml:AbstractGML (gml 3.2) or gml:_GML (earlier versions).

Parent Class: AbstractObject (See Clause 5.7.6.6.3.1.3.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractGML.*

Table 5.419 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
description	Text description of object.	String See Clause 6.6.3.7.	[0..1]

5.7.6.6.3.1.3. AbstractObject «Type» (Class)

Abstract object class.

Parent Class: FSGIMIdentifiedObject (See Clause 5.2.3.4.)

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\CommonUsagePackages\ISO 19136 Annex D.3\AbstractObject.*

5.7.6.6.4. Elements Defined in the ISO Standard Packages Model

The following elements are defined in this model.

5.7.6.6.4.1. Elements Defined in the ISO 03166 Country Codes Model

The following elements are defined in this model.

5.7.6.6.4.1.1. CountryCode (Enumeration)

ISO 03166 Country Code enumeration.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 03166 Country Codes\CountryCode*.

5.7.6.6.4.2. Elements Defined in the ISO 19139 Metadata Model

The following elements are defined in this model.

5.7.6.6.4.2.1. Elements Defined in the Cultural and linguistic adaptability Model

The following elements are defined in this model.

5.7.6.6.4.2.2. PT_Locale (Class)

A class describing the cultural locale for a location including language, country, and character set.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19139 Metadata\Cultural and linguistic adaptability\PT_Locale*.

Table 5.420 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
languageCode	The language code of the locale.	LanguageCode See Clause 5.7.6.6.4.3.1.	[1]
country	The country code of the locale.	CountryCode See Clause 5.7.6.6.4.1.1.	[1]
characterSetCode	The character set code of the locale.	MD_CharacterSetCode See Clause 5.7.6.6.4.6.19.	[1]

5.7.6.6.4.3. Elements Defined in the ISO 00639 Human Language Model

The following elements are defined in this model.

5.7.6.6.4.3.1. LanguageCode (Enumeration)

ISO standard enumeration for language codes.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 00639 Human Language\LanguageCode*.

Table 5.421 – LanguageCode Enumerated Values

Enumerated Value	Definition
Afrikaans	Afrikaans.
Albanian	Albanian.
Arabic	Arabic.
Basque	Basque.
Belarusian	Belarusian.
Bulgarian	Bulgarian.
Catalan	Catalan.
Chinese	Chinese.
Croatian	Croatian.
Czech	Czech.
Danish	Danish.
Dutch	Dutch.
English	English.
Estonian	Estonian.
Faeroese	Faeroese.
Finnish	Finnish.
French	French.
FrenchCanadian	French(Canadian).
German	German.
Greek	Greek.
Hawaiian	Hawaiian.
Hebrew	Hebrew.
Hungarian	Hungarian.
Icelandic	Icelandic.
Indonesian	Indonesian.
Italian	Italian.
Japanese	Japanese.
Korean	Korean.
Latvian	Latvian.
Lithuanian	Lithuanian.
Malaysian	Malaysian.
Norwegian	Norwegian.
Polish	Polish.
Portuguese	Portuguese.
Romanian	Romanian.
Russian	Russian.
Serbian	Serbian.

Enumerated Value	Definition
Slovak	Slovak.
Slovenian	Slovenian.
Spanish	Spanish.
Swahili	Swahili.
Swedish	Swedish.
Turkish	Turkish.
Ukranian	Ukranian.

5.7.6.6.4.4. Elements Defined in the ISO 19107 Spatial Schema Model

The following elements are defined in this model.

5.7.6.6.4.4.1. Elements Defined in the Geometry Model

The following elements are defined in this model.

5.7.6.6.4.4.2. Elements Defined in the Coordinate geometry Model

The following elements are defined in this model.

5.7.6.6.4.4.3. DirectPosition «DataType» (Class)

DirectPosition object data types (Figure 14) hold the coordinates for a position within some coordinate reference system. The coordinate reference system is described in ISO 19111. Since DirectPositions, as data types, will often be included in larger objects (such as GM_Objects) that have references to ISO 19111::SC_CRS, the DirectPosition::coordinateReferenceSystem may be left NULL if this particular DirectPosition is included in a larger object with such a reference to a SC_CRS. In this case, the DirectPosition::coordinateReferenceSystem is implicitly assumed to take on the value of the containing object's SC_CRS.

UML Element Location: *Model\Model Elements from External Sources\WXXMISO Standard Packages\ISO 19107 Spatial Schema\Geometry\Coordinate geometry\DirectPosition.*

Table 5.422 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
coordinate	Array of coordinates for the direct position in the Coordinate Reference System units.	Real See Clause 6.6.3.6.	[1..*]
dimension (Derived attribute)	Number of coordinates.	Integer See Clause 6.6.3.5.	[1]

This element has the following connections to other elements:

Table 5.423 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	CRS	The coordinate system for this position.	SC_CRS See Clause 5.7.6.4.5.2.	[0..1]

5.7.6.6.4.4.4. GM_Envelope «DataType» (Class)

GM_Envelope is often referred to as a minimum bounding box or rectangle. Regardless of dimension, a GM_Envelope can be represented without ambiguity as two direct positions (coordinate points). To encode a GM_Envelope, it is sufficient to encode these two points. This is consistent with all of the data types in this standard, their state is represented by their publicly accessible attributes.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19107 Spatial Schema\Geometry\Coordinate geometry\GM_Envelope.*

Table 5.424 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
lowerCorner	A coordinate consisting of all minimal values of the ordinates of all points within the GM_Envelope. The "lowerCorner" of a GM_Envelope is a coordinate position consisting of all the minimal ordinates for each dimension for all points within the GM_Envelope. GM_Envelope::lowerCorner : DirectPosition	DirectPosition See Clause 5.7.6.6.4.4.3.	[1]
upperCorner	A coordinate consisting of all maximal values of the ordinates of all points within the GM_Envelope. The "upperCorner" of a GM_Envelope is a coordinate position consisting of all the maximal ordinates for each dimension for all points within the GM_Envelope. GM_Envelope::upperCorner : DirectPosition	DirectPosition See Clause 5.7.6.6.4.4.3.	[1]

5.7.6.6.4.5. Elements Defined in the ISO 19111 Spatial Referencing by Coordinates Model

The following elements are defined in this model.

5.7.6.6.4.5.1. Elements Defined in the Coordinate Reference Systems Model

The following elements are defined in this model.

5.7.6.6.4.5.2. SC_CRS «Type» (Abstract Class)

Coordinate reference system which is usually single but may be compound.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19111 Spatial Referencing by Coordinates\Coordinate Reference Systems\SC_CRS.*

5.7.6.6.4.6. Elements Defined in the ISO 19115 Metadata Model

The following elements are defined in this model.

5.7.6.6.4.6.1. Elements Defined in the Citation and responsible party information Model

The following elements are defined in this model.

5.7.6.6.4.6.2. CI_Address «DataType» (Class)

Location of the responsible individual or organisation.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Address.*

Table 5.425 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
administrativeArea[0..1]	State, province of the physical address.	String See Clause 6.6.3.7.	[1]
city[0..1]	City of the physical address.	String See Clause 6.6.3.7.	[1]
country[0..1]	Country of the physical address.	String See Clause 6.6.3.7.	[1]
deliveryPoint[0..*]	Address line for the physical address (Street name, box number, suite).	String See Clause 6.6.3.7.	[1]
electronicMailAddress[0..*]	Address of the electronic mailbox of the responsible organisation or individual.	String See Clause 6.6.3.7.	[1]
postalCode[0..1]	ZIP or other postal code.	String See Clause 6.6.3.7.	[1]

5.7.6.6.4.6.3. CI_Citation «DataType» (Class)

Standardized resource reference.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Citation.*

Table 5.426 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
alternateTitle[0..*]	Short name or other language name by which the cited information is known. -Example: "Digital Chart of the World" or "DCW".	String See Clause 6.6.3.7.	[1]
citedResponsibleParty[0..*]	Name and position information for an individual or organisation that is responsible for the resource.	CI_ResponsibleParty See Clause 6.7.6.6.4.6.6.	[1]
collectiveTitle[0..1]	Common title with holdings note.	String See Clause 6.6.3.7.	[1]
date[1..*]	Reference date for the cited information.	UTCDateTime See Clause 6.6.1.11.	[1]
edition[0..1]	Version of the dataset.	String See Clause 6.6.3.7.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
editionDate[0..1]	Date of the edition.	UTCDateTime See Clause 6.6.1.11.	[1]
identifier[0..*]	value uniquely identifying an object within a namespace.	MD_Identifier See Clause 5.7.6.4.6.25.	[1]
ISBN[0..1]	International Standard Book Number.	String See Clause 6.6.3.7.	[1]
ISSN[0..1]	International Standard Serial Number.	String See Clause 6.6.3.7.	[1]
otherCitationDetails[0..1]	Other information required to complete the citation.	String See Clause 6.6.3.7.	[1]
presentationForm[0..*]	Mode in which the data is represented.	CI_PresentationFormCode See Clause 5.7.6.4.6.10.	[1]
series[0..1]	Name of the series of which the dataset is a part.	CI_Series See Clause 5.7.6.4.6.7.	[1]
title	Name by which the cited information is known.	String See Clause 6.6.3.7.	[1]

5.7.6.6.4.6.4. CI_Contact «DataType» (Class)

Information required enabling contact with the responsible person and/or organisation

UML Element Location: Model\UML Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Contact.

Table 5.427 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
address[0..1]	Physical and email address at which the organisation or individual may be contacted.	CI_Address See Clause 5.7.6.4.6.2.	[1]
contactInstructions[0..1]	Supplemental instructions on how or when to contact the individual or organisation.	String See Clause 6.6.3.7.	[1]
hoursOfService[0..1]	Time period (including time zone) when individuals can contact the organisation or individual.	String See Clause 6.6.3.7.	[1]
onlineResource[0..1]	Online information that can be used to contact the individual or organisation.	CI_OnlineResource See Clause 5.7.6.4.6.5.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
phone[0..1]	Telephone numbers at which the organisation or individual may be contacted.	CI_Telephone See Clause 5.7.6.4.6.8.	[1]

5.7.6.4.6.5. CI_OnlineResource «DataType» (Class)

Information about online sources from which the dataset, specification, or community profile name and extended metadata elements can be obtained.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_OnlineResource.*

Table 5.428 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
applicationProfile[0..1]	Name of an application profile that can be used with the resource.	String See Clause 6.6.3.7.	[1]
description[0..1]	Description of what the resource is/does.	String See Clause 6.6.3.7.	[1]
function[0..1]	Function performed by the resource.	CI_OnLineFunctionCode See Clause 5.7.6.4.6.9.	[1]
linkage	Method, source, or location for online access. Example: a Uniform Resource Locator (URL) such as http://www.gii.getty.edu/ign_browser/	AnyURI See Clause 6.6.3.2.	[1]
name[0..1]	Name of the resource.	String See Clause 6.6.3.7.	[1]
protocol[0..1]	Connection protocol to be used.	String See Clause 6.6.3.7.	[1]

5.7.6.4.6.6. CI_ResponsibleParty «DataType» (Class)

Identification of, and means of communication with, person(s) and organisations associated with the dataset.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_ResponsibleParty.*

Table 5.429 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
contactInfo[0..1]	Address of the responsible party.	CI_Contact See Clause 5.7.6.4.6.4.	[1]
individualName[0..1]	Name of the responsible person- SURNAME, given name, title separated by a delimiter.	String See Clause 6.6.3.7.	[1]
organisationName[0..1]	Name of the responsible organisation.	String See Clause 6.6.3.7.	[1]
positionName[0..1]	Role or position of the responsible person.	String See Clause 6.6.3.7.	[1]
role	Function performed by the responsible party.	CI_RoleCode See Clause 5.7.6.4.6.11.	[1]

5.7.6.6.4.6.7. CI_Series «DataType» (Class)

Information about the series, or aggregate resource, to which a resource belongs.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Series.*

Table 5.430 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
issueIdentification[0..1]	Information identifying the issue of the series.	String See Clause 6.6.3.7.	[1]
name[0..1]	Name of the series, or aggregate resource, of which the resource is a part.	String See Clause 6.6.3.7.	[1]
page[0..1]	Details on which pages of the publication the article was published.	String See Clause 6.6.3.7.	[1]

5.7.6.6.4.6.8. CI_Telephone «DataType» (Class)

Telephone numbers for contacting the responsible individual or organisation.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_Telephone.*

Table 5.431 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
facsimile[0..*]	Telephone number of a facsimile machine for the responsible organisation or individual.	String See Clause 6.6.3.7.	[1]
voice[0..*]	Telephone number by which individuals can speak to the responsible organisation or individual.	String See Clause 6.6.3.7.	[1]

5.7.6.6.4.6.9. CI_OnLineFunctionCode (Enumeration)

Function performed by the resource

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_OnLineFunctionCode.*

Table 5.432 - CI_OnLineFunctionCode Enumerated Values

Enumerated Value	Definition
download	Online instructions provide the information necessary to acquire data
information	Online instructions provide more information about the data
offlineAccess	Online instructions provide the ability to transfer data from one storage device or system to another
order	Online instructions provide the ability to acquire data
search	Online instructions provide the ability to seek out information about a dataset

5.7.6.6.4.6.10. CI_PresentationFormCode (Enumeration)

Mode in which the data is represented

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_PresentationFormCode.*

Table 5.433 - CI_PresentationFormCode Enumerated Values

Enumerated Value	Definition
documentDigital	Piece of written or printed matter that provides a record or evidence of events, an agreement, ownership, identification, etc..
documentHardcopy	Representation of a map which is printed on paper, photographic material, or other media and can be interpreted directly by the human user
imageDigital	Permanent record of the likeness of any natural or man-made features, objects, and activities reproduced on photographic materials. This image can be acquired through the sensing of visual or any other segment of the electromagnetic spectrum by sensors, such as thermal infrared, and high resolution radar.
imageHardcopy	Hard-copy of image.
mapDigital	Digital map.
mapHardcopy	Hard copy of map.
modelDigital	Representation in three dimensions of geospatial data.
modelHardcopy	Hard copy of model.
profileDigital	Vertical cross-section of geospatial data
profileHardcopy	Hard copy of profile.
tableDigital	Digital table representation.
tableHardcopy	Hard copy representation of table.
videoDigital	Digital video.
videoHardcopy	Hard copy of digital video.

5.7.6.6.4.6.11. CI_RoleCode (Enumeration)

Function performed by the responsible party.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Citation and responsible party information\CI_RoleCode.*

Table 5.434 - CI_RoleCode Enumerated Values

Enumerated Value	Definition
author	Author role.
custodian	Guardian or keeper responsible for maintaining the data.
distributor	Person or organisation who distributes the data.
originator	Responsible party who created the dataset or metadata
owner	Person who owns the data.
pointOfContact	Responsible party who can be contacted for acquiring knowledge about or acquisition of the data.
principalInvestigator	Key person responsible for gathering information and conducting research.
processor	Responsible party who has processed the data in a manner in which the data has been modified.
publisher	Responsible party who published the data
resourceProvider	Party that supplies the data.
user	Person who uses the data.

5.7.6.6.4.6.12. Elements Defined in the Data quality information Model

The following elements are defined in this model.

5.7.6.6.4.6.13. DQ_Element «Abstract» (Abstract Class)

Used to provide user specific quality information about a dataset's suitability for a particular application.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Data quality information\DQ_Element.*

Table 5.435 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
dateTime[0..*]	%%/% Time of quality evaluation.	UTCTime See Clause 6.6.1.11.	[1]
evaluationMethodDescription[0..1]	Evaluation method description.	String See Clause 6.6.3.7.	[1]
evaluationMethodType[0..1]	Evaluation method type.	DQ_EvaluationMethod TypeCode See Clause 5.7.6.4.6.15.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
evaluationProcedure[0..1]	Evaluation procedure.	CI_Citation See Clause 5.7.6.4.6.3.	[1]
measureDescription[0..1]	Measurement description.	String See Clause 6.6.3.7.	[1]
measureIdentification[0..1]	Measurement identification.	MD_Identifier See Clause 5.7.6.4.6.25.	[1]
nameOfMeasure[0..*]	Name of measurement.	String See Clause 6.6.3.7.	[1]
result[1..2]	Result.	DQ_Result See Clause 5.7.6.4.6.14.	[1]

5.7.6.6.4.6.14. DQ_Result «Abstract» (Abstract Class)
Abstract class to hold result.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Data quality information\DQ_Result.*

5.7.6.6.4.6.15. DQ_EvaluationMethodTypeCode (Enumeration)
Evaluation method type code.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Data quality information\DQ_EvaluationMethodTypeCode.*

Table 5.436 – DQ_EvaluationMethodTypeCode Enumerated Values

Enumerated Value	Definition
directExternal	Evaluation is direct external.
directInternal	Evaluation is direct internal.
indirect	Evaluation is indirect.

5.7.6.6.4.6.16. Elements Defined in the Extent information Model
The following elements are defined in this model.

5.7.6.6.4.6.17. EX_Extent «DataType» (Class)
Information about spatial, vertical, and temporal extent.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Extent information\EX_Extent.*

Table 5.437 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
description[0..1]	Spatial and temporal extent for the referring object.	String See Clause 6.6.3.7.	[1]

5.7.6.6.4.6.18. Elements Defined in the Identification Information Model

The following elements are defined in this model.

5.7.6.6.4.6.19. MD_CharacterSetCode (Enumeration)

Character set code list.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Identification information\MD_CharacterSetCode.*

Table 5.438 – MD_CharacterSetCode Enumerated Values

Enumerated Value	Definition
(reserved for future use)	Reserved for future use.
ucs2	16-bit fixed size Universal Character Set, based on ISO/IEC 10646.
ucs4	32-bit fixed size Universal Character Set, based on ISO/IEC 10646.
utf7	7-bit variable size UCS Transfer Format, based on ISO/IEC 10646.
utf8	8-bit variable size UCS Transfer Format, based on ISO/IEC 10646.
utf16	16-bit variable size UCS Transfer Format, based on ISO/IEC 10646.
8859part1	ISO/IEC 8859-1, Information technology - 8-bit single byte coded graphic character sets - Part 1Å : Latin alphabet No.1.
8859part2	ISO/IEC 8859-2, Information technology - 8-bit single byte coded graphic character sets - Part 2Å : Latin alphabet No.2.
8859part3	ISO/IEC 8859-3, Information technology - 8-bit single byte coded graphic character sets - Part 3Å : Latin alphabet No.3.
8859part4	ISO/IEC 8859-4, Information technology - 8-bit single byte coded graphic character sets - Part 4Å : Latin alphabet No.4.
8859part5	ISO/IEC 8859-5, Information technology - 8-bit single byte coded graphic character sets - Part 5Å : Latin/Cyrillic alphabet.
8859part6	ISO/IEC 8859-6, Information technology - 8-bit single byte coded graphic character sets - Part 6Å : Latin/Arabic alphabet.
8859part7	ISO/IEC 8859-7, Information technology - 8-bit single byte coded graphic character sets - Part 7Å : Latin/Greek alphabet.
8859part8	ISO/IEC 8859-8, Information technology - 8-bit single byte coded graphic character sets - Part 8Å : Latin/Hebrew alphabet.
8859part9	ISO/IEC 8859-9, Information technology - 8-bit single byte coded graphic character sets - Part 9Å : Latin alphabet No.5.
8859part10	ISO/IEC 8859-10, Information technology - 8-bit single byte coded graphic character sets - Part 10Å : Latin alphabet No.6.
8859part11	ISO/IEC 8859-11, Information technology - 8-bit single byte coded graphic character sets - Part 11Å : Latin/Thai alphabet.
8859part13	ISO/IEC 8859-13, Information technology - 8-bit single byte coded graphic character sets - Part 13Å : Latin alphabet No.7.
8859part14	ISO/IEC 8859-14, Information technology - 8-bit single byte coded graphic character sets - Part 14Å : Latin alphabet No.8 (Celtic).
8859part15	ISO/IEC 8859-15, Information technology - 8-bit single byte coded graphic character sets - Part 15Å : Latin alphabet No.9.
8859part16	ISO/IEC 8859-16, Information technology - 8-bit single byte coded graphic character sets - Part 16Å : Latin alphabet No.10.
jis	japanese code set used for electronic transmission.

Enumerated Value	Definition
shiftJIS	japanese code set used on MS-DOS machines.
eucJP	japanese code set used on UNIX based machines.
usAscii	United States ASCII code set (ISO 646 US).
ebedic	IBM mainframe code set.
eucKR	Korean code set.
big5	traditional Chinese code set used in Taiwan, Hong Kong of China and other areas.
GB2312	simplified Chinese code set.

5.7.6.6.4.6.20. Elements Defined in the Maintenance information Model

The following elements are defined in this model.

5.7.6.6.4.6.21. MD_ScopeCode (Enumeration)

Class of information to which the referencing entity applies.

UML Element Location: *Model\Model Elements from External Sources\WXXMMISO Standard Packages\ISO 19115 Metadata\Maintenance information\MD_ScopeCode*.

Table 5.439 - MD_ScopeCode Enumerated Values

Enumerated Value	Definition
attribute	Information applies to the attribute class.
attributeType	Information applies to the attribute type.
collectionHardware	Information applies to the collection hardware class.
collectionSession	Information applies to the collection session.
dataset	Information applies to the attribute class.
dimensionGroup	Information applies to a dimension group.
feature	Information applies to a feature.
featureType	Information applies to a feature type.
fieldSession	Information applies to a field session.
model	Information applies to a model.
nonGeographicDataset	Information applies to non-geographic data.
propertyType	Information applies to a property type.
series	Information applies to the series.
service	Information applies to a service.
software	Information applies to software.
tile	Information applies to a tile.

5.7.6.6.4.6.22. Elements Defined in the Metadata entity set information Model

The following elements are defined in this model.

5.7.6.6.4.6.23. MD_Metadata (Class)

Information about the metadata.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Metadata entity set information\MD_Metadata.*

Table 5.440 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
characterSet[0..1]	Full name of the ISO character coding standard used for the metadata set.	MD_CharacterSetCode See Clause 5.7.6.6.4.6.19.	[1]
contact[1..*]	Party responsible for the metadata information.	CI_ResponsibleParty See Clause 5.7.6.6.4.6.6.	[1]
dataSet[0..1]	Identifiable collection of data.	String See Clause 6.6.3.7.	[1]
dateStamp	Date that the metadata were created or last updated.	UTCDateTime See Clause 6.6.1.11.	[1]
fileIdentifier[0..1]	Unique identifier for this metadata file.	String See Clause 6.6.3.7.	[1]
hierarchyLevel[0..*]	Scope to which the metadata applies (see informative Annex J for more information about metadata heirarchy levels).	MD_ScopeCode See Clause 5.7.6.6.4.6.21.	[1]
hierarchyLevelName[0..*]	Name of the hierarchy level.	String See Clause 6.6.3.7.	[1]
language[0..1]	Language used for documenting metadata.	String See Clause 6.6.3.7.	[1]
locale[0..*]	Locale of the data set.	PT_Locale See Clause 5.7.6.6.4.2.2.	[1]
metadataStandardName[0..1]	Name of the metadata standard used.	String See Clause 6.6.3.7.	[1]
metadataStandardVersion[0..1]	Version of the metadata standard used.	String See Clause 6.6.3.7.	[1]
parentIdentifier[0..1]	Unique identifier of the parent metadata file.	String See Clause 6.6.3.7.	[1]

5.7.6.6.4.6.24. Elements Defined in the Reference system information Model

The following elements are defined in this model.

5.7.6.6.4.6.25. MD_Identifier «Data Type» (Class)

Generic identifier class used to associate a reference code with an identification authority.

UML Element Location: *Model\Model Elements from External Sources\WXXM\ISO Standard Packages\ISO 19115 Metadata\Reference system information\MD_Identifier*.

Table 5.441 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
authority[0..1]	Organization or party responsible for definition and maintenance of the code space or code.	CI_Citation See Clause 5.7.6.6.4.6.3.	[1]
code	Identifier code or name, often from a controlled list or pattern defined by a code space.	String See Clause 6.6.3.7.	[1]

5.7.6.6.5. Elements Defined in the SWE Model

The following elements are defined in this model.

5.7.6.6.5.1. Elements Defined in the OMI_Observation Model

The following elements are defined in this model.

5.7.6.6.5.1.1. Elements Defined in the observation Model

The following elements are defined in this model.

5.7.6.6.5.1.2. Observation «FeatureType» (Class)

Weather observation. This class comprises the measurements and observed phenomena at a point in time and for a given weather area.

UML Element Location: *Model\Model Elements from External Sources\WXXM\SWE\OMI_Observation\observation\Observation*.

Table 5.442 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
metadata	Metadata for the observation.	MD_Metadata See Clause 5.7.6.6.4.6.23.	[1]
parameter	This holds any data primitive of complex structure.	AnyType See Clause 6.6.3.1.	[1]
resultQuality	Quality metadata about the observation.	DQ_Element See Clause 5.7.6.6.4.6.13.	[1]

Attribute Name	Description	Attribute Type	Multiplicity
resultTime	Time when observation procedure was applied. Can be used to store product generation time or issue time.	UTCDateTime See Clause 6.6.1.11.	[1]
samplingTime	Sampling time of observation.	UTCDateTime See Clause 6.6.1.11.	[1]

This element has the following connections to other elements:

Table 5.443 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	featureOfInterest	The featureOfInterest is a feature of any type (ISO 19109, ISO 19101), which is a representation of the observation target, being the real-world object regarding which the observation is made, such as a specimen, station, tract, mountain, pixel, etc. The spatial properties (location) of this feature of interest are typically of most interest for spatial analysis of the observation result. The procedure is the description of a process used to generate the result. It must be suitable for the observed property.	AbstractFeature See Clause 5.7.6.6.3.1.1.	[1]
Association	procedure	NOTE: At this level we do not distinguish between sensor-observations, estimations made by an observer, or algorithms, simulations, computations and complex processing chains. The result contains the value generated by the procedure. The type of the observation result must be consistent with the observed property, and the scale or scope for the value must be consistent with the quantity or category type.	Process See Clause 5.7.6.6.5.3.2.	[0..1]
Association	result	Application profiles may choose to constrain the type of the result.	AbstractObject See Clause 5.7.6.6.3.1.3.	[1]

5.7.6.6.5.1.3. ObservationCollection «FeatureType» (Class)

Collection of arbitrary observations.

UML Element Location: *Model\UML Elements from External Sources\WXXM\SWE\OM1_Observation\observation\ObservationCollection*.
This element has the following connections to other elements:

Table 5.444 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	member	Member of observation collection.	Observation See Clause 5.7.6.6.5.1.2.	[1..*]

5.7.6.6.5.2. Elements Defined in the OM2_Sampling Model

The following elements are defined in this model.

5.7.6.6.5.2.1. Elements Defined in the samplingBase Model

The following elements are defined in this model.

5.7.6.6.5.2.2. SamplingFeature «FeatureType» (Abstract Class)

A "SamplingFeature" is a feature used primarily for taking observations.

Examples: a sounding typically samples a water- or atmospheric-column; a well samples an aquifer; a tissue specimen samples a part of an organism

A SamplingFeature is distinguished from typical domain feature types in that it has a set of [0..*] navigable associations with Observations, given the rolename relatedObservation.

The SamplingFeature has an implicit set of properties corresponding to the observedProperty values of the related observations.

These properties may not be direct members of the sampledFeature, since the proximate observed property may require additional processing to obtain the value of the ultimate domain-feature property.

Sampling features are frequently related to each other, as parts of networks, through sub-sampling, etc. This is supported by the relatedSamplingFeature association with a generic SamplingFeatureRelation association class, which carries a source, target and role.

A common requirement for sampling features is an indication of the SurveyProcedure that provides the surveyDetails related to determination of its location and shape.

Substitutability of SamplingFeature for AnyFeature is implied by the class stereotype <<FeatureType>>.

UML Element Location: *Model\Model Elements from External Sources\WXXM\SWE\OM2_Sampling\samplingBase\SamplingFeature.*

This element has the following connections to other elements:

Table 5.445 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	relatedObservation	A SamplingFeature is distinguished from typical domain feature types in that it has a set of [0..*] navigable associations with Observations, given the rolename relatedObservation. This complements the association role "featureOfInterest" which is constrained to point back from the Observation to the Sampling-Feature. The usual requirement of an Observation feature-of-interest is that its type has a property matching the observed-property on the Observation. In the case of Sampling-features, the topology of the model and	Observation See Clause 5.7.6.6.5.1.2.	[0..*]

Connection Type	Role Name	Description	Role Type	Multiplicity
		navigability of the relatedObservation role means that this requirement is satisfied automatically: a property of the sampling-feature is implied by the observedProperty of a related observation. This effectively provides an unlimited set of "soft-typed" properties on a Sampling Feature.		

5.7.6.6.5.3. Elements Defined in the SensorML Model

The following elements are defined in this model.

5.7.6.6.5.3.1. Elements Defined in the process Model

The following elements are defined in this model.

5.7.6.6.5.3.2. Process «FeatureType» (Abstract Class)

Process feature.

UML Element Location: *Model\UML Model Elements from External Sources\WXXM\SWT\SensorML\process\Process.*

6. COMMON PRIMITIVE TYPES, CLASSES, AND ENUMERATIONS

This package contains the primitive data types, classes, and enumerations that are used throughout the FSGIM.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations*

6.1. Time

This package contains reusable classes to express time-related concepts, such as date, time, duration, etc.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Time*

6.1.1. Time Diagram

This diagram shows the classes and primitive datatypes that make up the Time package.

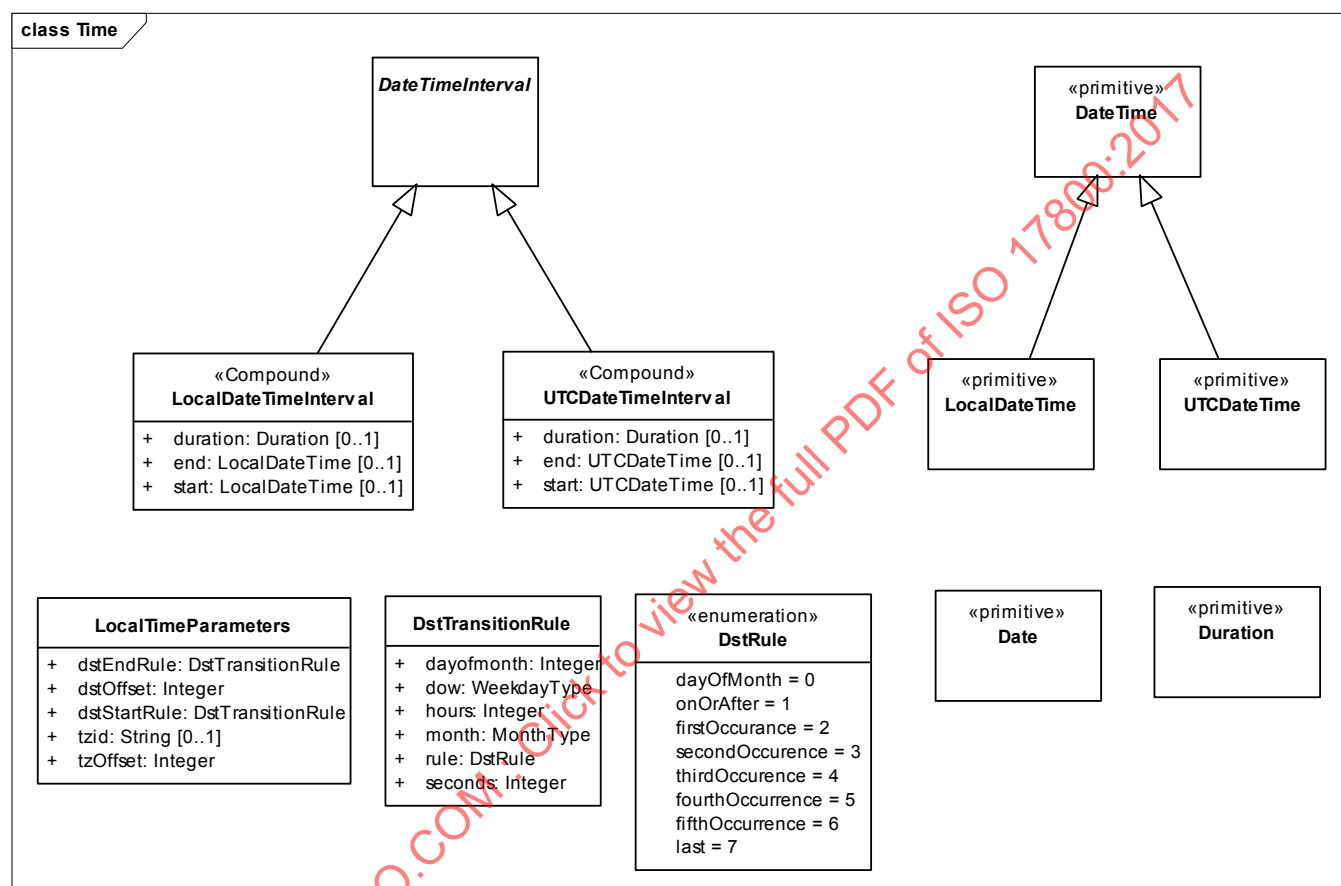


Figure 6.1 - Time

The Time Diagram makes use of the following elements:

DateTimeInterval (Abstract Class)

An abstract class representing an interval of date and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\DateTimeInterval*. See Clause 6.6.1.1 for the complete definition of this element.

DstRule (Enumeration)

Enumeration of the rules for Daylight Savings Time transitions.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\DstRule*. See Clause 6.6.1.6 for the complete definition of this element.

DstTransitionRule (Class)

Rules for DST transition.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\DstTransitionRule*. See Clause 6.6.1.2 for the complete definition of this element.

LocalDateTimeInterval (Class)

An interval of date and time where the start and end times are expressed in local time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\LocalDateTimeInterval*. See Clause 6.6.1.3 for the complete definition of this element.

LocalTimeParameters (Class)

Contains attributes related to the configuration of the time service.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\LocalTimeParameters*. See Clause 6.6.1.5 for the complete definition of this element.

UTCDateTimeInterval (Class)

Interval of date and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\UTCDateTimeInterval*. See Clause 6.6.1.4 for the complete definition of this element.

6.2. Enumerations

This package contains enumerations used in FSGIM. There are two kinds of enumerations -- simple and Code List. A simple enumeration is a set of strings and descriptions that represent a set of values for the enumeration. An attribute that has the type of the enumeration can be expected to "have" a single value from the list. Code Lists are enumerations that have both string and numeric values. In this case the semantic value is a tuple -- string and code.

The representations of enumerations in communications protocols is a local matter and typically highly specific to the standard encoding it. Therefore, FSGIM code lists do not have a data type. The implementing standards choose whether to encode the enumerated value (string or numeric) and what transport data representation to use.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations*

6.2.1. Enumerations Diagram

This diagram depicts the FSGIM standard enumerations.

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

class Enumerations

«enumerati... CurrencyKind	«enumerati... SIScaleCodeType	«enumeration» UnitSymbolKind	«enumeration» QualityOfReading	«enumeration» WeekdayType	«enumeration» MonthType
AED = 784 AFN = 971 ALL = 8 AMD = 51 ANG = 532 AOA = 973 ARS = 32 AUD = 36 AWG = 533 AZN = 944 BAM = 977 BBD = 52 BDT = 50 BGN = 975 BHD = 48 BIF = 108 BMD = 60 BND = 96 BOB = 68 BOV = 984 BRL = 986 BSD = 44 BTN = 64 BWP = 72 BYR = 974 BZD = 84 CAD = 124 CDF = 976 CHF = 756 CLF = 990 CLP = 152 CNY = 156 COP = 170 COU = 970 CRC = 188 CUC = 931 CUP = 192 CVE = 132 CZK = 203 DJF = 262 DKK = 208 DOP = 214 DZD = 12 EEK = 233 EGP = 818 ERN = 232 ETB = 230 EUR = 978 FJD = 242 FKP = 238 GBP = 826 GEL = 981 GHS = 936 GIP = 929 GMD = 270 GNF = 324 GTQ = 320 GYD = 328 HKD = 344 HNL = 340 HRK = 191 HTG = 332 HUF = 348 IDR = 360 ILS = 376 INR = 356 IQD = 366 IRR = 364 ISK = 352 JMD = 388 JOD = 400 JPY = 392 KES = 404 KGS = 417 KHR = 116 KMF = 174 KPW = 408 KRW = 410	y = -24 z = -21 a = -18 f = -15 p = -12 n = -9 micro = -6 m = -3 c = -2 d = -1 none = 0 da = 1 h = 2 k = 3 M = 6 G = 9 T = 12 P = 15 E = 18 Z = 21 Y = 24	m = 2 g = 3 a = 5 degK = 6 mol = 7 cd = 8 s = 27 rad = 10 sr = 11 gy = 21 bq = 22 degC = 23 sv = 24 f = 25 c = 26 h = 28 v = 29 ohm = 30 J = 31 n = 32 hz = 33 lx = 34 lm = 35 wb = 36 t = 37 W = 38 pa = 39 siemens = 53 paA = 155 paG = 140 kat = 158 revPerS = 4 m2 = 41 m3 = 42 mPerS = 43 mPerS2 = 44 m3PerS = 45 mPerM3 = 46 gM = 47 gPerM3 = 48 m2PerS = 49 wPerMK = 50 jPerK = 51 radPerS = 54 VA = 61 var = 63 vS = 66 v2 = 67 aS = 68 a2 = 69 a2S = 70 VAn = 71 Vh = 72 varh = 73 vPerHz = 74 hzPerS = 75 gM2 = 78 wPerS = 81 IPerS = 82 q = 100 qh = 101 ohmM = 102 aPerM = 103 v2H = 104 a2H = 105 aH = 106 wHPPerM3 = 107 timeStamp = 108 wHPPerRev = 115 vArHPPerRev = 116 vAHPPerRev = 117 m3PerH = 125 m3CompensatedPerH = 126 m3UncompensatedPerH = 127 IPerH = 137 IUncompensatedPerH = 139 ICompensatedPerH = 138 a45 = 161	estimated forecast mixed raw validated normalizedForWeather other	sunday = 0 monday = 1 tuesday = 2 wednesday = 3 thursday = 4 friday = 5 saturday = 6	january = 1 february = 2 march = 3 april = 4 may = 5 june = 6 july = 7 august = 8 september = 9 october = 10 november = 11 december = 12

class			
	KPW = 408 KRW = 410 KWD = 414 KYD = 136 KZT = 398 LAK = 418 LBP = 422 LKR = 144 LRD = 430 LSL = 426 LTL = 440 LVL = 428 LYD = 434 MAD = 504 MDL = 498 MGA = 969 MKD = 807 MMK = 104 MNT = 496 MOP = 446 MRO = 478 MUR = 480 MVR = 462 MWK = 454 MXN = 484 MYR = 458 MZN = 943 NAD = 516 NGN = 566 NIO = 558 NOK = 578 NPR = 524 NZD = 554 OMR = 512 PAB = 590 PEN = 604 PGK = 598 PHP = 608 PKR = 586 PLN = 985 PYG = 600 QAR = 634 RON = 946 RSD = 941 RUB = 643 RWF = 646 SAR = 682 SBD = 90 SCR = 690 SDG = 938 SEK = 752 SGD = 702 SHP = 654 SLL = 694 SOS = 706 SRD = 968 STD = 678 SYP = 760 SZL = 748 THB = 764 TJS = 972 TMT = 934 TND = 788 TOP = 776 TRY = 949 TTD = 780 TWD = 901 TZS = 834 UAH = 980 UGX = 800 USD = 840 UYU = 858 UZS = 860 VEF = 937 VND = 704 VUV = 548 WST = 882 XAF = 950 XCD = 951 XOF = 952 XPF = 953 YER = 886 ZAR = 710	tonCompensatedPerH = 139 lCompensatedPerH = 138 q45 = 161 q60 = 162 q45H = 163 q60H = 164 jPerKg = 165 m3Uncompensated = 166 m3Compensated = 167 m1 = 173 m3PerKg = 174 paS = 175 nM = 176 nPerM = 177 radPerS2 = 178 wPerM2 = 179 jPerKgK = 180 jPerM3 = 181 vPerM = 182 cPerM3 = 183 cPerM2 = 184 fPerM = 185 hPerM = 186 jPerMol = 187 jPerMolK = 188 cPerKg = 189 gyPerS = 190 wPerSr = 191 wPerM2Sr = 192 katPerM3 = 193 min = 159 hr = 160 d = 195 deg = 9 angleMin = 196 angleSec = 197 ha = 198 l = 134 tonne = 199 none = 0 cosTheta = 65 bel = 79 status = 109 count = 111 bm = 113 code = 114 meCode = 118 lPerL = 143 gPerG = 144 molPerM3 = 145 molPerMol = 146 molPerKg = 146 mPerM = 148 sPerS = 149 hzPerHz = 150 vPerV = 151 aPerA = 152 wPerVA = 153 rev = 154 wPerW = 168 refractiveIndexN = 170 relativePermeabilityMur = 171 np = 172 eV = 200 da = 201 u = 202 ua = 203 c0 = 204 nuH = 205 nuMe = 206 nuHPerNuMeC02 = 207 auE = 208 auMe = 209 auH = 210 auA0 = 211 auEh = 212 auHPerAuEh = 213 char = 76 charPerSec = 77 money = 80 ft3 = 119 ft3Compensated = 120 ft3Uncompensated = 121 ft3PerH = 122	

class	YER = 886 ZAR = 710 ZMK = 894 ZWL = 932	ft3Uncompensated = 121 ft3PerH = 122 ft3CompensatedPerH = 123 ft3UncompensatedPerH = 124 uSGal = 128 uSGalPerH = 129 impGal = 130 impGalPerH = 129 Btu = 132 BtuPerH = 133 psiA = 141 psiG = 142 lUncompensated = 156 lCompensated = 157 therm = 169 bar = 214 mmHg = 215 angstrom = 216 nmi = 217 barn = 218 kn = 219 ci = 220 r = 221 doseRad = 222 rem = 223 in = 224 ft = 225 rod = 226 fur = 227 mi = 228 ft2 = 229 yd2 = 230 rod2 = 231 acre = 232 mi2 = 233 sectionOfLand = 234 township = 235 yd3 = 237 li = 238 ch = 239 uSLiqPt = 240 uSLiqQt = 241 flDrAp = 242 flOzAp = 243 usDryPt = 244 usDryQt = 245 usPk = 246 usBu = 247 gr = 248 avdpDr = 249 avdpOz = 250 avdpLb = 251 shortCwt = 252 shortTon = 253 longCwt = 254 longTon = 255 dwt = 256 ozT = 257 lbT = 258 sAp = 259 drAp = 260 ozAp = 261 lbAp = 262 mpgImp = 263 mpgUS = 264 mPGeUS = 265 lPer100km = 266 wHPerMi = 267 wHPer100Mi = 268 degF = 279 erg = 269 dyn = 270 p = 271 st = 272 sb = 273 ph = 274 gal = 275 mx = 276 gauss = 277 oe = 278 gPerS = 283

Figure 6.2 - Enumerations

The Enumerations Diagram makes use of the following elements:

CurrencyKind (Enumeration)

Currency is represented in FSGIM as an enumeration. The use of this is optional so it can be omitted when an actual currency is not known. The enumeration is based on ISO 4217 three character codes and the FSIM model contains a snapshot of this list. Any implementation can extend this list as needed but must unambiguously identify how extensions are separate from the ISO standard set.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\CurrencyKind*. See Clause 6.6.2.1 for the complete definition of this element.

MonthType (Enumeration)

An enumerated list of months of the year.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\MonthType*. See Clause 6.6.2.2 for the complete definition of this element.

QualityOfReading (Enumeration)

List of codes indicating the quality of the reading.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\QualityOfReading*. See Clause 6.6.2.5 for the complete definition of this element.

SiScaleCodeType (Enumeration)

Unit multiplier, where the initial value of literal equals the exponent of the multiplier value in base 10. Only the value "none" is normative to the FSGIM.

Note: A value that is representing a percentage can use the UnitSymbolKind of 1 (none) and an SiScaleCodeType of -2 (centi).

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\SiScaleCodeType*. See Clause 6.6.2.3 for the complete definition of this element.

UnitSymbolKind (Enumeration)

This identifies the units of measure based on the NIST Special Publication 811 2008 edition -- Guide for the Use of the International System of Units (SI). Note that these reference document supports SI units as well as common units of measurements in practice including CGS units.

Only VA, W, var, VAh, Wh, varh, Btu, J, therm, BtuPerh, g, gPerM3, gPerS, m, mPerS, WPerM2, rad, sb, pa, and degK are normative for the FSGIM standard. The enumerated values of VA, W, var, VAh, Wh, and varh were changed from their original NAESB values of vA, w, vAr, vAH, wH, and vArH to make them consistent with the standard SI abbreviations for these units. The enumerated values of Btu and BtuPerh were changed from their original NAESB values of btu and btuPerH to make them consistent with the standard abbreviations for British thermal units and hours.

Note: The abbreviation MBtu is often used to represent thousands of Btu's (where M represents the Roman numeral for 1000). The FSGIM *always* uses the SI prefixes listed in the SiScaleCodeType enumeration and not the prefixes based on Roman numerals. This means that 1000 Btu's would be represented in the FSGIM as EnergyThermalQuantity.quantity = 1, EnergyThermalType.itemUnits = Btu, and EnergyThermalType.siScaleCode = k, *not* EnergyThermalType.siScaleCode = M.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\UnitSymbolKind*. See Clause 6.6.2.6 for the complete definition of this element.

WeekdayType (Enumeration)

An enumerated list of day of the week.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\WeekdayType*. See Clause 6.6.2.4 for the complete definition of this element.

6.3. Primitive Data Types

This package contains the primitive data types that are used within the Facility Smart Grid Information Model. These data types do not contain any processor specific or programming language specific restrictions on their range or

resolution. Application specific implementations of the FSGIM may further restrict these data types to allow for application specific constraints, provided that they clearly define these restrictions.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types*

6.4. Measurements

Measurements of power, energy, and emissions are used extensively in the Facility Smart Grid Information Model. Examples include:

- actual measurements, such as those that would come from a meter;
- nominal measurements, such as those that would appear on a product datasheet,
- estimated measurements, such as those that would be created when trying to predict day-ahead demand, or
- derived measurements, such as emissions measurements derived from power or energy measurements.

The various external standards that are used within the FSGIM employ a variety of representations for these measurements. In order to more easily integrate the various external standards, the FSGIM uses the harmonized representation of measurements described in this clause.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements*

6.4.1. High Level Measurements View Diagram

All measurements of power, energy, and emissions within the FSGIM include both a quantity, which represents the magnitude of the thing being measured, and a set of metadata, which relates to that measurement. The metadata includes information such as the units of the measurement, the accuracy and the precision of the measurement.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

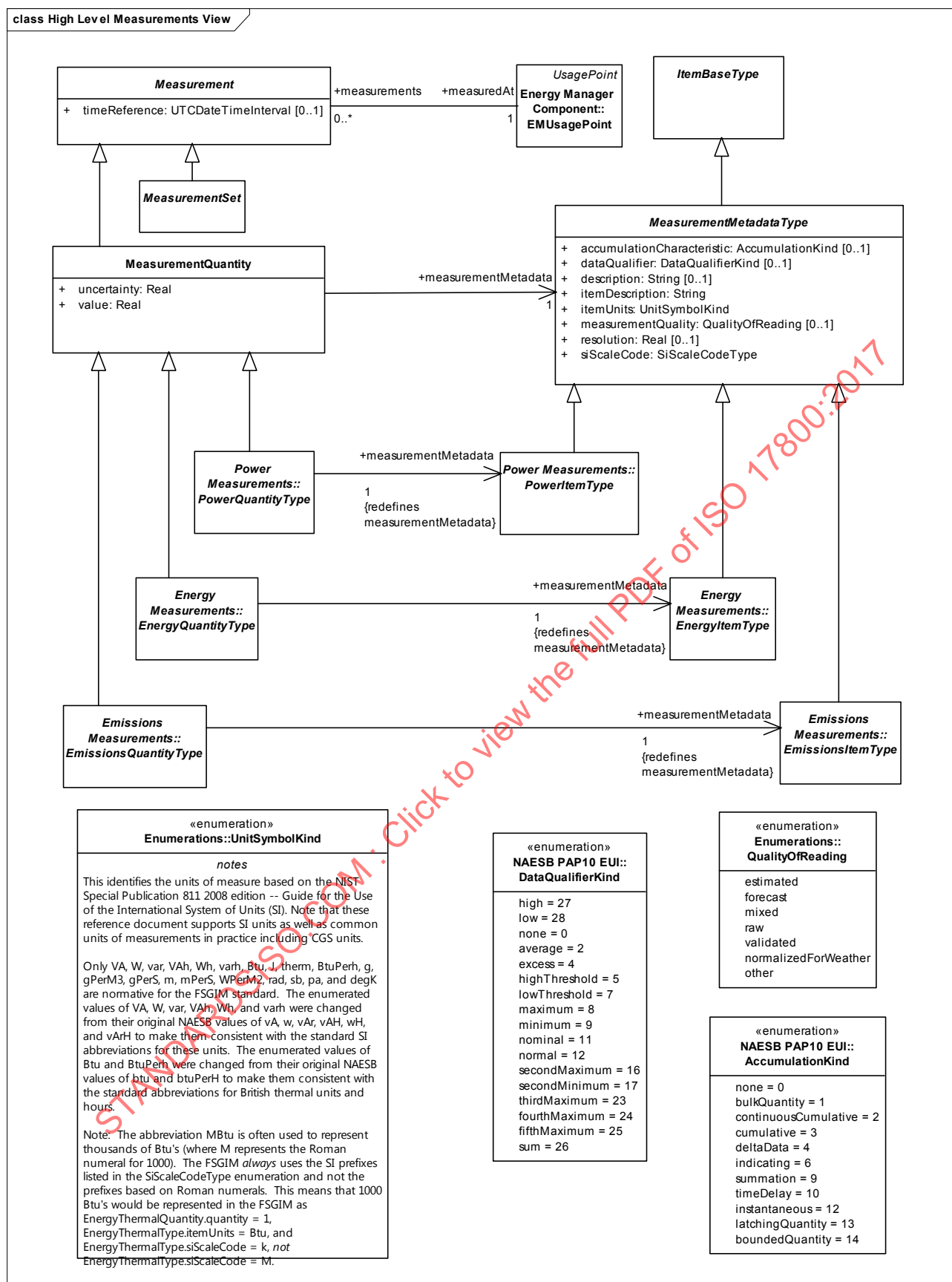


Figure 6.3 - High Level Measurements View

The High Level Measurements View Diagram makes use of the following elements:

AccumulationKind (Enumeration)

This attribute provides metadata about how the measurements described are made. These enumerated values refer to the common present and historic methods by which physical measurements are made in revenue-quality metering instruments and are based on that terminology.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\AccumulationKind. See Clause 5.7.5.2.1.35 for the complete definition of this element.

DataQualifierKind (Enumeration)

This value qualifies the measurement as to what it represents -- for example a minimum or maximum value or a nominal or nameplate value.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\DataQualifierKind. See Clause 5.7.5.2.1.37 for the complete definition of this element.

EmissionsItemType (Abstract Class)

Metadata associated with an emissions measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsItemType*. See Clause 6.6.4.8.4 for the complete definition of this element.

EmissionsQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of emissions measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsQuantityType*. See Clause 6.6.4.8.13 for the complete definition of this element.

EMUsagePoint (Class)

Logical point on a network at which consumption or production is either measured (e.g., metered) or estimated (e.g., unmetered street lights).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMUsagePoint*. See Clause 5.6.7.8 for the complete definition of this element.

EnergyItemType (Abstract Class)

Metadata associated with an energy measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyItemType*. See Clause 6.6.4.7.8 for the complete definition of this element.

EnergyQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyQuantityType*. See Clause 6.6.4.7.10 for the complete definition of this element.

ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType*. See Clause 6.6.4.3 for the complete definition of this element.

Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*. See Clause 6.6.4.1 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and*

Enumerations\Measurements\MeasurementMetadataType. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and*

Enumerations\Measurements\MeasurementQuantity. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*.

See Clause 6.6.4.2 for the complete definition of this element.

PowerItemType (Abstract Class)

Metadata associated with a power measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerItemType. See Clause 6.6.4.6.10 for the complete definition of this element.

PowerQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerQuantityType. See Clause 6.6.4.6.12 for the complete definition of this element.

QualityOfReading (Enumeration)

List of codes indicating the quality of the reading.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\QualityOfReading*.

See Clause 6.6.2.5 for the complete definition of this element.

UnitSymbolKind (Enumeration)

This identifies the units of measure based on the NIST Special Publication 811 2008 edition -- Guide for the Use of the International System of Units (SI). Note that this reference document supports SI units as well as common units of measurements in practice including CGS units.

Only VA, W, var, VAh, Wh, varh, Btu, J, therm, BtuPerh, g, gPerM3, gPerS, m, mPerS, WPerM2, rad, sb, pa, and degK are normative for the FSGIM standard. The enumerated values of VA, W, var, VAh, Wh, and varh were changed from their original NAESB values of vA, w, vAr, vAH, wH, and vArH to make them consistent with the standard SI abbreviations for these units. The enumerated values of Btu and BtuPerh were changed from their original NAESB values of btu and btuPerh to make them consistent with the standard abbreviations for British thermal units and hours.

Note: The abbreviation MBtu is often used to represent thousands of Btu's (where M represents the Roman numeral for 1000). The FSGIM *always* uses the SI prefixes listed in the SiScaleCodeType enumeration and not the prefixes based on Roman numerals. This means that 1000 Btu's would be represented in the FSGIM as EnergyThermalQuantity.quantity = 1, EnergyThermalType.itemUnits = Btu, and EnergyThermalType.siScaleCode = k, *not* EnergyThermalType.siScaleCode = M.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\UnitSymbolKind*.

See Clause 6.6.2.6 for the complete definition of this element.

6.4.2. Measurement Sets Diagram

Some concepts in the Facility Smart Grid Information Model, such as aggregation, apply equally to PowerMeasurementSets, EnergyMeasurementSets, and EmissionsMeasurementSets. In these cases, the

PowerMeasurementSet, EnergyMeasurementSet, and EmissionsMeasurementSet can be abstracted and referred to as a MeasurementSet.

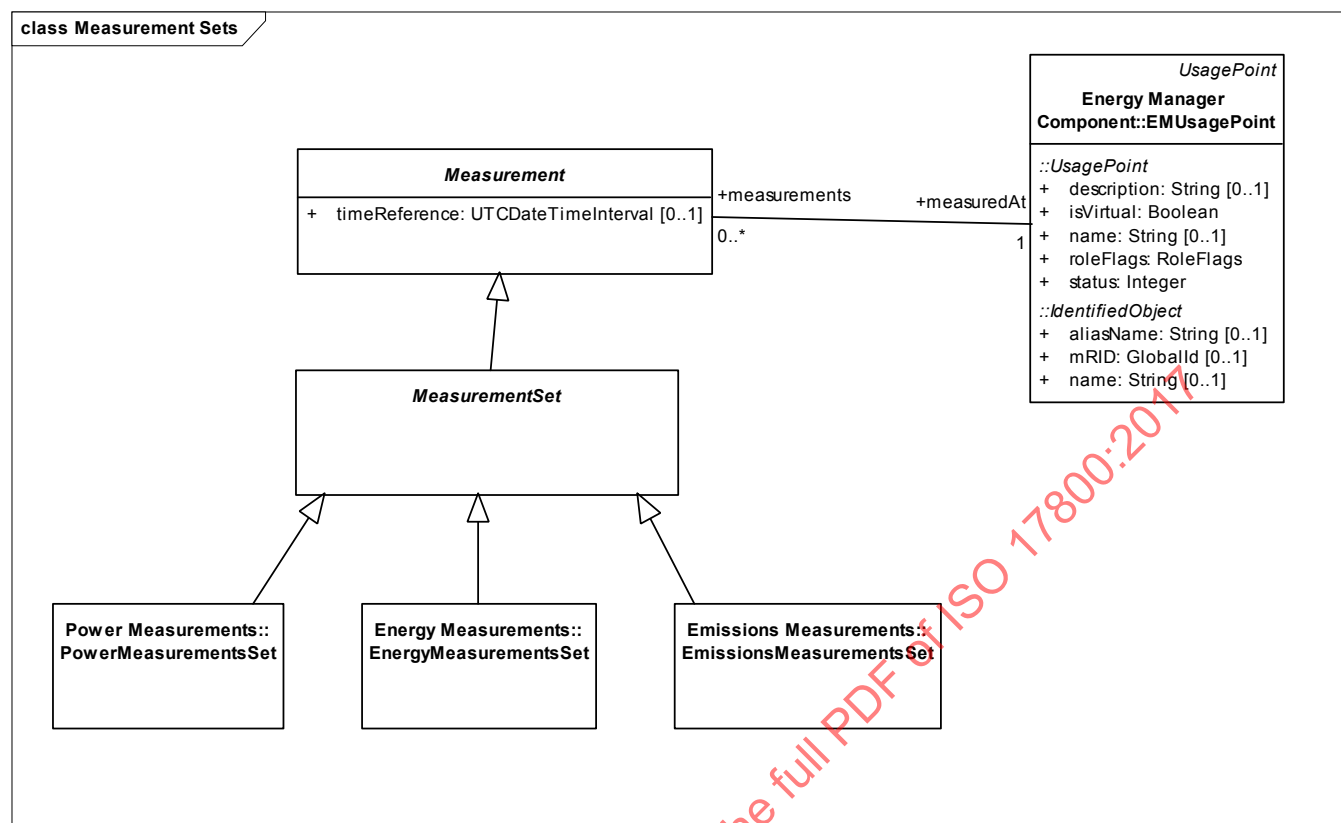


Figure 6.4 - Measurement Sets

The Measurement Sets Diagram makes use of the following elements:

EmissionsMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions quantity measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMeasurementsSet*. See Clause 6.6.4.8.12 for the complete definition of this element.

EMUsagePoint (Class)

Logical point on a network at which consumption or production is either measured (e.g., metered) or estimated (e.g., unmetered street lights).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMUsagePoint*. See Clause 5.6.7.8 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*. See Clause 6.6.4.1 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

6.4.3. Power Measurements

This package contains the classes that are defined for the purpose of modeling power measurements.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements*

6.4.3.1. Power Measurements Diagram

Power, the rate at which energy is transformed into work, is represented in the FSGIM using one of the classes derived from the abstract PowerQuantityType and one of the classes derived from the abstract PowerItemType. The class derived from the PowerQuantityType contains the magnitude of the measured value. The class derived from the PowerItemType contains the metadata associated with the measured value. Multiple magnitudes of the measured value may share the same metadata or they may have separate metadata.

When other portions of the FSGIM need to include a generic power measurement without specifying the exact type of power measurement, they do this by including the PowerQuantityType. Even if a PowerItemType is not shown in the class diagram, its presence is implied due to the multiplicity of one association between the PowerQuantityType class and the PowerItemType class.

When other portions of the FSGIM need to include a specific type of power measurement, they do this by including the specific class that is derived from PowerQuantityType. For example, to specifically indicate that the type of power measurement was a reactive power measurement, a PowerReactiveQuantity class would be included. The presence of a PowerReactiveQuantity would indicate that the corresponding PowerReactiveType metadata class is also present.

As additional types of power measurements that are important to the smart grid are identified in the future, they may be added to the model by creating new classes that are derived from PowerQuantityType and PowerItemType.

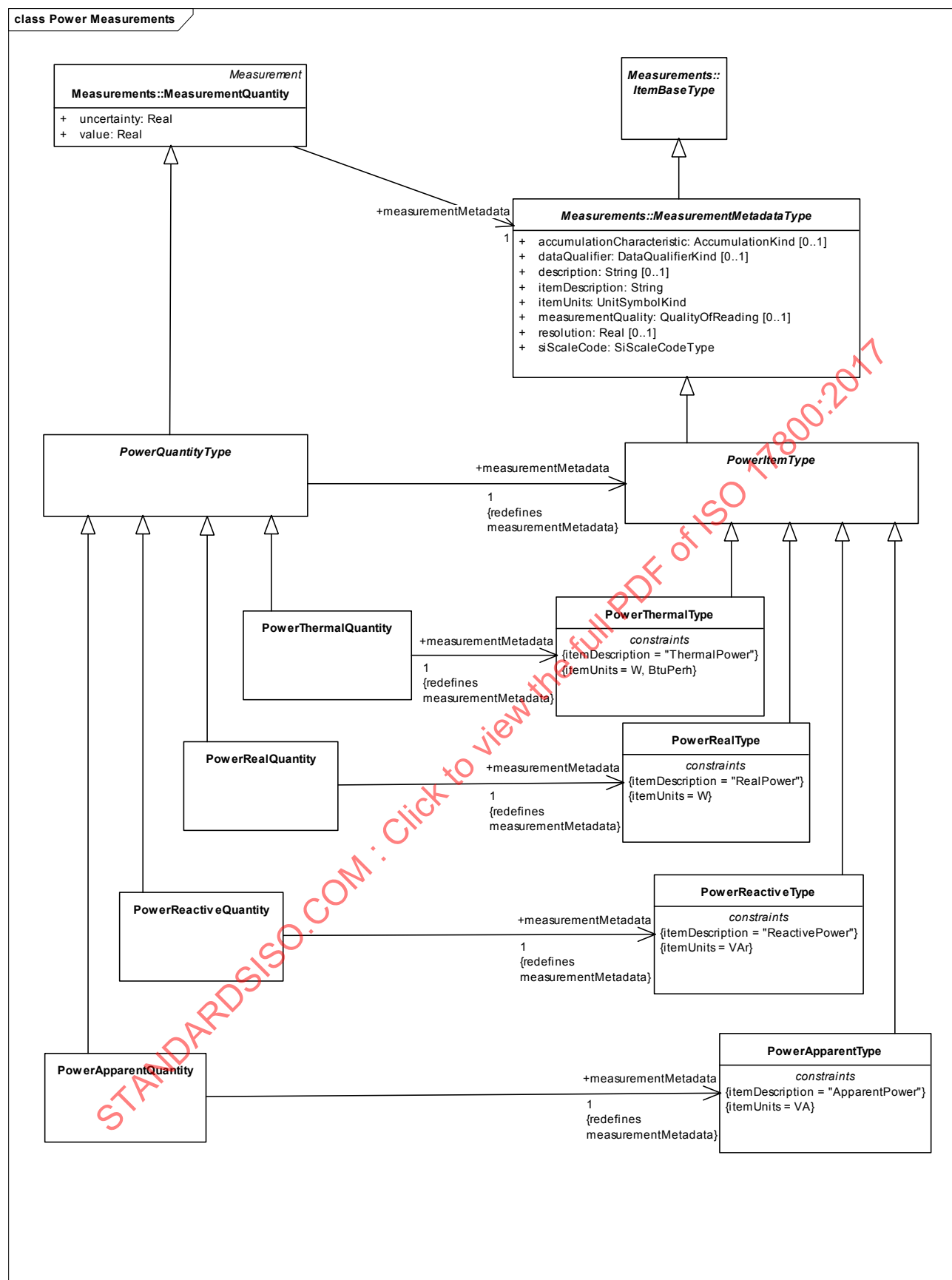


Figure 6.5 - Power Measurements

The Power Measurements Diagram makes use of the following elements:

ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType*. See Clause 6.6.4.3 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

PowerApparentQuantity (Class)

Quantity and pair used to represent apparent power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerApparentQuantity*. See Clause 6.6.4.6.8 for the complete definition of this element.

PowerApparentType (Class)

Metadata associated with measurements of apparent power.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerApparentType*. See Clause 6.6.4.6.9 for the complete definition of this element.

PowerItemType (Abstract Class)

Metadata associated with a power measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerItemType*. See Clause 6.6.4.6.10 for the complete definition of this element.

PowerQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerQuantityType*. See Clause 6.6.4.6.12 for the complete definition of this element.

PowerReactiveQuantity (Class)

Quantity and pair used to represent reactive power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerReactiveQuantity*. See Clause 6.6.4.6.13 for the complete definition of this element.

PowerReactiveType (Class)

Metadata associated with measurements of reactive power.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerReactiveType*. See Clause 6.6.4.6.14 for the complete definition of this element.

PowerRealQuantity (Class)

Quantity and metadata pair used to represent real power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerRealQuantity*. See Clause 6.6.4.6.15 for the complete definition of this element.

PowerRealType (Class)

Metadata associated with measurements of real power.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealType*. See Clause 6.6.4.6.16 for the complete definition of this element.

PowerThermalQuantity (Class)

Quantity/metadata pair used to represent thermal power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerThermalQuantity*. See Clause 6.6.4.6.17 for the complete definition of this element.

PowerThermalType (Class)

Metadata associated with measurements of thermal power.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerThermalType*. See Clause 6.6.4.6.18 for the complete definition of this element.

6.4.3.2. Sets of Power Measurements Diagram

It is sometimes necessary to know more than one type of power measurement for a given location and time. For example, it is often important to know both the real power and the reactive power for a given point at a given time. When this need arises, a PowerMeasurementSet class is used. Each of the subclasses of the abstract PowerQuantityType is optionally included within this PowerMeasurementSet. Which of these optionally included classes is actually included in a particular real life application of the FSGIM is problem domain specific. For example, some residential applications may only need to include the PowerRealQuantity while some industrial applications may need all of the PowerQuantityType derived classes.

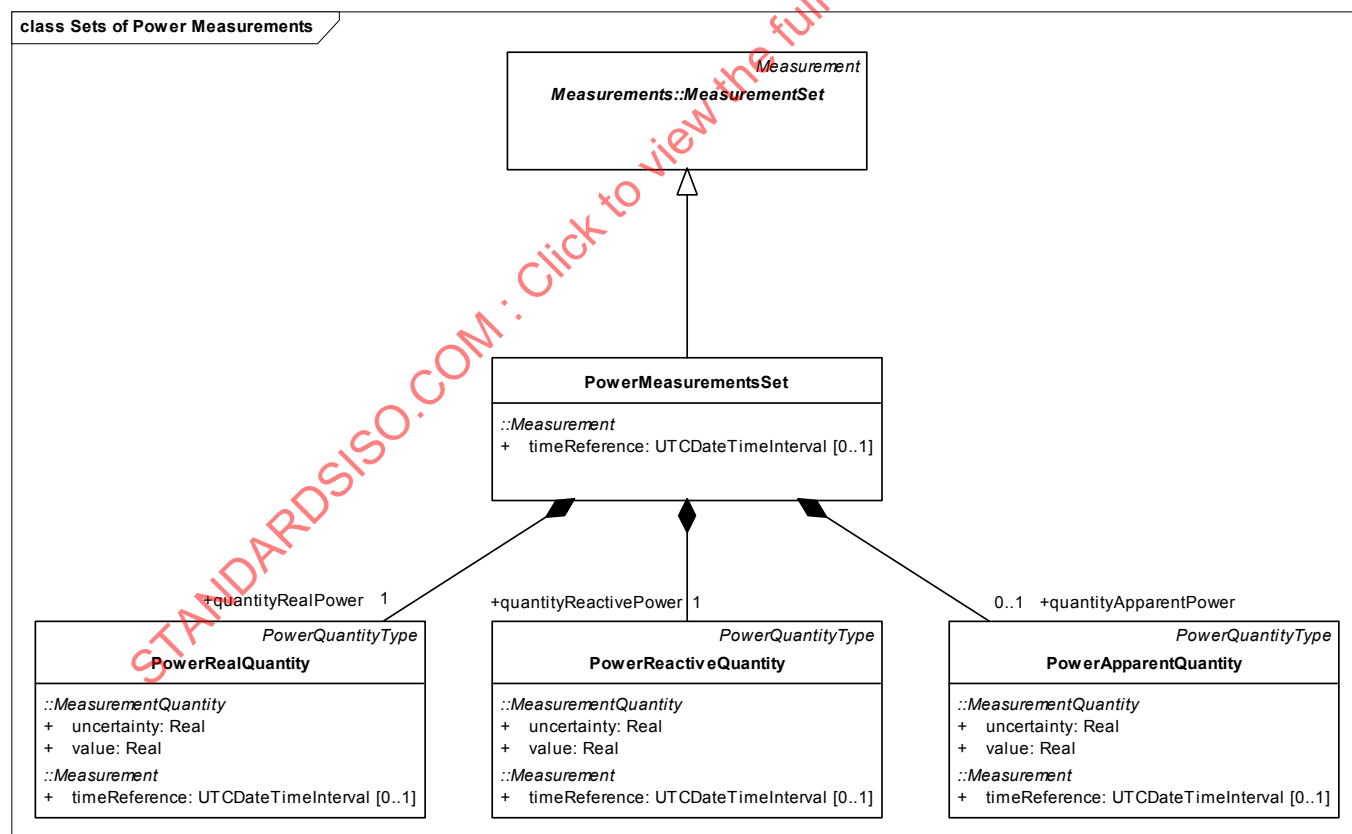


Figure 6.6 - Sets of Power Measurements

The Sets of Power Measurements Diagram makes use of the following elements:

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

PowerApparentQuantity (Class)

Quantity and pair used to represent apparent power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerApparentQuantity*. See Clause 6.6.4.6.8 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

PowerReactiveQuantity (Class)

Quantity and pair used to represent reactive power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveQuantity*. See Clause 6.6.4.6.13 for the complete definition of this element.

PowerRealQuantity (Class)

Quantity and metadata pair used to represent real power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealQuantity*. See Clause 6.6.4.6.15 for the complete definition of this element.

6.4.3.3. Aggregation of Power Measurements Diagram

This diagram illustrates the rules that are generally associated with aggregations of power measurements.

For quantities to be suitable for aggregation, they must have consistent values. For example, measurements that have identical timeReference and MeasurementMetadataType are inherently consistent. When timeReferences are not identical or MeasurementMetadataTypes are not equivalent, however, it is a local matter to decide how to adapt actual measurement values to make them consistent.

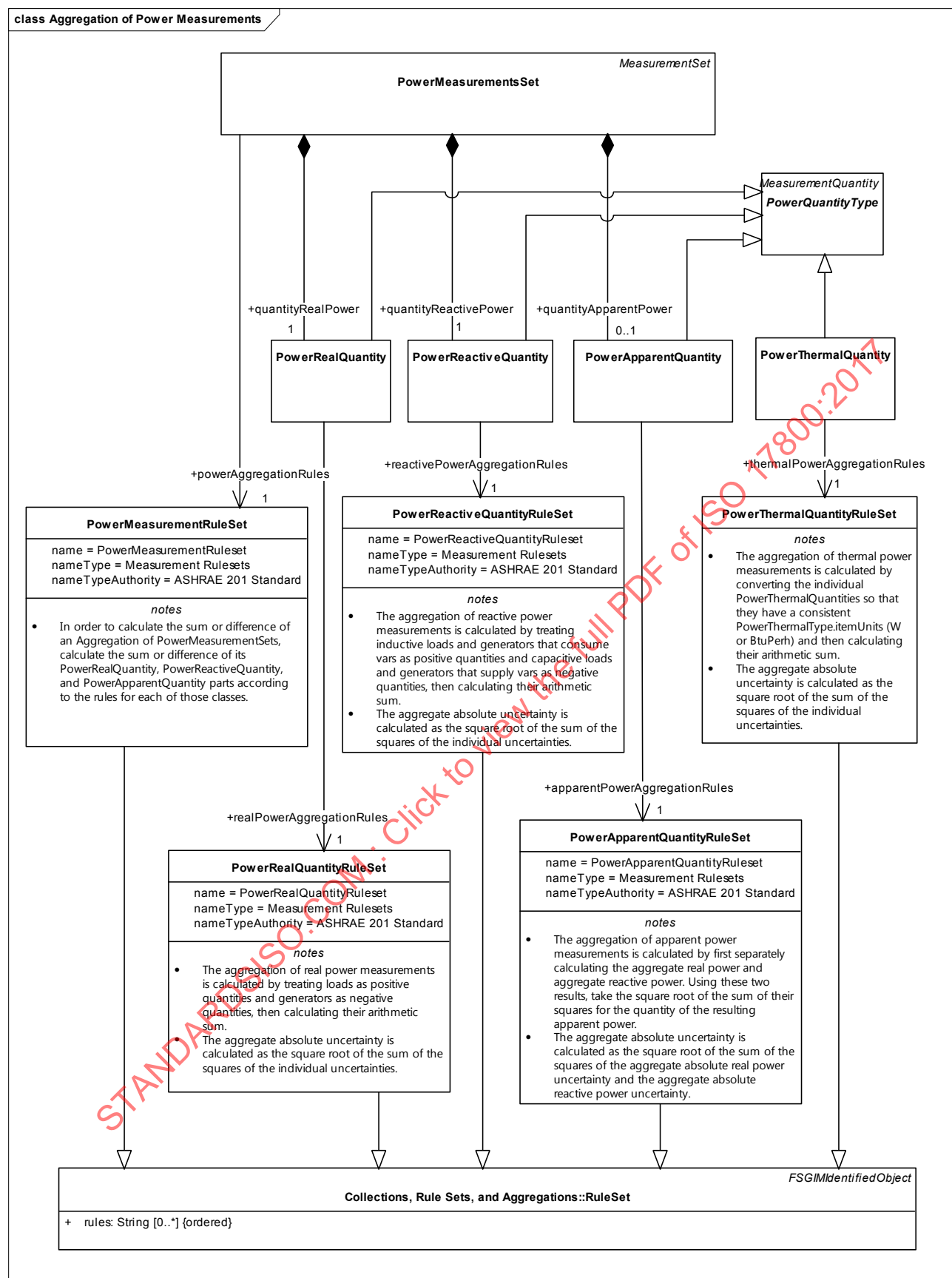


Figure 6.7 - Aggregation of Power Measurements

The Aggregation of Power Measurements Diagram makes use of the following elements:

PowerApparentQuantity (Class)

Quantity and pair used to represent apparent power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerApparentQuantity*. See Clause 6.6.4.6.8 for the complete definition of this element.

PowerApparentQuantityRuleSet (Class)

- The aggregation of apparent power measurements is calculated by first separately calculating the aggregate real power and aggregate reactive power. Using these two results, take the square root of the sum of their squares for the quantity of the resulting apparent power.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the aggregate absolute real power uncertainty and the aggregate absolute reactive power uncertainty.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerApparentQuantityRuleSet*. See Clause 6.6.4.6.1 for the complete definition of this element.

PowerMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of PowerMeasurementSets, calculate the sum or difference of its PowerRealQuantity, PowerReactiveQuantity, and PowerApparentQuantity parts according to the rules for each of those classes.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementRuleSet*. See Clause 6.6.4.6.3 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerMeasurementsSet*. See Clause 6.6.4.6.11 for the complete definition of this element.

PowerQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerQuantityType*. See Clause 6.6.4.6.12 for the complete definition of this element.

PowerReactiveQuantity (Class)

Quantity and pair used to represent reactive power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveQuantity*. See Clause 6.6.4.6.13 for the complete definition of this element.

PowerReactiveQuantityRuleSet (Class)

- The aggregation of reactive power measurements is calculated by treating inductive loads and generators that consume vars as positive quantities and capacitive loads and generators that supply vars as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveQuantityRuleSet*. See Clause 6.6.4.6.5 for the complete definition of this element.

PowerRealQuantity (Class)

Quantity and metadata pair used to represent real power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealQuantity*. See Clause 6.6.4.6.15 for the complete definition of this element.

PowerRealQuantityRuleSet (Class)

- The aggregation of real power measurements is calculated by treating loads as positive quantities and generators as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealQuantityRuleSet*. See Clause 6.6.4.6.6 for the complete definition of this element.

PowerThermalQuantity (Class)

Quantity/metadata pair used to represent thermal power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerThermalQuantity*. See Clause 6.6.4.6.17 for the complete definition of this element.

PowerThermalQuantityRuleSet (Class)

- The aggregation of thermal power measurements is calculated by converting the individual PowerThermalQuantities so that they have a consistent PowerThermalType.itemUnits (W or BtuPerh) and then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerThermalQuantityRuleSet*. See Clause 6.6.4.6.7 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

6.4.4. Energy Measurements

This package contains the classes that are defined for the purpose of modeling energy measurements.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements*

6.4.4.1. Energy Measurements Diagram

Energy, the capacity to do work, is represented in the FSGIM using one of the classes derived from the abstract EnergyQuantityType and one of the classes derived from the abstract EnergyItemType. The class derived from the EnergyQuantityType contains the magnitude of the measured value. The class derived from the EnergyItemType contains the metadata associated with the measured value. Multiple magnitudes of the measured value may share the same metadata or they may have separate metadata.

When other portions of the FSGIM need to include a generic energy measurement without specifying the exact type of energy measurement, they do this by including the EnergyQuantityType. Even if a EnergyItemType is not shown in the class diagram, its presence is implied due to the multiplicity of one association between the EnergyQuantityType class and the EnergyItemType class.

When other portions of the FSGIM need to include a specific type of energy measurement, they do this by including the specific class that is derived from EnergyQuantityType. For example, to specifically indicate that the type of energy measurement was a reactive energy measurement, a EnergyReactiveQuantity class would be included. The presence of a EnergyReactiveQuantity would indicate that the corresponding EnergyReactiveType metadata class is also present. As additional types of energy measurements that are important to the smart grid are identified in the future, they may be added to the model by creating new classes that are derived from EnergyQuantityType and EnergyItemType.

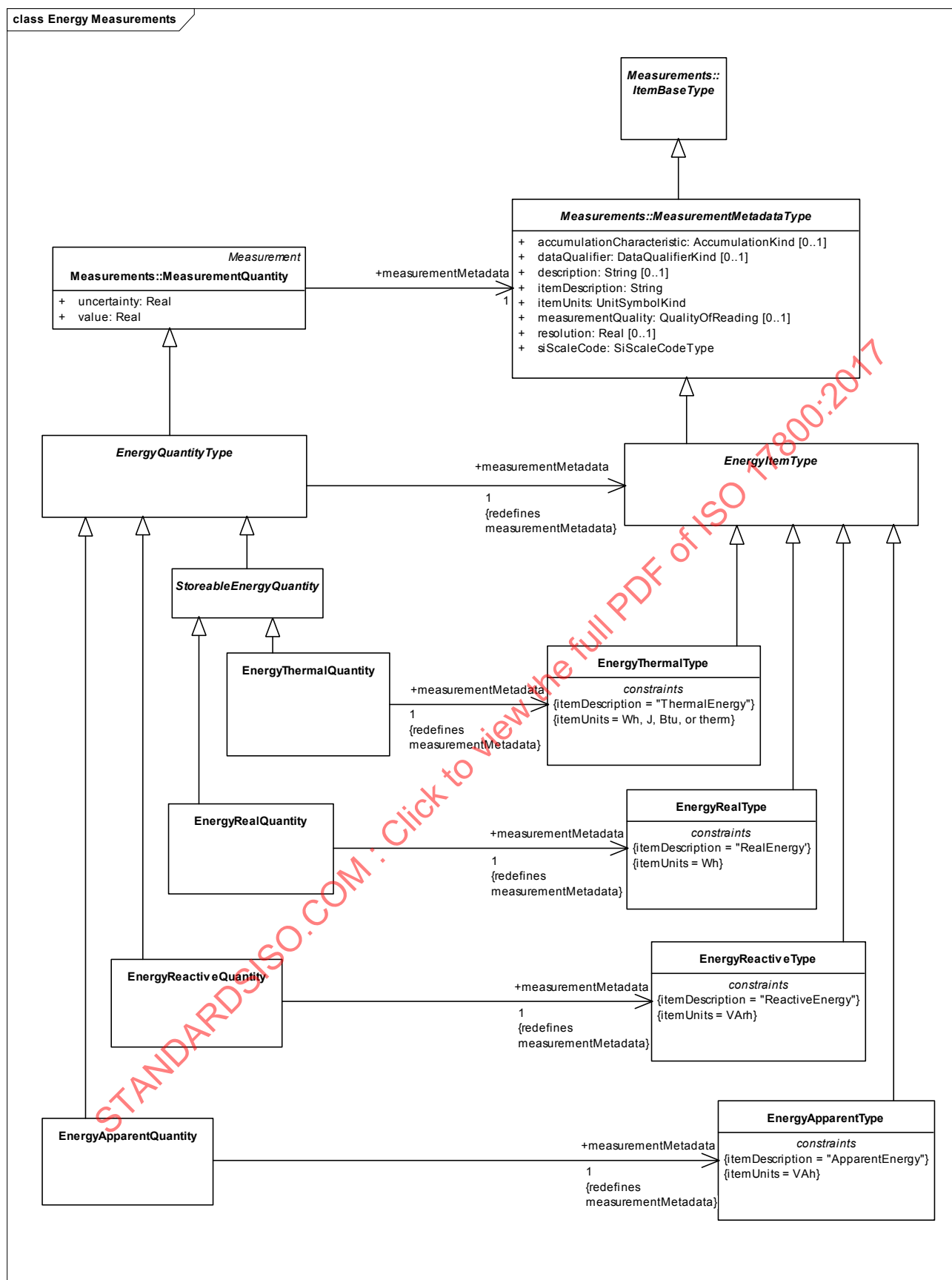


Figure 6.8 - Energy Measurements

The Energy Measurements Diagram makes use of the following elements:

EnergyApparentQuantity (Class)

Quantity/metadata pair used to represent apparent energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentQuantity*. See Clause 6.6.4.7.6 for the complete definition of this element.

EnergyApparentType (Class)

Metadata associated with measurements of apparent energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentType*. See Clause 6.6.4.7.7 for the complete definition of this element.

EnergyItemType (Abstract Class)

Metadata associated with an energy measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyItemType*. See Clause 6.6.4.7.8 for the complete definition of this element.

EnergyQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyQuantityType*. See Clause 6.6.4.7.10 for the complete definition of this element.

EnergyReactiveQuantity (Class)

Quantity/metadata pair used to represent reactive energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveQuantity*. See Clause 6.6.4.7.11 for the complete definition of this element.

EnergyReactiveType (Class)

Metadata associated with measurements of reactive energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveType*. See Clause 6.6.4.7.12 for the complete definition of this element.

EnergyRealQuantity (Class)

Quantity/metadata pair used to represent real energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantity*. See Clause 6.6.4.7.13 for the complete definition of this element.

EnergyRealType (Class)

Metadata associated with measurements of real energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealType*. See Clause 6.6.4.7.14 for the complete definition of this element.

EnergyThermalQuantity (Class)

Quantity/metadata pair used to represent thermal energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyThermalQuantity*. See Clause 6.6.4.7.15 for the complete definition of this element.

EnergyThermalType (Class)

Metadata associated with measurements of thermal energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyThermalType*. See Clause 6.6.4.7.16 for the complete definition of this element.

ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType*. See Clause 6.6.4.3 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

StoreableEnergyQuantity (Abstract Class)

Quantity used to represent real electrical or thermal energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\StoreableEnergyQuantity*. See Clause 6.6.4.7.17 for the complete definition of this element.

6.4.4.2. Sets of Energy Measurements Diagram

It is sometimes necessary to know more than one type of energy measurement for a given location and time period. For example, it is often important to know both the real energy and the reactive energy for a given point during a given time period. When this need arises, an EnergyMeasurementSet class is used. Each of the subclasses of the abstract EnergyQuantityType is optionally included within this EnergyMeasurementSet. Which of these optionally included classes is actually included in a particular real life application of the FSGIM is problem domain specific. For example, some residential applications may only need to include the EnergyRealQuantity while some industrial applications may need all of the EnergyQuantityType derived classes.

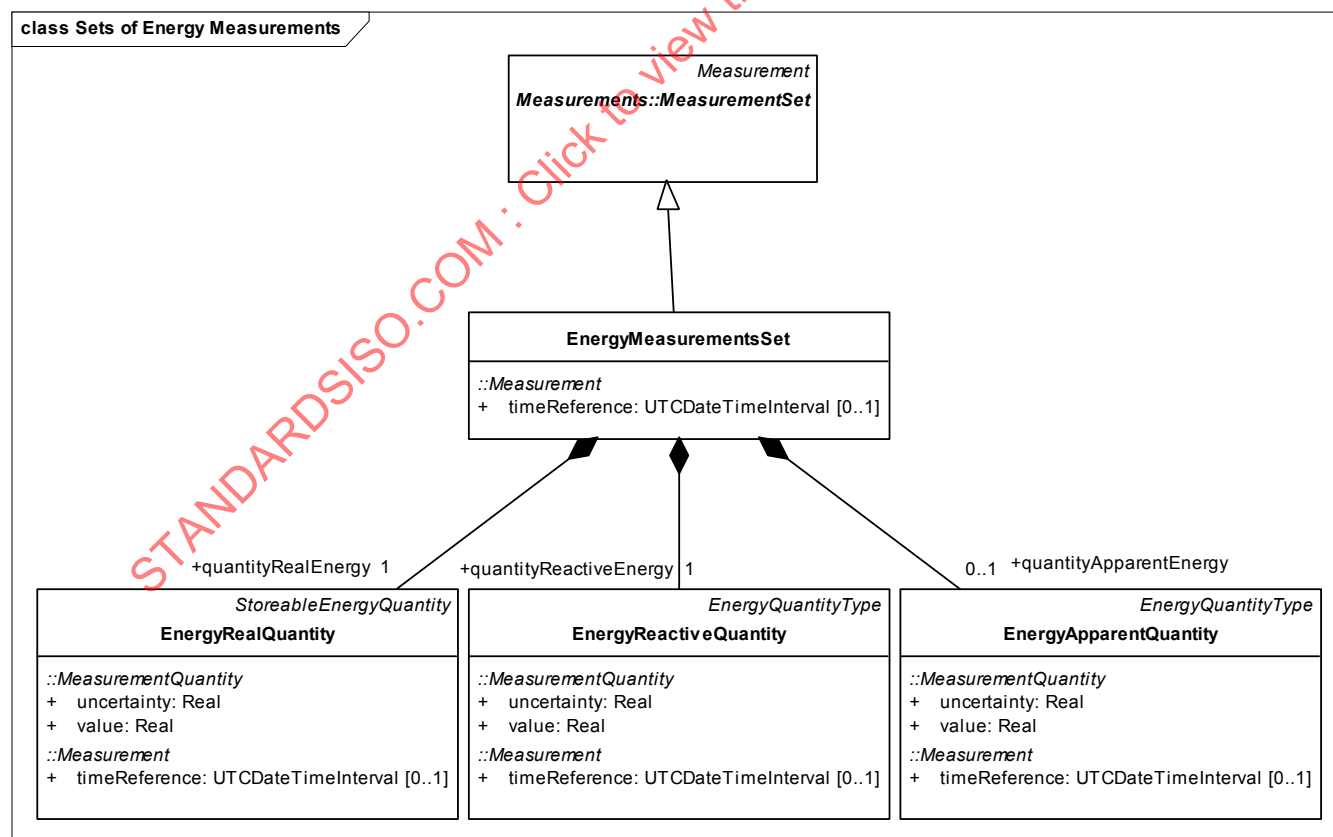


Figure 6.9 - Sets of Energy Measurements

The Sets of Energy Measurements Diagram makes use of the following elements:

EnergyApparentQuantity (Class)

Quantity/metadata pair used to represent apparent energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentQuantity*. See Clause 6.6.4.7.6 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

EnergyReactiveQuantity (Class)

Quantity/metadata pair used to represent reactive energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveQuantity*. See Clause 6.6.4.7.11 for the complete definition of this element.

EnergyRealQuantity (Class)

Quantity/metadata pair used to represent real energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantity*. See Clause 6.6.4.7.13 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

6.4.4.3. Aggregation of Energy Measurements Diagram

This diagram illustrates the rules that are generally associated with aggregations of energy measurements.

For quantities to be suitable for aggregation, they must have consistent values. For example, measurements that have identical timeReference and MeasurementMetadataType are inherently consistent. When timeReferences are not identical or MeasurementMetadataTypes are not equivalent, however, it is a local matter to decide how to adapt actual measurement values to make them consistent.

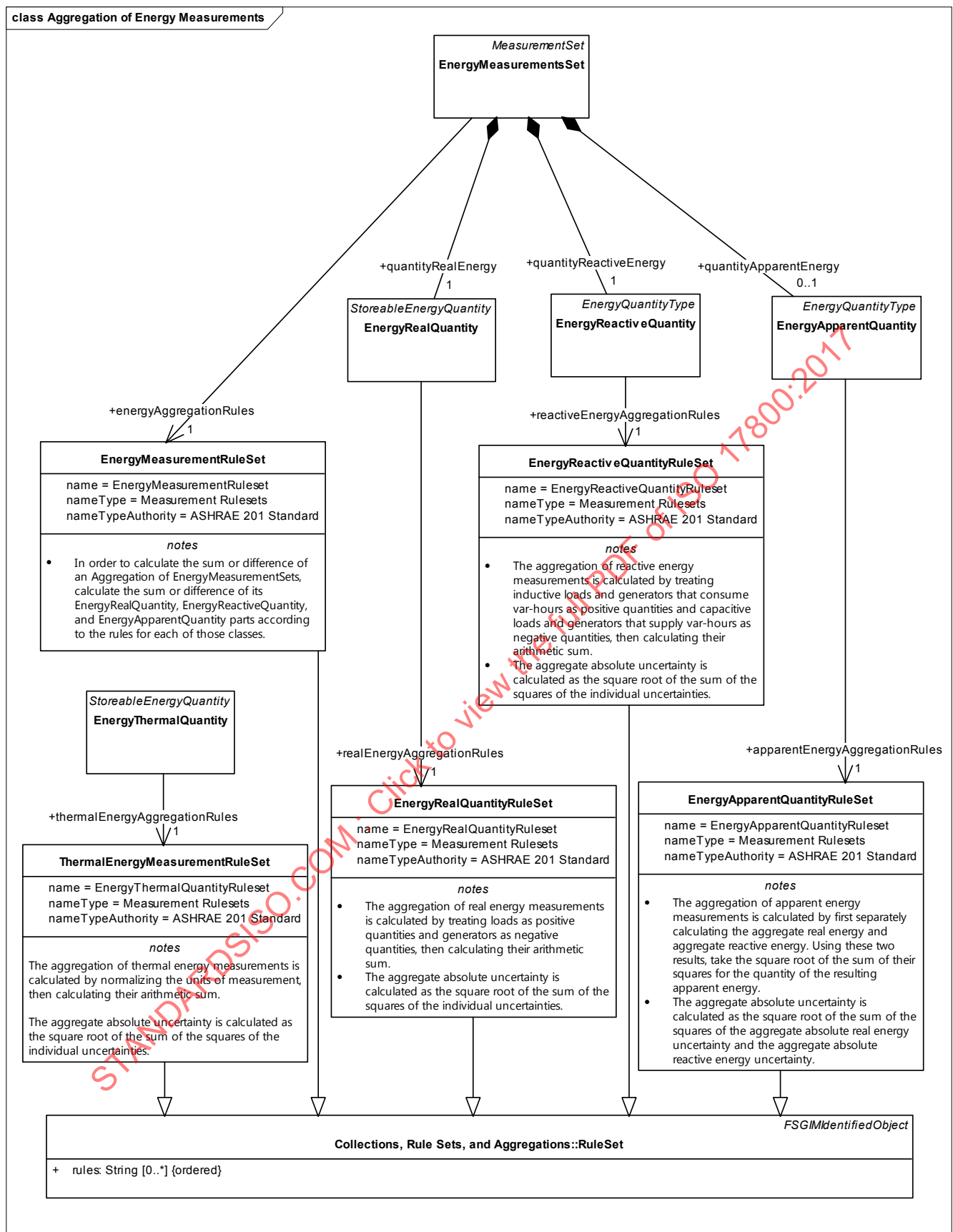


Figure 6.10 - Aggregation of Energy Measurements

The Aggregation of Energy Measurements Diagram makes use of the following elements:

EnergyApparentQuantity (Class)

Quantity/metadata pair used to represent apparent energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyApparentQuantity*. See Clause 6.6.4.7.6 for the complete definition of this element.

EnergyApparentQuantityRuleSet (Class)

- The aggregation of apparent energy measurements is calculated by first separately calculating the aggregate real energy and aggregate reactive energy. Using these two results, take the square root of the sum of their squares for the quantity of the resulting apparent energy.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the aggregate absolute real energy uncertainty and the aggregate absolute reactive energy uncertainty.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyApparentQuantityRuleSet*. See Clause 6.6.4.7.2 for the complete definition of this element.

EnergyMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EnergyMeasurementSets, calculate the sum or difference of its EnergyRealQuantity, EnergyReactiveQuantity, and EnergyApparentQuantity parts according to the rules for each of those classes.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyMeasurementRuleSet*. See Clause 6.6.4.7.3 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

EnergyReactiveQuantity (Class)

Quantity/metadata pair used to represent reactive energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyReactiveQuantity*. See Clause 6.6.4.7.11 for the complete definition of this element.

EnergyReactiveQuantityRuleSet (Class)

- The aggregation of reactive energy measurements is calculated by treating inductive loads and generators that consume var-hours as positive quantities and capacitive loads and generators that supply var-hours as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyReactiveQuantityRuleSet*. See Clause 6.6.4.7.4 for the complete definition of this element.

EnergyRealQuantity (Class)

Quantity/metadata pair used to represent real energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EnergyMeasurements\EnergyRealQuantity*. See Clause 6.6.4.7.13 for the complete definition of this element.

EnergyRealQuantityRuleSet (Class)

- The aggregation of real energy measurements is calculated by treating loads as positive quantities and generators as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantityRuleSet*. See Clause 6.6.4.7.5 for the complete definition of this element.

EnergyThermalQuantity (Class)

Quantity/metadata pair used to represent thermal energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyThermalQuantity*. See Clause 6.6.4.7.15 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

ThermalEnergyMeasurementRuleSet (Class)

The aggregation of thermal energy measurements is calculated by normalizing the units of measurement, then calculating their arithmetic sum.

The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\ThermalEnergyMeasurementRuleSet*. See Clause 6.6.4.7.1 for the complete definition of this element.

6.4.5. Emissions Measurements

This package contains the classes that are defined for the purpose of modeling emissions measurements.

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements*

6.4.5.1. Emissions Measurements Diagram

Emissions of air pollutants are represented in the FSGIM using one of the classes derived from the abstract EmissionsQuantityType and one of the classes derived from the abstract EmissionsItemType. The class derived from the EmissionsQuantityType contains the magnitude of the measured value. The class derived from the EmissionsItemType contains the metadata associated with the measured value. Multiple magnitudes of the measured value may share the same metadata or they may have separate metadata.

When other portions of the FSGIM need to include a generic emissions measurement without specifying the exact type of emissions measurement, they do this by including the EmissionsQuantityType. Even if a EmissionsItemType is not shown in the class diagram, its presence is implied due to the multiplicity of one association between the EmissionsQuantityType class and the EmissionsItemType class.

When other portions of the FSGIM need to include a specific type of emissions measurement, they do this by including either an EmissionsMassQuantity, an EmissionsConcentrationQuantity or an EmissionsMassRateQuantity depending upon whether the emissions are measured in grams, grams per cubic meter, or grams per second. Unlike with the representation of power measurements and energy measurements, if a specific type of emissions measurement needs to be included, then the corresponding EmissionsMassType, EmissionsConcentrationType, or EmissionsMassRateType must also be shown. This is necessary because the presence of an EmissionsMassQuantity, EmissionsConcentrationQuantity, or EmissionsMassRateQuantity does not sufficiently constrain the value of the itemDescription attribute to indicate what type of emissions, such as sulfur dioxide, is being measured.

As additional types of emissions measurements that are important to the smart grid are identified in the future, they may be added to the model by creating a new value for the itemDescription that indicates the type of emissions.

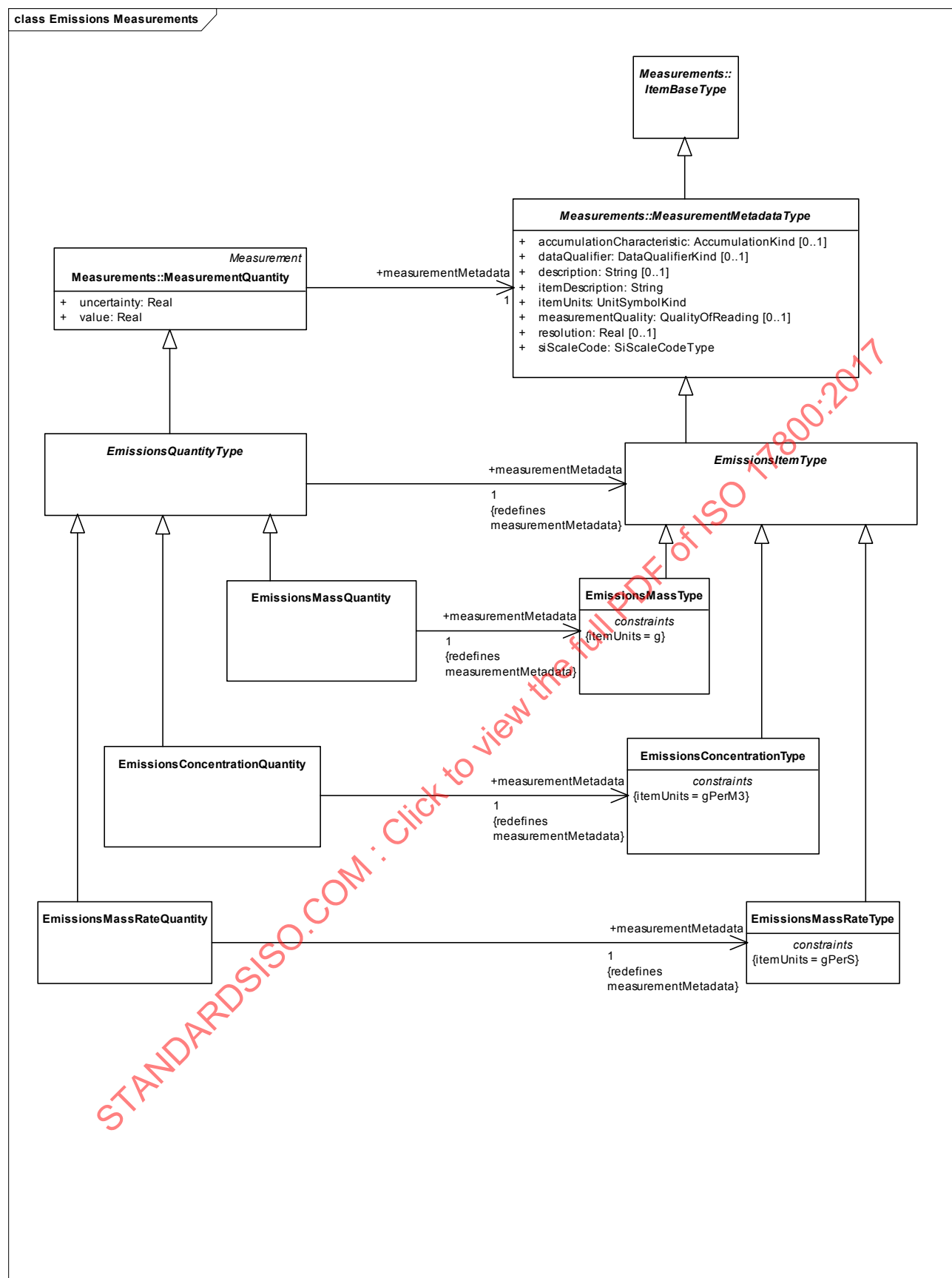


Figure 6.11 - Emissions Measurements

The Emissions Measurements Diagram makes use of the following elements:

EmissionsConcentrationQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by concentration. Note: The determination of which instances of EmissionsConcentrationQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationQuantity*. See Clause 6.6.4.8.1 for the complete definition of this element.

EmissionsConcentrationType (Class)

Metadata associated with an emissions measurement whose quantity indicates the concentration of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationType*. See Clause 6.6.4.8.3 for the complete definition of this element.

EmissionsItemType (Abstract Class)

Metadata associated with an emissions measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsItemType*. See Clause 6.6.4.8.4 for the complete definition of this element.

EmissionsMassQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass. Note: The determination of which instances of EmissionsMassQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassQuantity*. See Clause 6.6.4.8.5 for the complete definition of this element.

EmissionsMassRateQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass rate. Note: The determination of which instances of EmissionsMassRateQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassRateQuantity*. See Clause 6.6.4.8.7 for the complete definition of this element.

EmissionsMassRateType (Class)

Metadata associated with an emissions measurement whose quantity indicates the rate of mass of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassRateType*. See Clause 6.6.4.8.9 for the complete definition of this element.

EmissionsMassType (Class)

Metadata associated with an emissions measurement whose quantity indicates the total mass of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassType*. See Clause 6.6.4.8.10 for the complete definition of this element.

EmissionsQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of emissions measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsQuantityType*. See Clause 6.6.4.8.13 for the complete definition of this element.

ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType*. See Clause 6.6.4.3 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

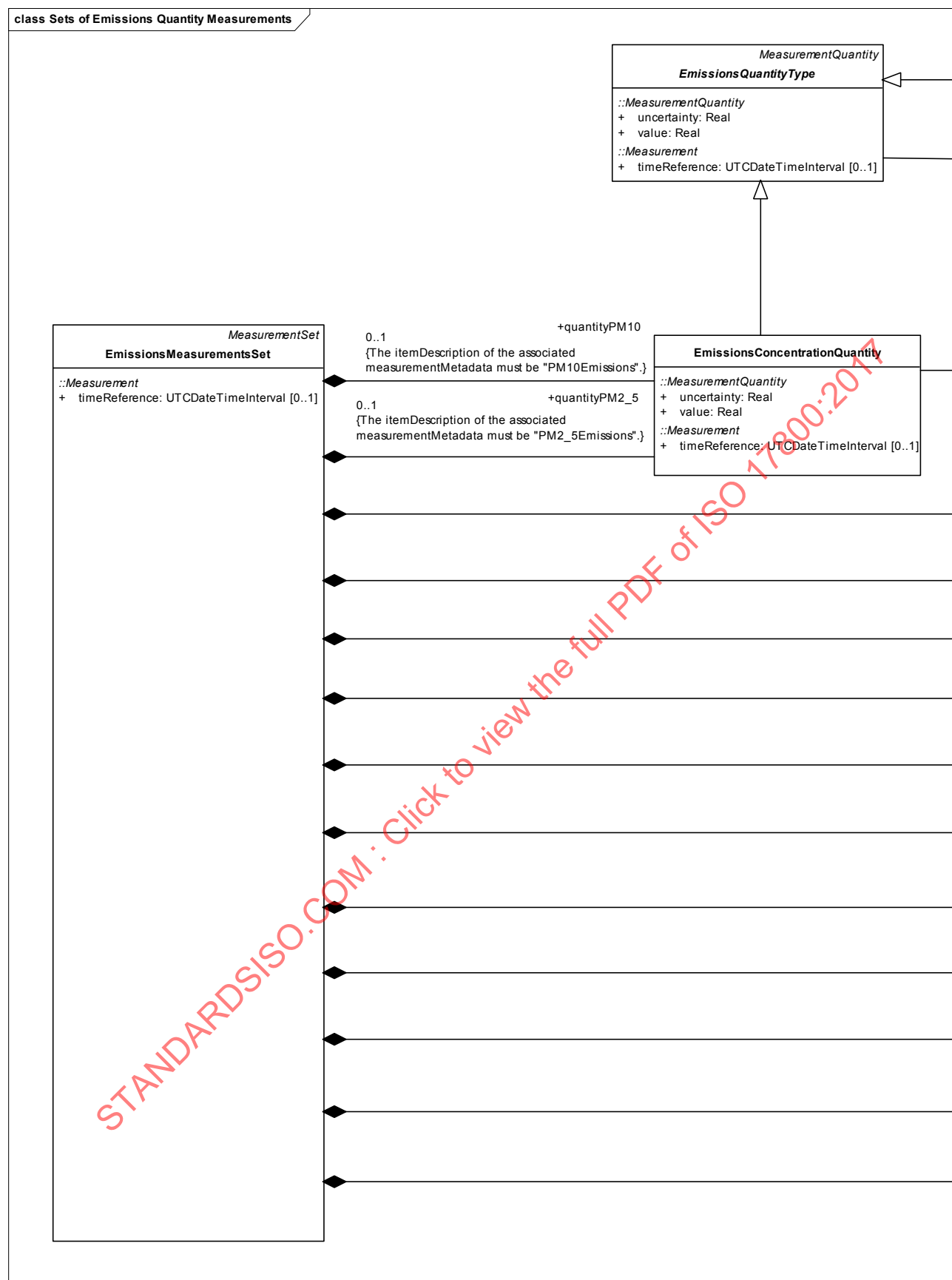
6.4.5.2. Sets of Emissions Quantity Measurements Diagram

It is sometimes necessary to know more than one type of emissions measurement for a given location and time period. For example, it may be necessary to measure sulfur dioxide, nitrogen dioxide, and mercury for a given point during a given time period. When this need arises, an EmissionsMeasurementSet class is used. Each of the individual emissions that can be represented using a subclass of the abstract EmissionsQuantityType is optionally included within this EmissionsMeasurementSet.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



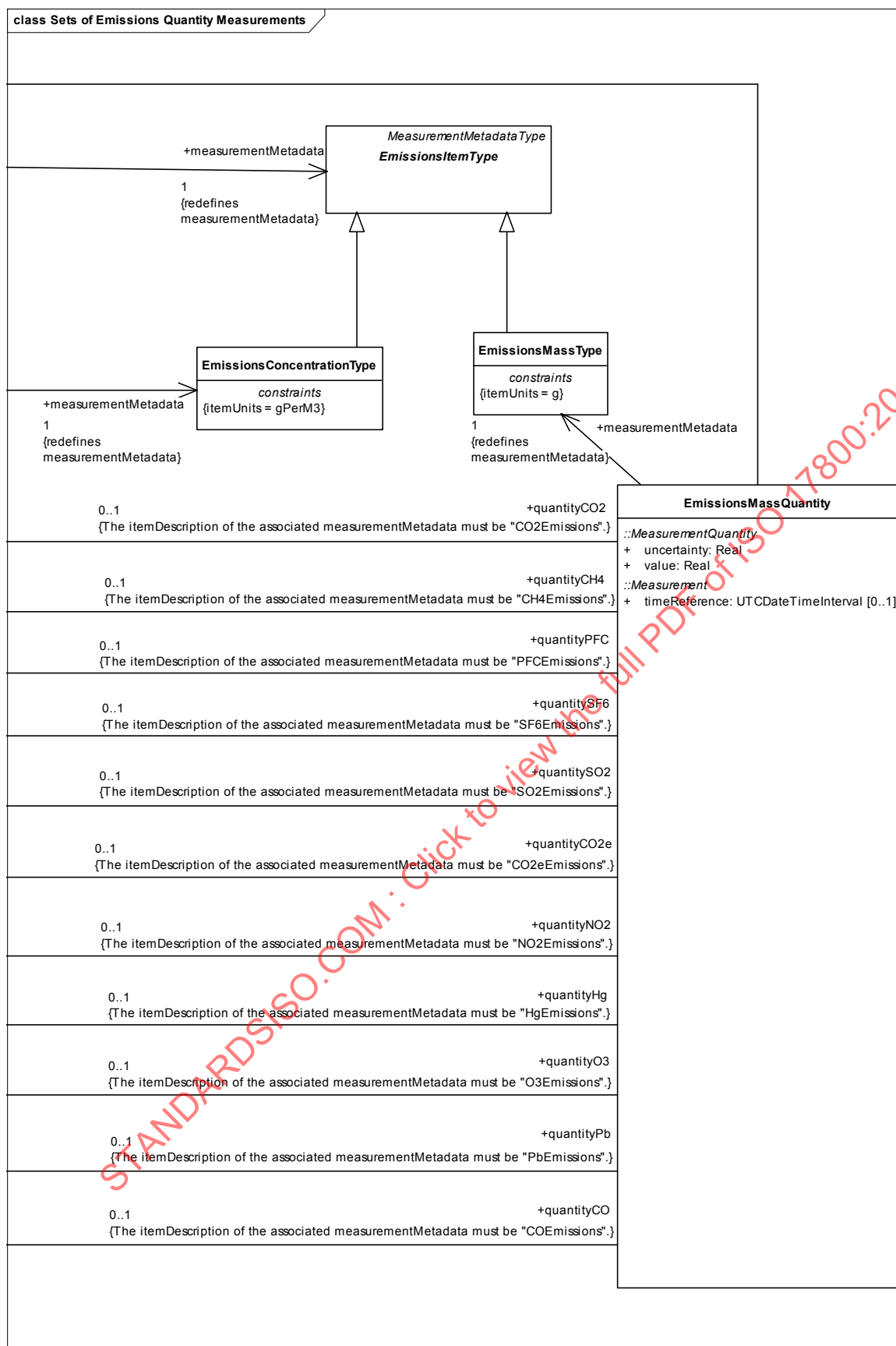


Figure 6.12 - Sets of Emissions Quantity Measurements

The Sets of Emissions Quantity Measurements Diagram makes use of the following elements:

EmissionsConcentrationQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by concentration. Note: The determination of which instances of EmissionsConcentrationQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationQuantity*. See Clause 6.6.4.8.1 for the complete definition of this element.

EmissionsConcentrationType (Class)

Metadata associated with an emissions measurement whose quantity indicates the concentration of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationType*. See Clause 6.6.4.8.3 for the complete definition of this element.

EmissionsItemType (Abstract Class)

Metadata associated with an emissions measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsItemType*. See Clause 6.6.4.8.4 for the complete definition of this element.

EmissionsMassQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass. Note: The determination of which instances of EmissionsMassQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassQuantity*. See Clause 6.6.4.8.5 for the complete definition of this element.

EmissionsMassType (Class)

Metadata associated with an emissions measurement whose quantity indicates the total mass of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassType*. See Clause 6.6.4.8.10 for the complete definition of this element.

EmissionsMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions quantity measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMeasurementsSet*. See Clause 6.6.4.8.12 for the complete definition of this element.

EmissionsQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of emissions measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsQuantityType*. See Clause 6.6.4.8.13 for the complete definition of this element.

6.4.5.3. Sets of Emissions Rate Measurements Diagram

It is sometimes necessary to know more than one type of emissions rate measurement for a given location and time period. For example, it may be necessary to measure the rate of sulfur dioxide, nitrogen dioxide, and mercury emissions for a given point during a given time period. When this need arises, an EmissionsRateMeasurementSet class is used. Each of the individual emissions rates that can be represented using a subclass of the abstract EmissionsQuantityType is optionally included within this EmissionsRateMeasurementSet.

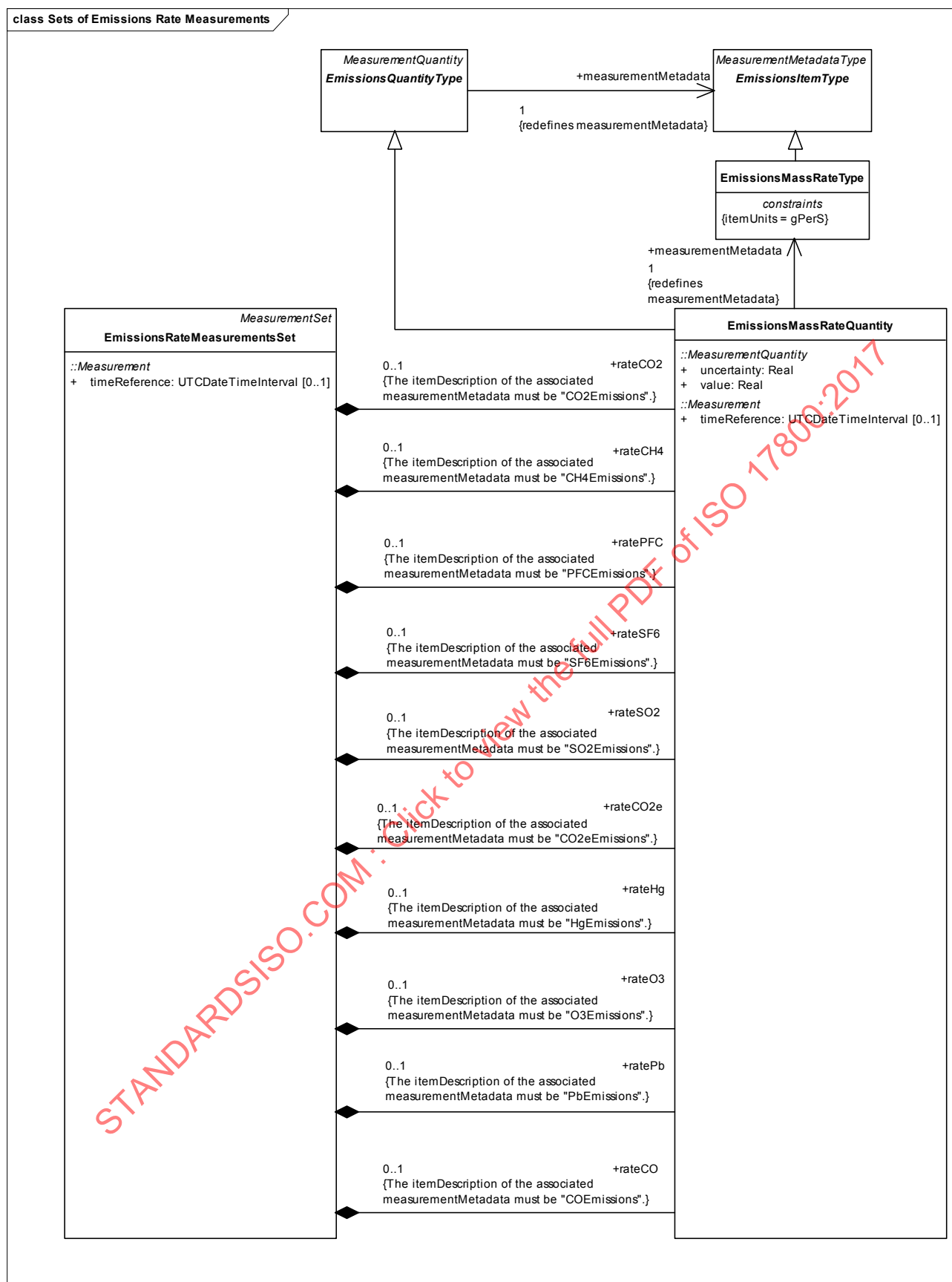


Figure 6.13 - Sets of Emissions Rate Measurements

The Sets of Emissions Rate Measurements Diagram makes use of the following elements:

EmissionsItemType (Abstract Class)

Metadata associated with an emissions measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsItemType*. See Clause 6.6.4.8.4 for the complete definition of this element.

EmissionsMassRateQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass rate.

Note: The determination of which instances of EmissionsMassRateQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassRateQuantity*. See Clause 6.6.4.8.7 for the complete definition of this element.

EmissionsMassRateType (Class)

Metadata associated with an emissions measurement whose quantity indicates the rate of mass of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassRateType*. See Clause 6.6.4.8.9 for the complete definition of this element.

EmissionsQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of emissions measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsQuantityType*. See Clause 6.6.4.8.13 for the complete definition of this element.

EmissionsRateMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions rate measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsRateMeasurementsSet*. See Clause 6.6.4.8.15 for the complete definition of this element.

6.4.5.4. Aggregation of Emissions Measurements Diagram

This diagram illustrates the rules that are generally associated with aggregations of emissions measurements.

For quantities to be suitable for aggregation, they must have consistent values. For example, measurements that have identical timeReference and MeasurementMetadataType are inherently consistent. When timeReferences are not identical or MeasurementMetadataTypes are not equivalent, however, it is a local matter to decide how to adapt actual measurement values to make them consistent.

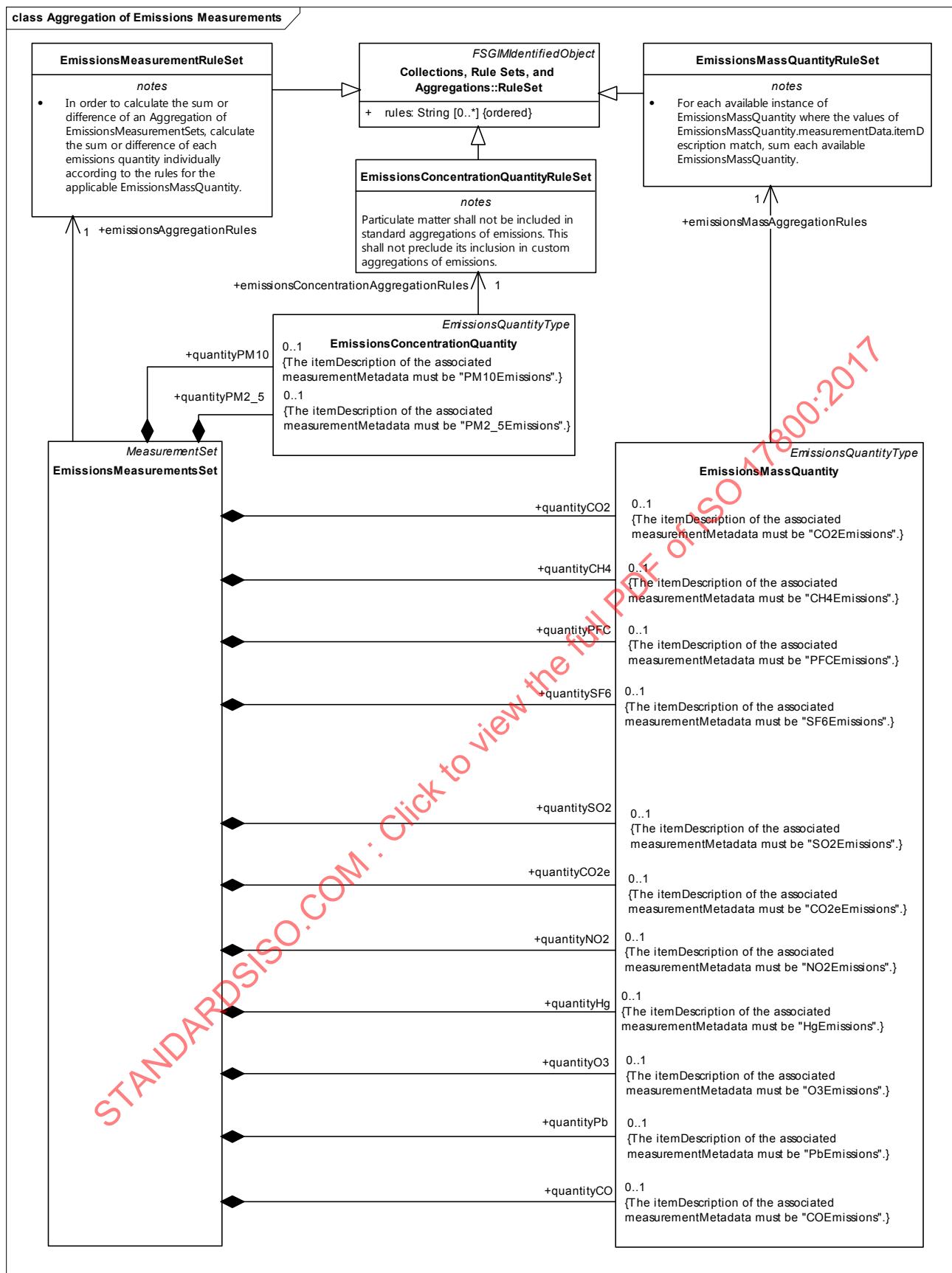


Figure 6.14 - Aggregation of Emissions Measurements

The Aggregation of Emissions Measurements Diagram makes use of the following elements:

EmissionsConcentrationQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by concentration. Note: The determination of which instances of EmissionsConcentrationQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationQuantity*. See Clause 6.6.4.8.1 for the complete definition of this element.

EmissionsConcentrationQuantityRuleSet (Class)

Particulate matter shall not be included in standard aggregations of emissions. This shall not preclude its inclusion in custom aggregations of emissions.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationQuantityRuleSet*. See Clause 6.6.4.8.2 for the complete definition of this element.

EmissionsMassQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass. Note: The determination of which instances of EmissionsMassQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassQuantity*. See Clause 6.6.4.8.5 for the complete definition of this element.

EmissionsMassQuantityRuleSet (Class)

- For each available instance of EmissionsMassQuantity where the values of EmissionsMassQuantity.measurementData.itemDescription match, sum each available EmissionsMassQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassQuantityRuleSet*. See Clause 6.6.4.8.6 for the complete definition of this element.

EmissionsMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EmissionsMeasurementSets, calculate the sum or difference of each emissions quantity individually according to the rules for the applicable EmissionsMassQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMeasurementRuleSet*. See Clause 6.6.4.8.11 for the complete definition of this element.

EmissionsMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions quantity measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMeasurementsSet*. See Clause 6.6.4.8.12 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

6.4.5.5. Aggregation of Emissions Rate Measurements Diagram

This diagram illustrates the rules that are generally associated with aggregations of emissions rate measurements.

For quantities to be suitable for aggregation, they must have consistent values. For example, measurements that have identical timeReference and MeasurementMetadataType are inherently consistent. When timeReferences are not

identical or MeasurementMetadataTypes are not equivalent, however, it is a local matter to decide how to adapt actual measurement values to make them consistent.

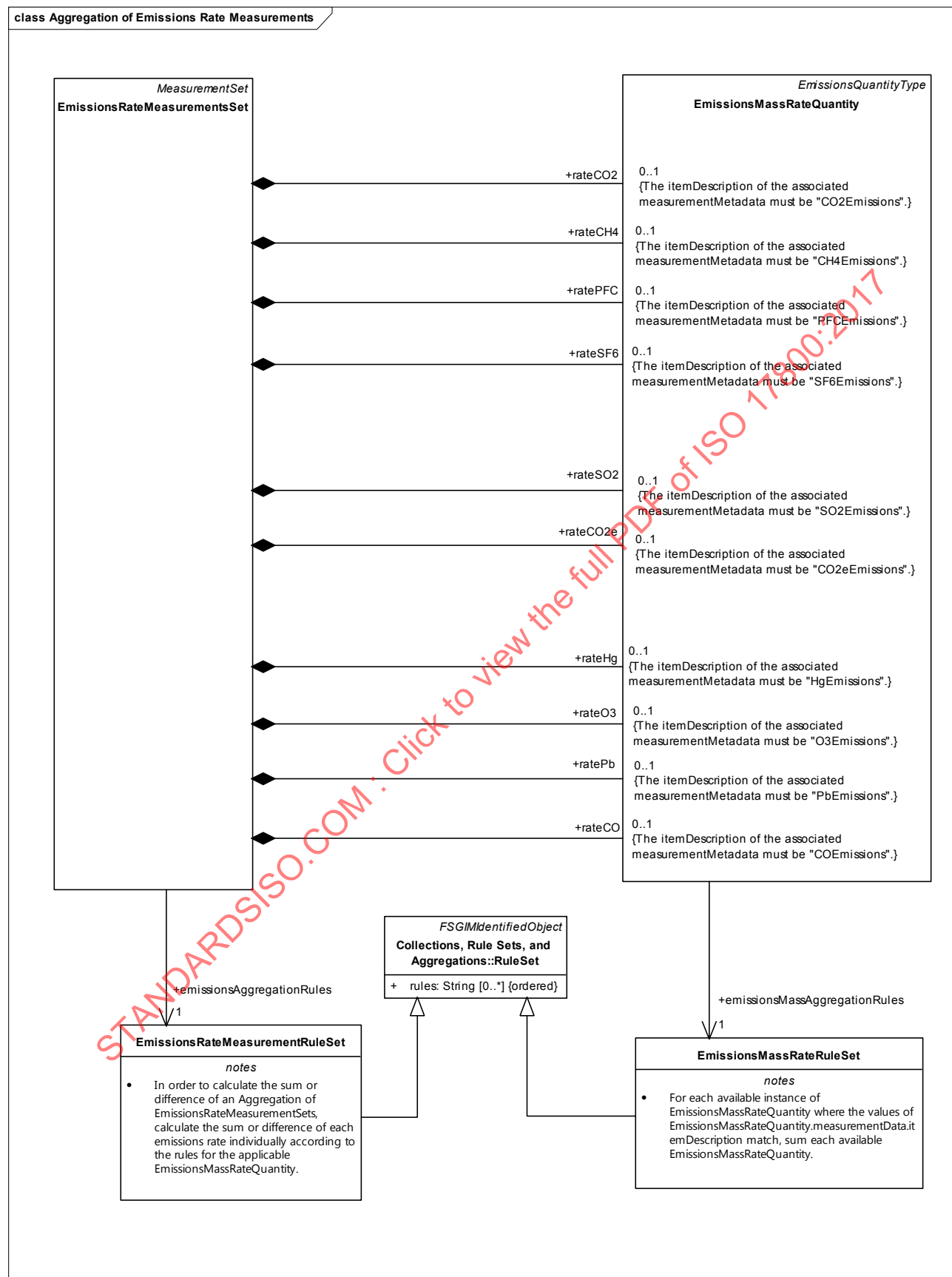


Figure 6.15 - Aggregation of Emissions Rate Measurements

The Aggregation of Emissions Rate Measurements Diagram makes use of the following elements:

EmissionsMassRateQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass rate.

Note: The determination of which instances of EmissionsMassRateQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassRateQuantity*. See Clause 6.6.4.8.7 for the complete definition of this element.

EmissionsMassRateRuleSet (Class)

- For each available instance of EmissionsMassRateQuantity where the values of EmissionsMassRateQuantity.measurementData.itemDescription match, sum each available EmissionsMassRateQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassRateRuleSet*. See Clause 6.6.4.8.8 for the complete definition of this element.

EmissionsRateMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EmissionsRateMeasurementSets, calculate the sum or difference of each emissions rate individually according to the rules for the applicable EmissionsMassRateQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsRateMeasurementRuleSet*. See Clause 6.6.4.8.14 for the complete definition of this element.

EmissionsRateMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions rate measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsRateMeasurementsSet*. See Clause 6.6.4.8.15 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

6.5. Other Common Classes

The package contains common classes and types that are used within the Facility Smart Grid Information Model other than primitive or measurement related classes and types .

UML Package Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes*

6.5.1. Circuit Diagram

In the FSGIM, the modeling of the electrical topology of the facility is accomplished using the Circuit Class and the ConnectionPoint Class (or its children). Each electrical circuit in the facility is modeled as a separate instance of the Circuit Class. The Load Class, Generator Class, Meter Class, and EM Class all have associations to one or more instances of the ConnectionPoint Class or children of the ConnectionPoint Class. These instances represent the electrical connections that loads, generators, meters, and energy managers (via their energy routers) have to other devices in the facility or to the grid. If two or more instances of the ConnectionPoint Class are connected to the same instance of the Circuit Class, then they are electrically connected to each other.

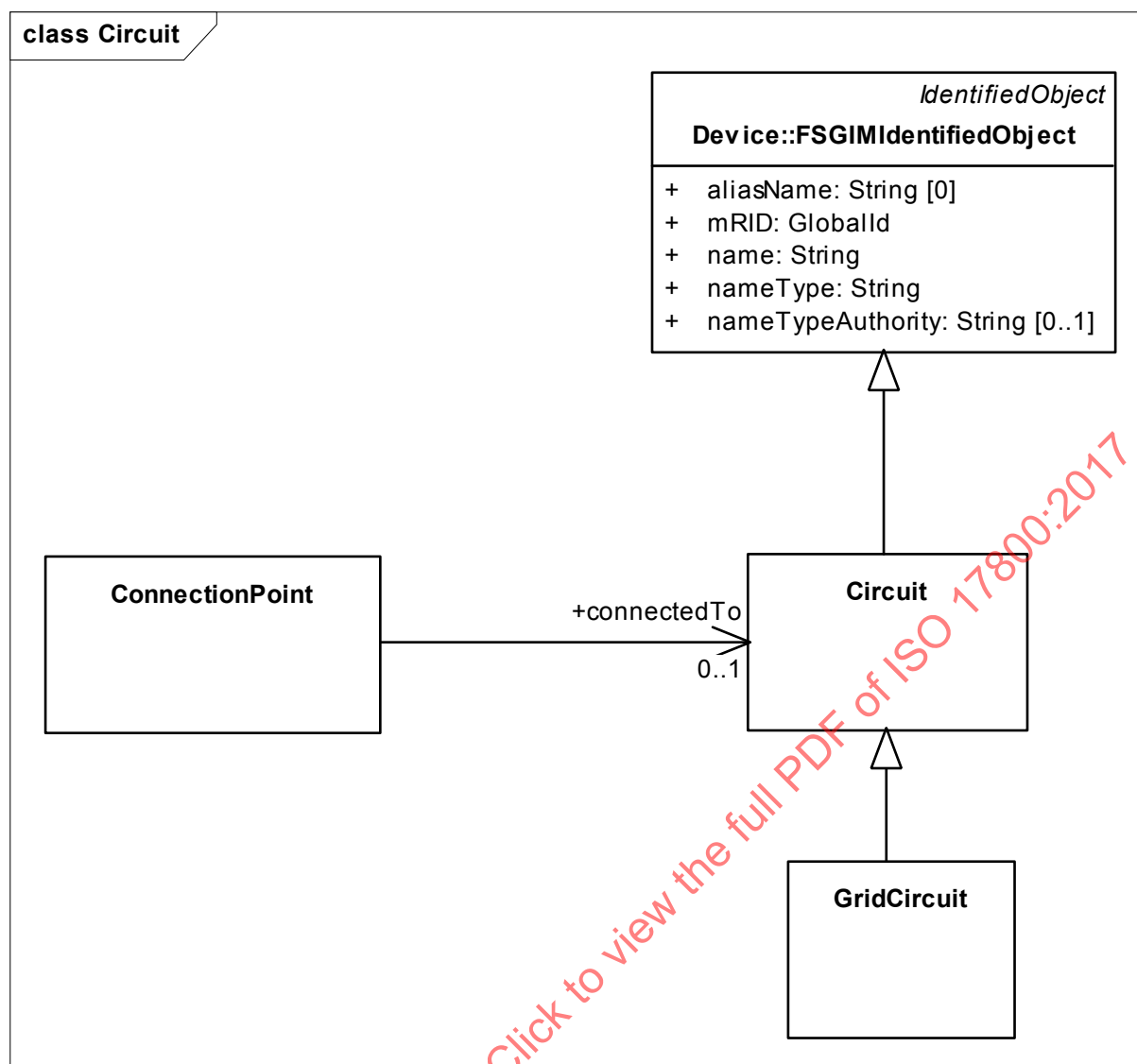


Figure 6.16 - Circuit

The Circuit Diagram makes use of the following elements:

Circuit (Class)

This class is used to model electrical circuits. All ConnectionPoints that are electrically tied together will point to the same instance of this class.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\Circuit*. See Clause 6.6.5.1 for the complete definition of this element.

ConnectionPoint (Class)

This class is used to model the physical connections from Generators, Loads, Meters, and Energy Routers to sources or sinks of energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\ConnectionPoint*. See Clause 6.6.5.2 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

GridCircuit (Class)

This class is used to model electrical circuits that are part of the utility grid.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\GridCircuit*. See Clause 6.6.5.3 for the complete definition of this element.

6.5.2. Monetary Quantities Diagram

Monetary values are expressed in the FSGIM using the MonetaryQuantity Class.

The extensive enumeration of three-character CurrencyKind values has been hidden in this drawing in order to keep the drawing from spanning multiple pages. See the documentation for CurrencyKind for the complete list.

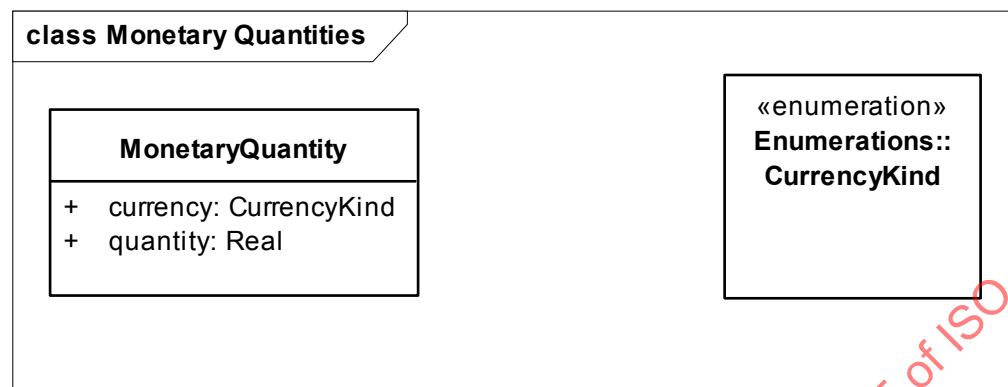


Figure 6.17 - Monetary Quantities

The Monetary Quantities Diagram makes use of the following elements:

CurrencyKind (Enumeration)

Currency is represented in FSGIM as an enumeration. The use of this is optional so it can be omitted when an actual currency is not known. The enumeration is based on ISO 4217 three character codes and the FSIM model contains a snapshot of this list. Any implementation can extend this list as needed but must unambiguously identify how extensions are separate from the ISO standard set.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\CurrencyKind*. See Clause 6.6.2.1 for the complete definition of this element.

MonetaryQuantity (Class)

A monetary quantity used to express the cost or value of something in some currency.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\MonetaryQuantity*. See Clause 6.6.5.4 for the complete definition of this element.

6.6. Elements Defined in the Common Primitive Types, Classes, and Enumerations Model

The following elements are defined in this model.

6.6.1. Elements Defined in the Time Model

The following elements are defined in this model.

6.6.1.1. DateTimeInterval (Abstract Class)

An abstract class representing an interval of date and time.

UML Element Location: *Model* \ *Common Primitive Types, Classes, and Enumerations* \ *Time* \ *DateTimeInterval*.

6.6.1.2. DstTransitionRule (Class)

Rules for DST transition.

UML Element Location: *Model* \ *Common Primitive Types, Classes, and Enumerations* \ *Time* \ *DstTransitionRule*.

Table 6.1 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
month	Month at which DST transition occurs 1-12.	MonthType See Clause 6.6.2.2.	[1]
rule	Rule by which DST transition is determined.	DstRule See Clause 6.6.1.6.	[1]
dayofmonth	Day of the month 0 = not applicable, 1 - 31	Integer See Clause 6.6.3.5.	[1]
dow	Day of week at which DST transition occurs: 0 = not applicable, 1 - 7 (Monday = 1).	WeekdayType See Clause 6.6.2.4.	[1]
hours	Hour at which DST transition occurs 0-23.	Integer See Clause 6.6.3.5.	[1]
seconds	Seconds at which transition occurs 0-3599.	Integer See Clause 6.6.3.5.	[1]

6.6.1.3. LocalDateTimeInterval «Compound» (Class)

An interval of date and time where the start and end times are expressed in local time.

Parent Class: DateTimeInterval (See Clause 6.6.1.1.)

UML Element Location: *Model* \ *Common Primitive Types, Classes, and Enumerations* \ *Time* \ *LocalDateTimeInterval*.

Table 6.2 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
duration	The length of time for this interval. If duration is specified, it must be greater than zero.	Duration See Clause 6.6.1.9.	[0..1]
end	The date and time at which this interval terminates.	LocalDateTime See Clause 6.6.1.10.	[0..1]
start	The date and time at which this interval begins.	LocalDateTime See Clause 6.6.1.10.	[0..1]

6.6.1.4. UTCDateTimeInterval «Compound» (Class)

Interval of date and time.

Constraint: An interval planned for the future need only have those attributes that have been determined. For example one might know the duration of a future production run as well as projected characteristics of its energy use, but not have yet scheduled it precisely

Constraint: At least one at at most two of the attributes start, end, and duration must be specified.

Constraint: When the start and end attributes are defined, the start attribute must precede the end attribute in time.

Parent Class: DateTimeInterval (See Clause 6.6.1.1.)

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Time* | *UTCDateTimeInterval*.

Table 6.3 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
start	Date and time that this interval started.	UTCDateTime See Clause 6.6.1.11.	[0..1]
end	Date and time that this interval ended.	UTCDateTime See Clause 6.6.1.11.	[0..1]
duration	The duration of this interval, in seconds Constraint: If duration is specified, it must be greater than zero.	Duration See Clause 6.6.1.9.	[0..1]

6.6.1.5. LocalTimeParameters (Class)

Contains attributes related to the configuration of the time service.

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Time* | *LocalTimeParameters*.

Table 6.4 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
dstEndRule	Rule to calculate end of daylight savings time in the current year.	DstTransitionRule See Clause 6.6.1.2.	[1]
dstOffset	Daylight savings time offset from local standard time in seconds.	Integer See Clause 6.6.3.5.	[1]
dstStartRule	Rule to calculate start of daylight savings time in the current year.	DstTransitionRule See Clause 6.6.1.2.	[1]
tzOffset	Local time zone offset from UTC time in seconds. Does not include any daylight savings time offsets.	Integer See Clause 6.6.3.5.	[1]
tzid	Time Zone Database ID as per IANA https://www.iana.org/time-zones. This optional attribute can be used to retrieve the IANA database and construct the values for the other attributes of this class.	String See Clause 6.6.3.7.	[0..1]

6.6.1.6. DstRule (Enumeration)

Enumeration of the rules for Daylight Savings Time transitions.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\DstRule*.

Table 6.5 – DstRule Enumerated Values

Enumerated Value	Definition
dayOfMonth	DST starts/ends on this day of the month.
onOrAfter	DST starts/ends on the Day of the Week that is on or after the Day of the Month
firstOccurrence	DST starts/ends on the first occurrence of the Day of the Week in a month
secondOccurrence	DST starts/ends on the second occurrence of the Day of the Week in a month.
thirdOccurrence	DST starts/ends on the third occurrence of the Day of the Week in a month.
fourthOccurrence	DST starts/ends on the fourth occurrence of the Day of the Week in a month.
fifthOccurrence	DST starts/ends on the fifth occurrence of the Day of the Week in a month.
last	DST starts/ends on the last occurrence of the Day of the Week in a month.

6.6.1.7. DateTime (PrimitiveType)

An abstract primitive datatype representing a date and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\DateTime*.

6.6.1.8. Date (PrimitiveType)

This represents a local date without a time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\Date.*

6.6.1.9. Duration (PrimitiveType)

This primitive data type represents a length of time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\Duration.*

6.6.1.10. LocalDateTime (PrimitiveType)

This primitive datatype allows the specification of a date with time of day in the local time of the location to which it pertains.

Parent Class: DateTime (See Clause 6.6.1.7.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\LocalDateTime.*

6.6.1.11. UTCDateTime (PrimitiveType)

This primitive datatype represents a specific UTC time on a specific UTC date consistent with the time, date, and UTC concepts described in ISO 8601.

Parent Class: DateTime (See Clause 6.6.1.7.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Time\UTCDateTime.*

6.6.2. Elements Defined in the Enumerations Model

The following elements are defined in this model.

6.6.2.1. CurrencyKind (Enumeration)

Currency is represented in FSGIM as an enumeration. The use of this is optional so it can be omitted when an actual currency is not known. The enumeration is based on ISO 4217 three character codes and the FSIM model contains a snapshot of this list. Any implementation can extend this list as needed but must unambiguously identify how extensions are separate from the ISO standard set.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\CurrencyKind.*

Table 6.6 - CurrencyKind Enumerated Values

Enumerated Value	Definition
AED	United Arab Emirates dirham
AFN	Afghan afghani

Enumerated Value	Definition
ALL	Albanian lek
AMD	Armenian dram
ANG	Netherlands Antillean guilder
AOA	Angolan kwanza
ARS	Argentine peso
AUD	Australian dollar
AWG	Aruban florin
AZN	Azerbaijani manat
BAM	Bosnia and Herzegovina convertible mark
BBB	Barbados dollar
BDT	Bangladeshi taka
BGN	Bulgarian lev
BHD	Bahraini dinar
BIF	Burundian franc
BMD	Bermudian dollar (customarily known as Bermuda dollar)
BND	Brunei dollar
BOB	Boliviano
BOV	Bolivian Mvdol (funds code)
BRL	Brazilian real
BSD	Bahamian dollar
BTN	Bhutanese ngultrum
BWP	Botswana pula
BYR	Belarusian ruble
BZD	Belize dollar
CAD	Canadian dollar
CDF	Congolese franc
CHF	Swiss franc
CLF	Unidad de Fomento (funds code), Chile
CLP	Chilean peso
CNY	Chinese yuan
COP	Colombian peso
COU	Unidad de Valor Real
CRC	Costa Rican colon
CUC	Cuban convertible peso
CUP	Cuban peso
CVE	Cape Verde escudo
CZK	Czech koruna
DJF	Djiboutian franc

Enumerated Value	Definition
DKK	Danish krone
DOP	Dominican peso
DZD	Algerian dinar
EEK	Estonian kroon
EGP	Egyptian pound
ERN	Eritrean nakfa
ETB	Ethiopian birr
EUR	Euro
FJD	Fiji dollar
FKP	Falkland Islands pound
GBP	Pound sterling
GEL	Georgian lari
GHS	Ghanaian cedi
GIP	Gibraltar pound
GMD	Gambian dalasi
GNF	Guinean franc
GTQ	Guatemalan quetzal
GYD	Guyanese dollar
HKD	Hong Kong dollar
HNL	Honduran lempira
HRK	Croatian kuna
HTG	Haitian gourde
HUF	Hungarian forint
IDR	Indonesian rupiah
ILS	Israeli new sheqel
INR	Indian rupee
IQD	Iraqi dinar
IRR	Iranian rial
ISK	Icelandic króna
JMD	Jamaican dollar
JOD	Jordanian dinar
JPY	Japanese yen
KES	Kenyan shilling
KGS	Kyrgyzstani som
KHR	Cambodian riel
KMF	Comoro franc
KPW	North Korean won
KRW	South Korean won

Enumerated Value	Definition
KWD	Kuwaiti dinar
KYD	Cayman Islands dollar
KZT	Kazakhstani tenge
LAK	Lao kip
LBP	Lebanese pound
LKR	Sri Lanka rupee
LRD	Liberian dollar
LSL	Lesotho loti
LTL	Lithuanian litas
LVL	Latvian lats
LYD	Libyan dinar
MAD	Moroccan dirham
MDL	Moldovan leu
MGA	Malagasy ariary
MKD	Macedonian denar
MMK	Myanma kyat
MNT	Mongolian tugrik
MOP	Macanese pataca
MRO	Mauritanian ouguiya
MUR	Mauritian rupee
MVR	Maldivian rufiyaa
MWK	Malawian kwacha
MXN	Mexican peso
MYR	Malaysian ringgit
MZN	Mozambican metical
NAD	Namibian dollar
NGN	Nigerian naira
NIO	Cordoba oro
NOK	Norwegian krone
NPR	Nepalese rupee
NZD	New Zealand dollar
OMR	Omani rial
PAB	Panamanian balboa
PEN	Peruvian nuevo sol
PGK	Papua New Guinea kina
PHP	Philippine peso
PKR	Pakistani rupee
PLN	Polish zloty

Enumerated Value	Definition
PYG	Paraguayan guaraní
QAR	Qatari riyal
RON	Romanian new leu
RSD	Serbian dinar
RUB	Russian rouble
RWF	Rwandan franc
SAR	Saudi riyal
SBD	Solomon Islands dollar
SCR	Seychelles rupee
SDG	Sudanese pound
SEK	Swedish krona/kronor
SGD	Singapore dollar
SHP	Saint Helena pound
SLL	Sierra Leonean leone
SOS	Somali shilling
SRD	Surinamese dollar
STD	São Tomé and Príncipe dobra
SYP	Syrian pound
SZL	Lilangeni
THB	Thai baht
TJS	Tajikistani somoni
TMT	Turkmenistani manat
TND	Tunisian dinar
TOP	Tongan paʻanga
TRY	Turkish lira
TTD	Trinidad and Tobago dollar
TWD	New Taiwan dollar
TZS	Tanzanian shilling
UAH	Ukrainian hryvnia
UGX	Ugandan shilling
USD	United States dollar
UYU	Uruguayan peso
UZS	Uzbekistan som
VEF	Venezuelan bolívar fuerte
VND	Vietnamese đồng
VUV	Vanuatu vatu
WST	Samoan tala
XAF	CFA franc BEAC

Enumerated Value	Definition
XCD	East Caribbean dollar
XOF	CFA Franc BCEAO
XPF	CFP franc
YER	Yemeni rial
ZAR	South African rand
ZMK	Zambian kwacha
ZWL	Zimbabwe dollar

6.6.2.2. MonthType (Enumeration)

An enumerated list of months of the year.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\MonthType*.

Table 6.7 – MonthType Enumerated Values

Enumerated Value	Definition
january	January
february	February
march	March
april	April
may	May
june	June
july	July
august	August
september	September
october	October
november	November
december	December

6.6.2.3. SiScaleCodeType (Enumeration)

Unit multiplier, where the initial value of literal equals the exponent of the multiplier value in base 10. Only the value "none" is normative to the FSGIM.

Note: A value that is representing a percentage can use the UnitSymbolKind of 1 (none) and an SiScaleCodeType of -2 (centi).

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\SiScaleCodeType*.

Table 6.8 - SiScaleCodeType Enumerated Values

Enumerated Value	Definition
y	yocto (10^{-24})
z	zepto (10^{-21})
a	atto (10^{-18})
f	femto (10^{-15})
p	pico (10^{-12})
n	nano (10^{-9})
micro	micro (10^{-6})
m	milli (10^{-3})
c	centi (10^{-2})
d	deci (10^{-1})
none	1 (10^0)
da	deca (10^1)
h	hecto (10^2)
k	kilo (10^3)
M	mega (10^6)
G	giga (10^9)
T	tera (10^{12})
P	petra (10^{15})
E	exa (10^{18})
Z	zetta (10^{21})
Y	yotta (10^{24})

6.6.2.4. WeekdayType (Enumeration)

An enumerated list of day of the week.

UML Element Location: *Model* \ *Common Primitive Types, Classes, and Enumerations* \ *Enumerations* \ *WeekdayType*

Table 6.9 - WeekdayType Enumerated Values

Enumerated Value	Definition
sunday	Sunday
monday	Monday
tuesday	Tuesday
wednesday	Wednesday
thursday	Thursday
friday	Friday
saturday	Saturday

6.6.2.5. QualityOfReading (Enumeration)

List of codes indicating the quality of the reading.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\QualityOfReading.*

Table 6.10 – QualityOfReading Enumerated Values

Enumerated Value	Definition
estimated	The quality is estimated.
forecast	This is forecast data.
mixed	There is a mixture of data quality.
raw	This is raw data.
validated	This data has been validated
normalizedForWeather	This data has been normalized for weather.
other	There is no defined quality for this data.

6.6.2.6. UnitSymbolKind (Enumeration)

This identifies the units of measure based on the NIST Special Publication 811 2008, edition -- Guide for the Use of the International System of Units (SI). Note that these reference document supports SI units as well as common units of measurements in practice including CGS units.

Only VA, W, var, VAh, Wh, varh, Btu, J, therm, BtuPerh, g, gPerM3, gPerS, m, mPerS, wPerM2, rad, sb, pa, and degK are normative for the FSGIM standard. The enumerated values of VA, W, var, VAh, Wh, and varh were changed from their original NAESB values of vA, w, vAr, vAH, wH, and vArH to make them consistent with the standard SI abbreviations for these units. The enumerated values of Btu and BtuPerh were changed from their original NAESB values of btu and btuPerH to make them consistent with the standard abbreviations for British thermal units and hours.

Note: The abbreviation MBtu is often used to represent thousands of Btu's (where M represents the Roman numeral for 1000). The FSGIM *always* uses the SI prefixes listed in the SiScaleCodeType enumeration and not the prefixes based on Roman numerals. This means that 1000 Btu's would be represented in the FSGIM as EnergyThermalQuantity.quantity = 1, EnergyThermalType.itemUnits = Btu, and EnergyThermalType.siScaleCode = k, *not* EnergyThermalType.siScaleCode = M.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\UnitSymbolKind.*

Table 6.11 – UnitSymbolKind Enumerated Values

Enumerated Value	Definition
m	Length, meter
g	Mass, gram
a	Current, ampere
degK	Temperature, Kelvin (Note: the unit "degrees" is implied)
mol	Amount of substance, mole
cd	Luminous intensity, candela

Enumerated Value	Definition
s	Time, second
rad	Plane angle, Radian (m/m)
sr	Solid angle, Steradian (m ² /m ²)
gy	Absorbed dose, Gray (J/kg)
bq	Radioactivity, Becquerel (1/s)
degC	Relative temperature, degrees Celsius
sv	Dose equivalent, Sievert (J/kg)
f	Electric capacitance, Farad (C/V)
c	Electric charge, Coulomb (Amp second)
h	Electric inductance, Henry (Wb/A)
v	Electric potential, Volt (W/A)
ohm	Electric resistance, Ohm (V/A)
J	Energy joule, (N · m = C · V = W · s)
n	Force newton, (kg m/s ²)
hz	Frequency, Cycles per second or (1/s)
lx	Illuminance lux, (lm/m ²)
lm	Luminous flux, lumen (cd sr)
wb	Magnetic flux, Weber (V s)
t	Magnetic flux density, Tesla (Wb/m ²)
W	Real power, Watt. By definition, one Watt equals one joule per second. Electrical power may have real and reactive components. The real portion of electrical power (I ² R) or V Icos ² , is expressed in Watts. (See also apparent power and reactive power.)
pa	Pressure, Pascal (N/m ²) (Note: the absolute or relative measurement of pressure is implied with this entry. See below for more explicit forms.)
siemens	Electric Conductance, Siemens (A / V = 1 / Ω)
paA	Pressure, Pascal, absolute pressure
paG	Pressure, Pascal, gauge pressure
kat	Catalytic activity, katal = mol / s
revPerS	Rotational speed, rotations per second (Note: compare to cycles per second, Hz)
m2	Area, square meter
m3	Volume, cubic meter
mPerS	Velocity, meter per second (m/s)
mPerS2	Acceleration, meter per second squared
m3PerS	Volumetric flow rate, cubic meters per second
mPerM3	Fuel efficiency, meter / cubic meter
gM	Moment of mass, kilogram meter (kg · m) (first moment of mass) (Note: users must supply the "k" prefix to obtain "kg m".)
gPerM3	Density, gram/cubic meter (Note: users must supply the prefix multiplier "k" to form kg/ m ³)
m2PerS	Viscosity, meter squared / second
wPerMK	Thermal conductivity, Watt/meter Kelvin

Enumerated Value	Definition
jPerK	Heat capacity, Joule/Kelvin
radPerS	Angular velocity, radians per second
VA	Apparent power, Volt Ampere (See also real power and reactive power.)
var	Reactive power, Volt Ampere reactive. The "reactive" or "imaginary" component of electrical power (VISin?). (See also real power and apparent power).
vS	Volt seconds, Volt seconds (W/s/A)
v2	Volts squared, Volt squared (W2/A2)
aS	Amp seconds, Amp seconds
a2	Amps squared, Amp squared
a2S	Amps squared time, square Amp second
VAh	Apparent energy, Volt Ampere hours
Wh	Real energy, Watt hours
varh	Reactive energy, Volt Ampere reactive hours
vPerHz	Magnetic flux, Volt per Hertz
hzPerS	Rate of change of frequency, Hertz per second
gM2	Moment of mass, kg m2 (Second moment of mass, commonly called the moment of inertia) (Note: users must supply the "k" prefix to obtain "kg m2".)
wPerS	Ramp rate, Watt per second
lPerS	Volumetric flow rate, liters per second
q	Quantity power, Q
qh	Quantity energy, Qh
ohmM	resistivity, Ohm meter, ? (rho)
aPerM	A/m, magnetic field strength, Ampere per meter
v2H	volt-squared hour, Volt-squared-hours
a2H	ampere-squared hour, Ampere-squared hour
aH	Ampere-hours, Ampere-hours
wHPerM3	Wh/m3, energy per volume
timeStamp	Timestamp, time and date per ISO 8601 format
wHPerRev	Kh-Wh, active energy metering constant
vArHPerRev	Kh-VArh, reactive energy metering constant
vAHPerRev	Kh-Vah, apparent energy metering constant
m3PerH	Volumetric flow rate, cubic meter per hour
m3CompensatedPerH	Volumetric flow rate, compensated cubic meter per hour
m3UncompensatedPerH	Volumetric flow rate, uncompensated cubic meter per hour
lPerH	Volumetric flow rate, liter per hour
lUncompensatedPerH	Volumetric flow rate, liter (uncompensated) per hour
lCompensatedPerH	Volumetric flow rate, liter (compensated) per hour
q45	Quantity power, Q measured at 45°

Enumerated Value	Definition
q60	Quantity power, Q measured at 60°
q45H	Quantity energy, Qh measured at 45°
q60H	Quantity energy, Qh measured at 60°
jPerKg	Specific energy, Joule / kg
m3Uncompensated	Volume, cubic meter, with the value uncompensated for weather effects.
m3Compensated	Volume, cubic meter, with the value compensated for weather effects.
m1	Wavenumber, reciprocal meter, (1/m)
m3PerKg	Specific volume, cubic meter per kilogram, v
paS	Dynamic viscosity, Pascal second
nM	Moment of force, Newton meter
nPerM	Surface tension, Newton per meter
radPerS2	Angular acceleration, radian per second squared
wPerM2	Heat flux density, irradiance, Watt per square meter
jPerKgK	Specific heat capacity, specific entropy, Joule per kilogram kelvin
jPerM3	energy density, Joule per cubic meter
vPerM	electric field strength, Volt per meter
cPerM3	electric charge density, Coulomb per cubic meter
cPerM2	surface charge density, Coulomb per square meter
fPerM	permittivity, Farad per meter
hPerM	permeability, Henry per meter
jPerMol	molar energy, Joule per mole
jPerMolK	molar entropy, molar heat capacity, Joule per mole kelvin
cPerKg	exposure (x rays), Coulomb per kilogram
gyPerS	absorbed dose rate, Gray per second
wPerSr	radiant intensity, Watt per steradian
wPerM2Sr	radiance, Watt per square meter steradian
katPerM3	catalytic activity concentration, katal per cubic meter
min	Time, minute = 60 s
hr	Time, hour = 60 min = 3600 s
d	Time, day = 24 h = 86400 s
deg	Plane angle, degree
angleMin	Plane angle, minute
angleSec	Plane angle, second
ha	Area, hectare
l	Volume, liter = dm ³ = m ³ /1000.
tonne	mass, "tonne" or "metric ton" (1000 kg = 1 Mg)
none	N/A, none (not applicable)
cosTheta	Power factor, dimensionless

Enumerated Value	Definition
bel	Logarithmic ratio, Bel, Note: users must combine this unit with the multiplier prefix "d" to form decibels (dB)
status	State, status, where: "1" = "true", "live", "on", "high", "set", "0" = "false", "dead", "off", "low", "cleared" Note: A Boolean value is preferred but other values may be supported
count	Amount of substance, Counter value
bm	Logarithmic ratio of signal strength, Bel-mW, normalized to 1mW. Note: to form "dBm" combine "Bm" with multiplier "d".
code	Application Value, Encoded value
meCode	EndDeviceEvent, Value to be interpreted as a EndDeviceEventCode
lPerL	Concentration, The ratio of the volume of a solute divided by the volume of the solution. (Note: Users may need use a prefix such as 'μ' to express a quantity such as 'μL/L')
gPerG	Concentration, The ratio of the mass of a solute divided by the mass of the solution. (Note: Users may need use a prefix such as 'μ' to express a quantity such as 'μg/g')
molPerM3	Concentration, The amount of substance concentration, (c), the amount of solvent in moles divided by the volume of solution in m ³ .
molPerMol	Concentration, Molar fraction, the ratio of the molar amount of a solute divided by the molar amount of the solution.
molPerKg	Concentration, Molality, the amount of solute in moles and the amount of solvent in kilograms.
mPerM	Length, Ratio of length
sPerS	Time, Ratio of time (Note: Users may need to supply a prefix such as 'μ' to show rates such as 'μs/s')
hzPerHz	Frequency, Rate of frequency change (Note: Users may need to supply a prefix such as 'm' to show rates such as 'mHz/Hz')
vPerV	Voltage, Ratio of voltages (Note: Users may need to supply a prefix such as 'm' to show rates such as 'mV/V')
aPerA	Current, Ratio of Amperages (Note: Users may need to supply a prefix such as 'm' to show rates such as 'mA/A')
wPerVA	Power factor, PF
rev	Amount of rotation, Revolutions
wPerW	Signal Strength, Ratio of power (Note: Users may need to supply a prefix such as 'm' to show rates such as 'mW/W')
refractiveIndexN	Refractive Index, n
relativePermeabilityMur	Relative Permeability, μ _r
np	Logarithmic ratio, neper
eV	energy, electronvolt (1 eV = 1.602 176 x 10 ⁻¹⁹ J)
da	mass, dalton (1 Da = 1.660 538 x 10 ⁻²⁷ kg)
u	mass, unified atomic mass unit (1u = 1 Da)
ua	length, astronomical unit (1ua = 1.495 978 x 10 ¹¹ m)
c0	speed, natural unit of speed (speed of light in a vacuum) 299 792 458 m/s
nuH	action, natural unit of action (reduced planck constant) 1.054 571 X 10 ⁻³⁴ J s
nuMe	mass, natural unit of mass (electron mass) 9.109 382 x 10 ⁻³¹ kg
nuHPertNuMeC02	time, natural unit of time
auE	charge, atomic units of charge (elementary charge) 1.602 176 x 10 ⁻¹⁹ C
auMe	mass, atomic units of mass (electron mass) 9.109 382 x 10 ⁻³¹ kg
auH	action, atomic unit of action (reduced planck constant) 1.054 571 X 10 ⁻³⁴ J s
auA0	length, atomic unit of length, bohr (Bohr radius) 0.529 177 x 10 ⁻¹⁰ m
auEh	energy, atomic unit of energy, hartree 4.359 744

Enumerated Value	Definition
auHPerAuEh	time, atomic unit of time
char	Number of characters, characters
charPerSec	Data rate, characters per second
money	Monetary unit, Generic money (Note: Specific monetary units are identified the currency class).
ft3	Volume, cubic foot
ft3Compensated	Volume, cubic foot compensated for weather
ft3Uncompensated	Volume, cubic foot uncompensated for weather
ft3PerH	Volumetric flow rate, cubic foot per hour
ft3CompensatedPerH	Volumetric flow rate, compensated cubic feet per hour
ft3UncompensatedPerH	Volumetric flow rate, uncompensated cubic feet per hour
uSGal	Volume, US gallon (1 gal = 231 in ³ = 128 fl oz.)
uSGalPerH	Volumetric flow rate, US gallon per hour
impGal	Volume, Imperial gallon
impGalPerH	Volumetric flow rate, Imperial gallon per hour
Btu	Energy, British Thermal Unit ref: 59 degrees F
BtuPerH	Power, BTU per hour
psiA	Pressure, Pound per square inch, absolute
psiG	Pressure, Pound per square inch, gauge
lUncompensated	Volume, Liter, with the value uncompensated for weather effects.
lCompensated	Volume, Liter, with the value compensated for weather effects.
therm	Energy, Therm
bar	Pressure, bar (1 bar = 100 kPa)
mmHg	Pressure, millimeter of mercury (1 mmHg ~ 133.3 Pa)
angstrom	Length, ångström (1 Å = 10 ⁻¹⁰ m)
nmi	Length, nautical mile (1 M = 1852 m)
barn	Area, barn (1 b = 100 fm ² = 10 ⁻²⁸ m ²)
kn	Speed, knot (1 kn = 1852/3600) m/s
ci	Activity, curie (1 Ci = 3.7 x 10 ¹⁰ Bq)
r	exposure, roentgen (1 R = 2.58 x 10 ⁻⁴ C/kg)
doseRad	absorbed dose, rad (1 rd = 1 cGy)
rem	dose equivalent, rem (1 rem = 10 ⁻² Sv)
in	length, inch
ft	length, foot (1 ft = 12 in)
rod	length, rod (1 rod = 16.5 ft)
fur	length, furlong (1 fur = 660 ft)
mi	length, mile (1 statute mile = 8 fur = 80 chains = 320 rods = 5280 feet)
ft2	area, square foot (1 ft2 = 144 in2)
yd2	area, square yard (1 yd2 = 9 ft2)

Enumerated Value	Definition
rod2	area, square rod (1 rod2 = 272.25 ft2)
acre	area, acre (1 acre = 160 rd2 = 43 560 ft2)
mi2	area, square mile (1 mi2 = 640 acres)
sectionOfLand	area, section of land (1 mi2 = 1 section of land)
township	area, township (1 township = 36 sections of land)
yd3	Volume, cubic yard (1 yd3 = 27 ft3)
li	length, link (1 li = 0.66 ft)
ch	length, chain (1 ch = 100 links = 66 ft)
usLiqPt	Volume, US liquid pint (1 pt = 28.875 in3 = 128 fl dr)
usLiqQt	Volume, US liquid quart (1 qt = 2 pt)
flDrAp	Volume, Apothecaries fluid dram
flOzAp	Volume, Apothecaries fluid ounce (1 fl oz ap = 8 fl dr ap)
usDryPt	Volume, US dry pint (1 pt = 67.2 in3)
usDryQt	Volume, US dry quart (1 qt = 2 pints)
usPk	Volume, US peck (1 pk = 8 qt)
usBu	Volume, US bushel (1 bu = 4 pk)
gr	Mass, grain (1 grain = 1/7000 avdp lb) Note: The "grain" is the same in the avoirdupois, troy, and apothecaries units of mass.
avdpDr	Mass, Avoirdupois dram (1 Avdp dr = 27-1/32 gr) Note: The abbreviation "dr" may be used if there is no chance of confusing the avoirdupois dram with the troy or apothecaries dram.
avdpOz	Mass, Avoirdupois ounce (1 Avdp oz = 16 Avdp dr) Note: The abbreviation "oz" may be used if there is no chance of confusing the avoirdupois ounce with the troy or apothecaries ounce.
avdpLb	Mass, Avoirdupois pound (1 lb = 16 oz) Note: Although the term "pound" is commonly used in many countries simply as "pound (lb)", the historical term "avoirdupois" is added to provide context and distinguish it in this setting from the troy and apothecaries systems of measure which use the same names. The abbreviation "lb" may be used instead of "avdp lb" if no confusion is possible with the pound named in these other systems.
shortCwt	Mass, Avoirdupois hundredweight (1 cwt = 100 lbs)
shortTon	Mass, Avoirdupois ton (1 short ton = 2000 lbs)
longCwt	Mass, Avoirdupois gross or long hundredweight (1 long cwt = 112 lbs)
longTon	Mass, Avoirdupois gross or long ton (1 long ton = 20 long cwt)
dwt	Mass, Troy pennyweight (1 dwt = 24 grains)
ozT	Mass, Troy ounce (1 oz t = 20 dwt)
lbT	Mass, Troy pound (1 lb t = 12 oz t)
sAp	Mass, Apothecaries scruple (1 s ap = 20 grains)
drAp	Mass, Apothecaries dram (1 dr ap = 3 s ap)
ozAp	Mass, Apothecaries ounce (1 oz ap = 8 dr ap)
lbAp	Mass, Apothecaries pound (1 lb ap = 12 oz ap)
mpgImp	Fuel economy, mile per imperial gallon
mpgUS	Fuel economy, mile per US gallon

Enumerated Value	Definition
mPGeUS	Fuel economy, mile per US gallon equivalent
lPer100km	Fuel economy, liter per 100 km
wHPeMi	Fuel economy, watt-hour per mile (Note: users must supply the "k" prefix to create "kWh/mi")
wHPe100Mi	Fuel economy, watt-hour per 100 mile (Note: users must supply the "k" prefix to create "kWh/(100 mi)")
degF	Temperature, degrees Fahrenheit
erg	Energy, erg (1 erg = 10 ⁻⁷ J)
dyn	Force, dyne (1 dyn = 10 ⁻⁵ N)
p	Dynamic viscosity, poise (1 P = 0.1 Pa s)
st	Kinematic viscosity, stokes (1 St = 1 cm ² /s)
sb	Luminance, stilb (1 sb = 10 ⁴ cd/m ²)
ph	Illuminance, phot (1 ph = 10 ⁴ lx)
gal	Acceleration, gal (1 Gal = 10 ⁻² m/s ²) Note: This "Gal" is an abbreviation for "Galileo" not "gallon"
mx	Magnetic flux, Maxwell (1 Mx = 10 ⁻⁸ Wb)
gauss	Magnetic flux density, Gauss (1 G = 10 ⁻⁴ T)
oe	Magnetic field, Oersted (1 Oe = (103/4π) A/m)
gPerS	Mass flow rate, g/s

6.6.3. Elements Defined in the Primitive Data Types Model

The following elements are defined in this model.

6.6.3.1. AnyType (PrimitiveType)

This is a placeholder type for which any explicit data type can be substituted.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types\AnyType*.

6.6.3.2. AnyURI (PrimitiveType)

An AnyURI primitive data type is meant to identify something. It has a restricted syntax that follows RFC 2396 as amended by RFC 2732. It is often (but not necessarily) dereferencable to a URL.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types\AnyURI*.

6.6.3.3. Boolean (PrimitiveType)

This primitive data type represents a binary value with allowable values of "true" and "false."

See <http://www.omg.org/spec/UML/20110701/PrimitiveTypes.xmi#Boolean>.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types\Boolean*.

6.6.3.4. GlobalId (PrimitiveType)

This primitive data type represents a globally unique identifier based on RFC 4122. Note that UUIDs may be constructed from name-based systems (Version 5) using the procedure in Section 4.3 of the referenced RFC.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types\GlobalId*.

6.6.3.5. Integer (PrimitiveType)

This primitive data type represents whole numbers ranging from minus infinity to plus infinity.
See <http://www.omg.org/spec/UML/20110701/PrimitiveTypes.xmi#Integer>.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types\Integer*.

6.6.3.6. Real (PrimitiveType)

This primitive data type represents the continuous range of numbers from minus infinity to plus infinity.
See <http://www.omg.org/spec/UML/20110701/PrimitiveTypes.xmi#Real>.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types\Real*.

6.6.3.7. String (PrimitiveType)

This primitive data type represents a contiguous sequence of characters.
See <http://www.omg.org/spec/UML/20110701/PrimitiveTypes.xmi#String>.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Primitive Data Types\String*.

6.6.4. Elements Defined in the Measurements Model

The following elements are defined in this model.

6.6.4.1. Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*.

Table 6.12 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
timeReference	Applicable time reference associated with this Measurement. The duration, end, and start can be populated as needed to describe the time reference of the measurement. If only a timestamp is necessary then omit duration and end. If the measurement occurred over a period, use the duration attribute to indicate this period. Only two of the three attributes may be included. If the Measurement is a MeasurementSet the contained MeasurementQuantity will not populate timeReference.	UTCDateTimeInterval See Clause 6.6.1.4.	[0..1]

This element has the following connections to other elements:

Table 6.13 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measuredAt	The logical reference for a point of measurement.	EMUsagePoint See Clause 5.6.7.8.	[1]

6.6.4.2. MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

Parent Class: Measurement (See Clause 6.6.4.1.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet.*

6.6.4.3. ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType.*

6.6.4.4. MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

Parent Class: ItemBaseType (See Clause 6.6.4.3.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*.

Table 6.14 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
accumulationCharacteristic	Accumulation behaviour of a reading over time, usually 'measuringPeriod', to be used with individual endpoints (as opposed to 'macroPeriod' and 'aggregate' that are used to describe aggregations of data from individual endpoints).	AccumulationKind See Clause 5.7.5.2.1.35.	[0..1]
dataQualifier	Salient attribute of the reading data aggregated from individual endpoints. This is mainly used to define a mathematical operation carried out over 'macroPeriod', but may also be used to describe an attribute of the data when the 'macroPeriod' is not defined.	DataQualifierKind See Clause 5.7.5.2.1.37.	[0..1]
description	User defined text describing this measurement or its usage.	String See Clause 6.6.3.7.	[0..1]
itemDescription	Pre-defined name for this type of measurement as specified by the derived class.	String See Clause 6.6.3.7.	[1]
itemUnits	Units for the measurement, not including any multiplier.	UnitSymbolKind See Clause 6.6.2.6.	[1]
measurementQuality	Quality of the measurement if other than "good."	QualityOfReading See Clause 6.6.2.5.	[0..1]
resolution	The granularity of the measurement, corresponding to the smallest increment reportable by a measurement instrument. Constraint: resolution > 0.	Real See Clause 6.6.3.6.	[0..1]
siScaleCode	SI multiplier that applies to the unit of measure contained in the itemUnits enumeration.	SiScaleCodeType See Clause 6.6.2.3.	[1]

6.6.4.5. MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

Parent Class: Measurement (See Clause 6.6.4.1.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*.

Table 6.15 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
value	Quantity of the measurement.	Real See Clause 6.6.3.6.	[1]
uncertainty	A positive number which represents the possible positive or negative deviation from the claimed measured value. If this attribute is not defined, a default value may be used as defined by local policy.	Real See Clause 6.6.3.6.	[1]

This element has the following connections to other elements:

Table 6.16 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with a MeasurementQuantity.	MeasurementMetadata Type See Clause 6.6.4.4.	[1]

6.6.4.6. Elements Defined in the Power Measurements Model

The following elements are defined in this model.

6.6.4.6.1. PowerApparentQuantityRuleSet (Class)

- The aggregation of apparent power measurements is calculated by first separately calculating the aggregate real power and aggregate reactive power. Using these two results, take the square root of the sum of their squares for the quantity of the resulting apparent power.
- The aggregate absolute uncertainty is calculated as the square-root of the sum of the squares of the aggregate absolute real power uncertainty and the aggregate absolute reactive power uncertainty.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model*\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerApparentQuantityRuleSet.

6.6.4.6.2. PowerCurve (Class)

This class describes the characteristics of a mathematical function used to estimate the power consuming characteristics of a load or the power generating characteristics of a generator.

UML Element Location: *Model*\Common Primitive Types, Classes, and Enumerations\Measurements\PowerMeasurements\PowerCurve.

Table 6.17 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
realPowerCurve	This attribute defines the real component of a single piecewise linear curve mapping the percentage of power consumed by the device as a function of the present level of operation of the device.	PiecewiseLinearSegment See Clause 6.6.5.5.	[0..*]
reactivePowerCurve	This attribute defines the reactive component of a single piecewise linear curve mapping the percentage of power consumed by the device as a function of the present level of operation of the device.	PiecewiseLinearSegment See Clause 6.6.5.5.	[0..*]
maximumRealPower	This attribute defines the maximum real power consumed (or supplied) by the device in units specified in PowerRealType. Constraint: The maximumRealPower attribute has a multiplicity of 0..1 to allow for the modeling of loads with an unknown demand such as some plug loads. In cases where the	PowerRealQuantity See Clause 6.6.4.6.15.	[0..1]

Attribute Name	Description	Attribute Type	Multiplicity
maximumReactivePower	maximum demand is known, the maximumRealPower attribute shall be required. This attribute defines the maximum reactive power consumed (or supplied) by the device in units specified in PowerReactiveType. Constraint: If reactivePowerCurve is specified, then the maximumReactivePower attribute must be specified.	PowerReactiveQuantity See Clause 6.6.4.6.13.	[0..1]

6.6.4.6.3. PowerMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of PowerMeasurementSets, calculate the sum or difference of its PowerRealQuantity, PowerReactiveQuantity, and PowerApparentQuantity parts according to the rules for each of those classes.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations/Measurements/Power Measurements/PowerMeasurementRuleSet.*

6.6.4.6.4. PowerRatings (Class)

This class describes the power characteristics of a Load (or Generator) component. The attributes defined allow specifying the minimum and maximum expected power draw from the load (supply from a generator). It also allows a series of predefined operation power curves to be defined with one designated as presently being operational.

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations/Measurements/Power Measurements/PowerRatings.*

Table 6.18 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
adjustedNoDRPower	This attribute defines the maximum expected power draw of a load (or the minimum power supplied by a generator) during operation. This value differs from the rated power since it may take into account operational considerations such as environmental, equipment safety or regulatory conditions.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
adjustedFullIDRPower	This attribute defines the minimum expected power draw of a load (or the maximum power supplied by a generator) during operation. This value differs from the rated power since it may take into account operational considerations such as environmental, equipment safety or regulatory conditions.	PowerMeasurementsSet See Clause 6.6.4.6.11.	[1]
powerCurves	This attribute defines one or more piecewise linear curves mapping the percentage of power consumed by the device as a function of the present level of operation of the device. Many loads draw power (or generators supply power) based on the present loading characteristics of the device. For example, a motor driving a fan will draw more power as the fan blade pitch is increased.	PowerCurve See Clause 6.6.4.6.2.	[0..*]

Attribute Name	Description	Attribute Type	Multiplicity
	The axes of the curve are defined in percent to allow loads of any type to utilize the attribute. For example a simple 60W incandescent light bulb would likely be modeled by a linear segment (0,0), (100,100). Here as the bulb is energized (dimmed) from off (0%) to fully on (100%), the power needed to energize the device also travels from 0% to 100%. A more complicated device such as a room air-conditioner may have a non-linear power curve. Here, as the coolness setting is adjusted from warmest to coolest, the air conditioner will draw relative more power when set to maximum cooling than when set to minimum cooling. When powerCurve is not present, the load or generator is assumed to be a two state device drawing no power when the device is off and adjustedNoDRPower when the device is on. When adjustedNoDRPower also is not present, the load or generator is assumed to be a two state device drawing no power when the device is off and maximumRealPower when the device is on.		
activePowerCurve	This attribute defines the index into the zero based array of powerCurves indicating which powerCurve is presently active. . Constraint: This attribute must be present if the powerCurves attribute is present.	Integer See Clause 6.6.3.5.	[0..1]

6.6.4.6.5. PowerReactiveQuantityRuleSet (Class)

- The aggregation of reactive power measurements is calculated by treating inductive loads and generators that consume vars as positive quantities and capacitive loads and generators that supply vars as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveQuantityRuleSet.*

6.6.4.6.6. PowerRealQuantityRuleSet (Class)

- The aggregation of real power measurements is calculated by treating loads as positive quantities and generators as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealQuantityRuleSet.*

6.6.4.6.7. **PowerThermalQuantityRuleSet (Class)**

- The aggregation of thermal power measurements is calculated by converting the individual PowerThermalQuantities so that they have a consistent PowerThermalType.itemUnits (W or BtuPerh) and then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerThermalQuantityRuleSet.*

6.6.4.6.8. **PowerApparentQuantity (Class)**

Quantity and pair used to represent apparent power measurements.

Parent Class: PowerQuantityType (See Clause 6.6.4.6.12.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerApparentQuantity.*

This element has the following connections to other elements:

Table 6.19 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	apparentPowerAggregation Rules	The ruleset that is associated with aggregating more than one PowerApparentQuantity.	PowerApparentQuantityRuleSet See Clause 6.6.4.6.1.	[1]
Association	measurementMetadata	The metadata associated with a PowerApparentQuantity. (This association redefines the measurementMetadata association between the PowerQuantityType Class (see Clause 6.6.4.6.12) and the PowerItemType Class (see Clause 6.6.4.6.10).)	PowerApparentType See Clause 6.6.4.6.9.	[1]

6.6.4.6.9. PowerApparentType (Class)

Metadata associated with measurements of apparent power.

Constraint: itemDescription = "ApparentPower"

Constraint: itemUnits = VA

Parent Class: PowerItemType (See Clause 6.6.4.6.10.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations* \Power Measurements\PowerApparentType.

6.6.4.6.10. PowerItemType (Abstract Class)

Metadata associated with a power measurement.

Parent Class: MeasurementMetadataType (See Clause 6.6.4.4.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations* \Power Measurements\PowerItemType.

6.6.4.6.11. PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

Parent Class: MeasurementSet (See Clause 6.6.4.2.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations* \Power Measurements\PowerMeasurementsSet.

This element has the following connections to other elements:

Table 6.20 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	powerAggregationRules	The ruleset that is associated with aggregating more than one PowerMeasurementSet.	PowerMeasurementRuleSet See Clause 6.6.4.6.3.	[1]
Composition	quantityApparentPower	The apparent power portion of the PowerMeasurementSet.	PowerApparentQuantity See Clause 6.6.4.6.8.	[0..1]
Composition	quantityReactivePower	The reactive power portion of the PowerMeasurementSet.	PowerReactiveQuantity See Clause 6.6.4.6.13.	[1]
Composition	quantityRealPower	The real power portion of the PowerMeasurementSet.	PowerRealQuantity See Clause 6.6.4.6.15.	[1]

6.6.4.6.12. PowerQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of power measurements.

Parent Class: MeasurementQuantity (See Clause 6.6.4.5.)

UML Element Location: *Model|Common Primitive Types, Classes, and Enumerations|Measurements|Power Measurements|PowerQuantityType*.

This element has the following connections to other elements:

Table 6.21 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with a PowerQuantityType. (This association redefines the measurementMetadata association between the MeasurementQuantity Class (see Clause 6.6.4.5) and the MeasurementMetadataType Class (see Clause 6.6.4.4).)	PowerItemType See Clause 6.6.4.6.10.	[1]

6.6.4.6.13. PowerReactiveQuantity (Class)

Quantity and pair used to represent reactive power measurements.

Parent Class: PowerQuantityType (See Clause 6.6.4.6.12.)

UML Element Location: *Model|Common Primitive Types, Classes, and Enumerations|Measurements|Power Measurements|PowerReactiveQuantity*.

This element has the following connections to other elements:

Table 6.22 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with a PowerReactiveQuantity. (This association redefines the measurementMetadata association between the PowerQuantityType Class (see Clause 6.6.4.6.12) and the PowerItemType Class (see Clause 6.6.4.6.10).)	PowerReactiveType See Clause 6.6.4.6.14.	[1]
Association	reactivePowerAggregationRules	The ruleset that is associated with aggregating more than one PowerReactiveQuantity.	PowerReactiveQuantity RuleSet See Clause 6.6.4.6.5.	[1]

6.6.4.6.14. PowerReactiveType (Class)

Metadata associated with measurements of reactive power.

Constraint: itemDescription = "ReactivePower"

Constraint: itemUnits = VAr

Parent Class: PowerItemType (See Clause 6.6.4.6.10.)

UML Element Location: *Model*|*Common Primitive Types, Classes, and Enumerations*|*Measurements*|*Power Measurements*|*PowerReactiveType*.

6.6.4.6.15. PowerRealQuantity (Class)

Quantity and metadata pair used to represent real power measurements.

Parent Class: PowerQuantityType (See Clause 6.6.4.6.12.)

UML Element Location: *Model*|*Common Primitive Types, Classes, and Enumerations*|*Measurements*|*Power Measurements*|*PowerRealQuantity*.

This element has the following connections to other elements:

Table 6.23 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with a PowerRealQuantity. (This association redefines the measurementMetadata association between the PowerQuantityType Class (see Clause 6.6.4.6.12) and the PowerItemType Class (see Clause 6.6.4.6.10).)	PowerRealType See Clause 6.6.4.6.16.	[1]
Association	realPowerAggregationRules	The ruleset that is associated with aggregating more than one PowerRealQuantity.	PowerRealQuantityRuleSet See Clause 6.6.4.6.6.	[1]

6.6.4.6.16. PowerRealType (Class)

Metadata associated with measurements of real power.
Constraint: itemDescription = "RealPower"
Constraint: itemUnits = W

Parent Class: PowerItemType (See Clause 6.6.4.6.10.)

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Measurements* | *Power Measurements* | *PowerRealType*.

6.6.4.6.17. PowerThermalQuantity (Class)

Quantity/metadata pair used to represent thermal power measurements.

Parent Class: PowerQuantityType (See Clause 6.6.4.6.12.)

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Measurements* | *Power Measurements* | *PowerThermalQuantity*.
This element has the following connections to other elements:

Table 6.24 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with a PowerThermalQuantity. (This association redefines the measurementMetadata association between the PowerQuantityType Class (see Clause 6.6.4.6.12) and the PowerItemType Class (see Clause 6.6.4.6.10).)	PowerThermalType See Clause 6.6.4.6.18.	[1]
Association	thermalPowerAggregationRules	The ruleset that is associated with aggregating more than one PowerThermalQuantity.	PowerThermalQuantity RuleSet See Clause 6.6.4.6.7.	[1]

6.6.4.6.18. PowerThermalType (Class)

Metadata associated with measurements of thermal power.
Constraint: itemDescription = "ThermalPower"
Constraint: itemUnits = W, BtuPerh

Parent Class: PowerItemType (See Clause 6.6.4.6.10.)

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Measurements* | *Power Measurements* | *PowerThermalType*.

6.6.4.7. Elements Defined in the Energy Measurements Model

The following elements are defined in this model.

6.6.4.7.1. ThermalEnergyMeasurementRuleSet (Class)

The aggregation of thermal energy measurements is calculated by normalizing the units of measurement, then calculating their arithmetic sum.

The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\ThermalEnergyMeasurementRuleSet.*

6.6.4.7.2. EnergyApparentQuantityRuleSet (Class)

- The aggregation of apparent energy measurements is calculated by first separately calculating the aggregate real energy and aggregate reactive energy. Using these two results, take the square root of the sum of their squares for the quantity of the resulting apparent energy.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the aggregate absolute real energy uncertainty and the aggregate absolute reactive energy uncertainty.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentQuantityRuleSet.*

6.6.4.7.3. EnergyMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EnergyMeasurementSets, calculate the sum or difference of its EnergyRealQuantity, EnergyReactiveQuantity, and EnergyApparentQuantity parts according to the rules for each of those classes.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementRuleSet.*

6.6.4.7.4. EnergyReactiveQuantityRuleSet (Class)

- The aggregation of reactive energy measurements is calculated by treating inductive loads and generators that consume var-hours as positive quantities and capacitive loads and generators that supply var-hours as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveQuantityRuleSet.*

6.6.4.7.5. EnergyRealQuantityRuleSet (Class)

- The aggregation of real energy measurements is calculated by treating loads as positive quantities and generators as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantityRuleSet.*

6.6.4.7.6. EnergyApparentQuantity (Class)

Quantity/metadata pair used to represent apparent energy measurements.

Parent Class: EnergyQuantityType (See Clause 6.6.4.7.10.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentQuantity.*

This element has the following connections to other elements:

Table 6.25 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	apparentEnergyAggregation Rules	The ruleset that is associated with aggregating more than one EnergyApparentQuantity.	EnergyApparentQuantityRuleSet See Clause 6.6.4.7.2.	[1]
Association	measurementMetadata	The metadata associated with an EnergyApparentQuantity. (This association redefines the measurementMetadata association between the EnergyQuantityType Class (see Clause 6.6.4.7.10) and the EnergyItemType Class (see Clause 6.6.4.7.8).)	EnergyApparentType See Clause 6.6.4.7.7.	[1]

6.6.4.7.7. EnergyApparentType (Class)

Metadata associated with measurements of apparent energy.

Constraint: itemDescription = "ApparentEnergy"

Constraint: itemUnits = VAh

Parent Class: EnergyItemtype (See Clause 6.6.4.7.8.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentType*.

6.6.4.7.8. EnergyItemType (Abstract Class)

Metadata associated with an energy measurement.

Parent Class: MeasurementMetadataType (See Clause 6.6.4.4.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyItemType*.

6.6.4.7.9. EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

Parent Class: MeasurementSet (See Clause 6.6.4.2.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*.

This element has the following connections to other elements:

Table 6.26 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	energyAggregationRules	The ruleset that is associated with aggregating more than one EnergyMeasurementSet.	EnergyMeasurementRuleSet See Clause 6.6.4.7.3.	[1]
Composition	quantityApparentEnergy	The apparent energy portion of the EnergyMeasurementSet.	EnergyApparentQuantity See Clause 6.6.4.7.6.	[0..1]
Composition	quantityReactiveEnergy	The reactive energy portion of the EnergyMeasurementSet.	EnergyReactiveQuantity See Clause 6.6.4.7.11.	[1]
Composition	quantityRealEnergy	The real energy portion of the EnergyMeasurementSet.	EnergyRealQuantity See Clause 6.6.4.7.13.	[1]

6.6.4.7.10. EnergyQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of energy measurements.

Parent Class: MeasurementQuantity (See Clause 6.6.4.5.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyQuantityType*.

This element has the following connections to other elements:

Table 6.27 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with an EnergyQuantityType. (This association redefines the measurementMetadata association between the MeasurementQuantity Class (see Clause 6.6.4.5) and the MeasurementMetadataType Class (see Clause 6.6.4.4).)	EnergyItemType See Clause 6.6.4.7.8.	[1]

6.6.4.7.11. EnergyReactiveQuantity (Class)

Quantity/metadata pair used to represent reactive energy measurements.

Parent Class: EnergyQuantityType (See Clause 6.6.4.7.10.)

UML Element Location: *Model|Common Primitive Types, Classes, and Enumerations|Measurements|Energy Measurements|EnergyReactiveQuantity.*

This element has the following connections to other elements:

Table 6.28 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with an EnergyReactiveQuantity. (This association redefines the measurementMetadata association between the EnergyQuantityType Class (see Clause 6.6.4.7.10) and the EnergyItemType Class (see Clause 6.6.4.7.8).)	EnergyReactiveType See Clause 6.6.4.7.12.	[1]
Association	reactiveEnergyAggregation Rules	The ruleset that is associated with aggregating more than one EnergyReactiveQuantity.	EnergyReactiveQuantityRuleSet See Clause 6.6.4.7.4.	[1]

6.6.4.7.12. EnergyReactiveType (Class)

Metadata associated with measurements of reactive energy.

Constraint: itemDescription = "ReactiveEnergy"

Constraint: itemUnits = VArh

Parent Class: EnergyItemType (See Clause 6.6.4.7.8.)

UML Element Location: *Model|Common Primitive Types, Classes, and Enumerations|Measurements|Energy Measurements|EnergyReactiveType.*

6.6.4.7.13. **EnergyRealQuantity (Class)**

Quantity/metadata pair used to represent real energy measurements.

Parent Class: StoreableEnergyQuantity (See Clause 6.6.4.7.17.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantity*.
This element has the following connections to other elements:

Table 6.29 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with an <u>EnergyRealQuantity</u> . (This association redefines the measurementMetadata association between the <u>EnergyQuantityType</u> Class (see Clause 6.6.4.7.10) and the <u>EnergyItemType</u> Class (see Clause 6.6.4.7.8).)	<u>EnergyRealType</u> See Clause 6.6.4.7.14.	[1]
Association	realEnergyAggregationRules	The ruleset that is associated with aggregating more than one <u>EnergyRealQuantity</u> .	<u>EnergyRealQuantityRuleSet</u> See Clause 6.6.4.7.5.	[1]

6.6.4.7.14. **EnergyRealType (Class)**

Metadata associated with measurements of real energy.
Constraint: itemDescription = "RealEnergy"
Constraint: itemUnits = Wh

Parent Class: EnergyItemType (See Clause 6.6.4.7.8.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealType*.

6.6.4.7.15. **EnergyThermalQuantity (Class)**

Quantity/metadata pair used to represent thermal energy measurements.

Parent Class: StoreableEnergyQuantity (See Clause 6.6.4.7.17.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyThermalQuantity*.

This element has the following connections to other elements:

Table 6.30 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with an EnergyThermalQuantity. (This association redefines the measurementMetadata association between the EnergyQuantityType Class (see Clause 6.6.4.7.10) and the EnergyItemType Class (see Clause 6.6.4.7.8).)	EnergyThermalType See Clause 6.6.4.7.16.	[1]
Association	thermalEnergyAggregation Rules	The ruleset that is associated with aggregating more than one EnergyThermalQuantity.	ThermalEnergyMeasurementRuleSet See Clause 6.6.4.7.1.	[1]

6.6.4.7.16. EnergyThermalType (Class)

Metadata associated with measurements of thermal energy.

Constraint: itemDescription = "ThermalEnergy"

Constraint: itemUnits = Wh, J, Btu, or therm

Parent Class: EnergyItemType (See Clause 6.6.4.7.8.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations* \Measurements\EnergyMeasurements\EnergyThermalType.

6.6.4.7.17. StoreableEnergyQuantity (Abstract Class)

Quantity used to represent real electrical or thermal energy.

Parent Class: EnergyQuantityType (See Clause 6.6.4.7.10.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations* \Measurements\EnergyMeasurements\StoreableEnergyQuantity.

6.6.4.8. Elements Defined in the Emissions Measurements Model

The following elements are defined in this model.

6.6.4.8.1. EmissionsConcentrationQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by concentration. Note: The determination of which instances of EmissionsConcentrationQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

Parent Class: EmissionsQuantityType (See Clause 6.6.4.8.13.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations* \Measurements\EmissionsMeasurements\EmissionsConcentrationQuantity.

This element has the following connections to other elements:

Table 6.31 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	emissionsConcentrationAggregationRules	The ruleset that is associated with aggregating more than one EmissionsConcentrationQuantity.	EmissionsConcentrationQuantityRuleSet See Clause 6.6.4.8.2.	[1]
Association	measurementMetadata	The metadata associated with an EmissionsConcentrationQuantity. (This association redefines the measurementMetadata association between the EmissionsQuantityType Class (see Clause 6.6.4.8.13) and the EmissionsItemType Class (see Clause 6.6.4.8.4).)	EmissionsConcentrationType See Clause 6.6.4.8.3.	[1]

6.6.4.8.2. EmissionsConcentrationQuantityRuleSet (Class)

Particulate matter shall not be included in standard aggregations of emissions. This shall not preclude its inclusion in custom aggregations of emissions.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Emissions Measurements\EmissionsConcentrationQuantityRuleSet.*

6.6.4.8.3. EmissionsConcentrationType (Class)

Metadata associated with an emissions measurement whose quantity indicates the concentration of the emission.

Constraint: itemUnits = gPerM3

Parent Class: EmissionsItemType (See Clause 6.6.4.8.4.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsConcentrationType.*

6.6.4.8.4. EmissionsItemType (Abstract Class)

Metadata associated with an emissions measurement.

Parent Class: MeasurementMetadataType (See Clause 6.6.4.4.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsItemType.*

6.6.4.8.5. EmissionsMassQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass. Note: The determination of which instances of EmissionsMassQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

Parent Class: EmissionsQuantityType (See Clause 6.6.4.8.13.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassQuantity*.
This element has the following connections to other elements:

Table 6.32 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	emissionsMassAggregation Rules	The ruleset that is associated with aggregating more than one EmissionsMassQuantity.	EmissionsMassQuantityRuleSet See Clause 6.6.4.8.6.	[1]
Association	measurementMetadata	The metadata associated with an EmissionsMassQuantity. (This association redefines the measurementMetadata association between the EmissionsQuantityType Class (see Clause 6.6.4.8.13) and the EmissionsItemType Class (see Clause 6.6.4.8.4).)	EmissionsMassType See Clause 6.6.4.8.10.	[1]

6.6.4.8.6. EmissionsMassQuantityRuleSet (Class)

- For each available instance of EmissionsMassQuantity where the values of EmissionsMassQuantity.measurementData.itemDescription match, sum each available EmissionsMassQuantity.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassQuantityRuleSet*.

6.6.4.8.7. EmissionsMassRateQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass rate. Note: The determination of which instances of EmissionsMassRateQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

Parent Class: EmissionsQuantityType (See Clause 6.6.4.8.13.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassRateQuantity*.

This element has the following connections to other elements:

Table 6.33 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	emissionsMassAggregation Rules	The ruleset that is associated with aggregating more than one EmissionsMassQuantity.	EmissionsMassRateRuleSet	[1]
Association	measurementMetadata	The metadata associated with an EmissionsMassQuantity. (This association redefines the measurementMetadata association between the EmissionsQuantityType Class (see Clause 6.6.4.8.13) and the EmissionsItemType Class (see Clause 6.6.4.8.4).)	EmissionsMassRateType	[1]

6.6.4.8.8. EmissionsMassRateRuleSet (Class)

- For each available instance of EmissionsMassRateQuantity where the values of EmissionsMassRateQuantity.measurementData.itemDescription match, sum each available EmissionsMassRateQuantity.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Measurements* | *Emissions Measurements* | *EmissionsMassRateRuleSet*.

6.6.4.8.9. EmissionsMassRateType (Class)

Metadata associated with an emissions measurement whose quantity indicates the rate of mass of the emission.

Constraint: itemUnits = gPerS

Parent Class: EmissionsItem Type (See Clause 6.6.4.8.4.)

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Measurements* | *Emissions Measurements* | *EmissionsMassRateType*.

6.6.4.8.10. EmissionsMassType (Class)

Metadata associated with an emissions measurement whose quantity indicates the total mass of the emission.

Constraint: itemUnits = g

Parent Class: EmissionsItem Type (See Clause 6.6.4.8.4.)

UML Element Location: *Model* | *Common Primitive Types, Classes, and Enumerations* | *Measurements* | *Emissions Measurements* | *EmissionsMassType*.

6.6.4.8.11. EmissionsMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EmissionsMeasurementSets, calculate the sum or difference of each emissions quantity individually according to the rules for the applicable EmissionsMassQuantity.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Emissions Measurements\EmissionsMeasurementRuleSet.*

6.6.4.8.12. EmissionsMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions quantity measurement that could be made for a given location and time.

Parent Class: MeasurementSet (See Clause 6.6.4.2.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Emissions Measurements\EmissionsMeasurementsSet.*

This element has the following connections to other elements:

Table 6.34 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	emissionsAggregationRules	The ruleset that is associated with aggregating more than one EmissionsMeasurementRuleset.	EmissionsMeasurementRuleSet See Clause 6.6.4.8.11.	[1]
Composition	quantityCH4	Quantity of methane emissions. Constraint: The itemDescription of the associated measurementMetadata must be "CH4Emissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityCO	Quantity of carbon monoxide emissions. Constraint: The itemDescription of the associated measurementMetadata must be "COEmissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityCO2	Quantity of carbon dioxide emissions. Constraint: The itemDescription of the associated measurementMetadata must be "CO2Emissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityCO2e	Quantity of carbon dioxide equivalent emissions. Carbon dioxide equivalent emissions are defined as the sum of the mass emissions of each individual greenhouse gas adjusted for its global warming potential. Constraint: The itemDescription of the associated measurementMetadata must be "CO2eEmissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityHg	Quantity of mercury emissions. Constraint: The itemDescription of the associated measurementMetadata must be "HgEmissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Composition	quantityNO2	Quantity of emissions of nitrogen dioxide. This is often used as a surrogate for NOx. Constraint: The itemDescription of the associated measurementMetadata must be "NO2Emissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityO3	Quantity of ozone emissions. Constraint: The itemDescription of the associated measurementMetadata must be "O3Emissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityPb	Quantity of lead emissions. Constraint: The itemDescription of the associated measurementMetadata must be "PbEmissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityPFC	Quantity of perfluorocarbon emissions as specified by the authority having jurisdiction. Constraint: The itemDescription of the associated measurementMetadata must be "PFC Emissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantityPM10	Concentration of inhalable particulate matter 10 micrometers and smaller. Constraint: The itemDescription of the associated measurementMetadata must be "PM10Emissions".	EmissionsConcentrationQuantity See Clause 6.6.4.8.1.	[0..1]
Composition	quantityPM2_5	Concentration of inhalable particulate matter 2.5 micrometers and smaller. Constraint: The itemDescription of the associated measurementMetadata must be "PM2_5Emissions".	EmissionsConcentrationQuantity See Clause 6.6.4.8.1.	[0..1]
Composition	quantitySF6	Quantity of sulfur hexafluoride emissions. Constraint: The itemDescription of the associated measurementMetadata must be "SF6Emissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]
Composition	quantitySO2	Quantity of sulfur dioxide emissions. This is often used as a surrogate for SOx. Constraint: The itemDescription of the associated measurementMetadata must be "SO2Emissions".	EmissionsMassQuantity See Clause 6.6.4.8.5.	[0..1]

6.6.4.8.13. EmissionsQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of emissions measurements.

Parent Class: MeasurementQuantity (See Clause 6.6.4.5.)

UML Element Location: *Model|Common Primitive Types, Classes, and Enumerations|Measurements|Emissions Measurements|EmissionsQuantityType.*

This element has the following connections to other elements:

Table 6.35 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	measurementMetadata	The metadata associated with an EmissionsQuantityType. (This association redefines the measurementMetadata association between the MeasurementQuantity Class (see Clause 6.6.4.5) and the MeasurementMetadataType Class (see Clause 6.6.4.4).)	EmissionsItemType See Clause 6.6.4.8.4.	[1]

6.6.4.8.14. EmissionsRateMeasurementRuleSet (Class)

- In order to calculate the sum or difference of EmissionsRateMeasurementSets, calculate the sum or difference of each emissions rate individually according to the rules for the applicable EmissionsMassRateQuantity.

Parent Class: RuleSet (See Clause 5.6.7.11.5.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsRateMeasurementRuleSet.*

6.6.4.8.15. EmissionsRateMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions rate measurement that could be made for a given location and time.

Parent Class: MeasurementSet (See Clause 6.6.4.2.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsRateMeasurementsSet.*

This element has the following connections to other elements:

Table 6.36 - Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	emissionsAggregationRules	The ruleset that is associated with aggregating more than one EmissionsMeasurementRuleset.	EmissionsRateMeasurementRuleSet See Clause 6.6.4.8.14.	[1]
Composition	rateCH4	Rate of methane emissions. Constraint: The itemDescription of the associated measurementMetadata must be "CH4Emissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	rateCO	Rate of carbon monoxide emissions. Constraint: The itemDescription of the associated measurementMetadata must be "COEmissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]

Connection Type	Role Name	Description	Role Type	Multiplicity
Composition	rateCO2	Rate of carbon dioxide emissions. Constraint: The itemDescription of the associated measurementMetadata must be "CO2Emissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	rateCO2e	Rate of carbon dioxide equivalent emissions. Carbon dioxide equivalent emissions are defined as the sum of the mass emissions of each individual greenhouse gas adjusted for its global warming potential. Constraint: The itemDescription of the associated measurementMetadata must be "CO2eEmissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	rateHg	Rate of mercury emissions. Constraint: The itemDescription of the associated measurementMetadata must be "HgEmissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	rateO3	Rate of ozone emissions. Constraint: The itemDescription of the associated measurementMetadata must be "O3Emissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	ratePb	Rate of lead emissions. Constraint: The itemDescription of the associated measurementMetadata must be "PbEmissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	ratePFC	Rate of perfluorocarbon emissions as specified by the authority having jurisdiction. Constraint: The itemDescription of the associated measurementMetadata must be "PFC Emissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	rateSF6	Rate of sulfur hexafluoride emissions. Constraint: The itemDescription of the associated measurementMetadata must be "SF6Emissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]
Composition	rateSO2	Rate of sulfur dioxide emissions. Constraint: The itemDescription of the associated measurementMetadata must be "SO2Emissions".	EmissionsMassRateQuantity See Clause 6.6.4.8.7.	[0..1]

6.6.5. Elements Defined in the Other Common Classes Model

The following elements are defined in this model.

6.6.5.1. Circuit (Class)

This class is used to model electrical circuits. All ConnectionPoints that are electrically tied together will point to the same instance of this class.

Parent Class: FSGIMIdentifiedObject (See Clause 5.2.3.4.)

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\Circuit.*

6.6.5.2. ConnectionPoint (Class)

This class is used to model the physical connections from Generators, Loads, Meters, and Energy Routers to sources or sinks of energy.

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Other Common Classes\ConnectionPoint*.

This element has the following connections to other elements:

Table 6.37 – Class Connections

Connection Type	Role Name	Description	Role Type	Multiplicity
Association	connectedTo	The Circuit to which this ConnectionPoint is electrically connected.	Circuit See Clause 6.6.5.1.	[0..1]
Association	logicalEMUsagePoint	Logical point on a network at which consumption or production for this ConnectionPoint is either measured (e.g., metered) or estimated (e.g., unmetered street lights).	EMUsagePoint See Clause 5.6.7.8.	[0..1]

6.6.5.3. GridCircuit (Class)

This class is used to model electrical circuits that are part of the utility grid.

Parent Class: Circuit (See Clause 6.6.5.1.)

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Other Common Classes\GridCircuit*.

6.6.5.4. MonetaryQuantity (Class)

A monetary quantity used to express the cost or value of something in some currency.

UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations\Other Common Classes\MonetaryQuantity*.

Table 6.38 – Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
quantity	The amount of currency for this monetary quantity.	Real See Clause 6.6.3.6.	[1]
currency	The code indicating the type of currency for this monetary quantity. Constraint: The values shall be codes in the ISO 4217 currency code list.	CurrencyKind See Clause 6.6.2.1.	[1]

6.6.5.5. PiecewiseLinearSegment (Class)

The PiecewiseLinearSegment class defines the attributes needed to specify a single straight line segment for a piecewise linear curve. Each straight line segment is specified by two X-axis coordinates, percentOFFFullRatedOutputBegin and percentOFFFullRatedOutputEnd; and by two Y-axis coordinates, percentOFFFullRatedInputPowerDrawnBegin and percentOFFFullRatedInputPowerDrawnEnd.

The entire piecewise linear curve is defined by the runningProfile attribute; where the 'percent of full rated input power' is a function of the 'percent of full rated output'. That is, as the output varies between 0 and 100 percent, the function maps to the percentage of input power (0..100) required to achieve the specified output.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\PiecewiseLinearSegment.*

Table 6.39 - Class Attributes

Attribute Name	Description	Attribute Type	Multiplicity
desiredFractionOffFullRatedOutputBegin	This attribute defines the starting x-coordinate of the straight line segment. Constraint: This attribute's value shall be greater than percentOFFFullRatedOutputEnd of the previous segment. Constraint: This attribute's value shall be greater than zero and less than or equal to 100	Real See Clause 6.6.3.6.	[1]
desiredFractionOffFullRatedOutputEnd	This attribute defines the ending x-coordinate of the straight line segment. Constraint: This attribute's value shall be greater than percentOFFFullRatedOutputBegin of this segment. Constraint: This attribute's value shall be greater than zero and less than or equal to 100	Real See Clause 6.6.3.6.	[1]
requiredFractionOffFullRatedInputPowerDrawnBegin	This attribute defines the starting y-coordinate of the straight line segment. Constraint: This attribute's value shall be greater than zero and less than or equal to 100	Real See Clause 6.6.3.6.	[1]
requiredFractionOffFullRatedInputPowerDrawnEnd	This attribute defines the ending y-coordinate of the straight line segment. Constraint: This attribute's value shall be greater than zero and less than or equal to 100	Real See Clause 6.6.3.6.	[1]

7. CONFORMANCE REQUIREMENTS

7.1. Introduction

The Facility Smart Grid Information Model (FSGIM) is a standard specifying an abstract model of energy information within a facility. This model is intended to be the basis for other new and existing standards containing architectures and communication protocols that support conventional or smart grid devices and energy services such as energy management. The FSGIM is not intended to replace the many existing standards that address grid devices, services, and energy management in buildings, homes, and industrial facilities. It is intended to foster interoperability between energy service providers, energy management systems in facilities, and energy consuming devices in facilities.

The FSGIM abstract model is specified by a set of conformance blocks that each includes a set of classes, attributes and behaviors as specified in Clause 7.3. Conformance blocks are intended to contain a concrete set of related functionality within a model component.

The users of this standard are expected to be standards developing organizations (accredited national, regional, and international standards bodies) and standards setting organizations (associations that develop specifications). For a target standard to conform to the FSGIM it may support the entire set of conformance blocks or a subset of the conformance blocks.

A diagram of the FSGIM conformance method is shown in Figure 7.1. It illustrates how a target standard claiming conformance to the FSGIM maps its features into the FSGIM. Each FSGIM conformance block, shown as a UML diagram, defines a subset of the FSGIM model to which a target standard may elect to conform. A target standard may support a conformance exactly as shown or may further constrain the FSGIM as specified in Clause 7.2.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

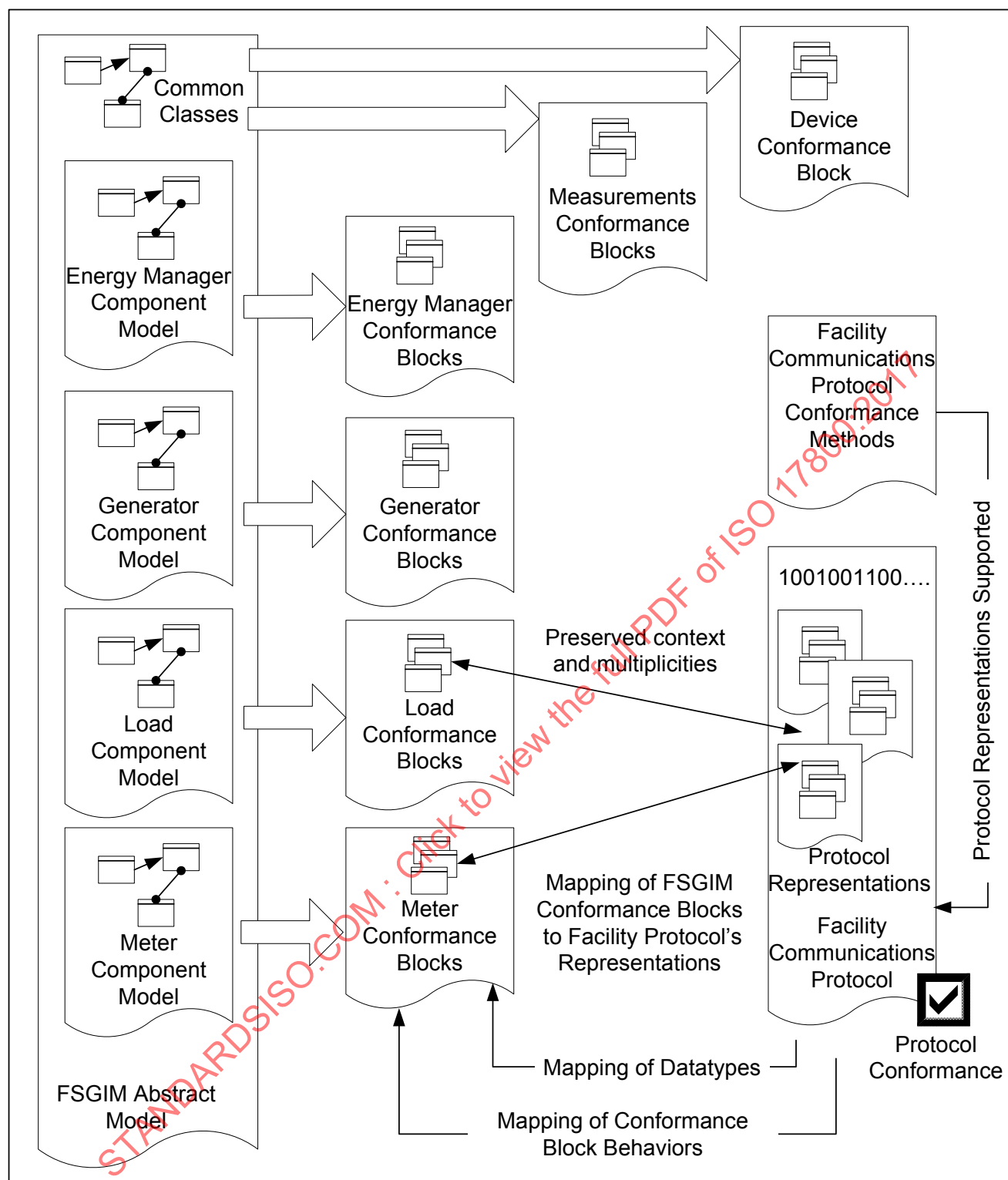


Figure 7.1 - FSGIM Conformance Requirements

A target standard claiming conformance to the FSGIM shall specify the conformance blocks to which it conforms. Each FSGIM conformance block, shown as a UML diagram, defines a subset of the FSGIM model to which a target standard may elect to conform. A target standard may support a conformance block exactly as shown or may further constrain the conformance block as specified in Clause 7.2.1 and 7.2.2.

7.2. Conformance Requirements

For a target standard to conform to the FSGIM it shall make publicly available the following information:

- profiles of one more conformance blocks as specified in Clause 7.3;
- primitive data types are represented;

- attribute mapping of FSGIM attributes to the target standard representations;
- mapping of conformance block behaviors;
- enumeration constraints;
- handling of unreliable/unavailable information

In the event that a supported information is unreliable or unavailable, it shall be ignored (not used) in calculations or in aggregations. When communicating unreliable or unavailable information, a target standard shall document how the information is identified.

A profile consists of a conformance block diagram in XMI format indicating:

- multiplicity of all supported attributes;
- multiplicity of all supported associations;
- any additional constraints on the conformance block diagram.

7.2.1. Conformance to FSGIM Primitive Data Types

Documentation for the target standard shall show explicitly how the FSGIM primitive data types are mapped into the specific data types employed by the target standard. This document shall also show, either explicitly or by reference to other published standards, any restrictions on the range or resolution of the FSGIM data types that are imposed by the target standard.

7.2.2. Attributes

A target standard shall document how each required and optional attribute in each conformance block is represented in the target standard. This documentation shall clearly state which optional attributes in the conformance block are required, optional, or not used in the target standard. The range or resolution of the attributes may have changed due to the documented limitations of the target standard, but the meaning of the attributes shall not change.

A target standard may use implied values for conformance block attributes provided that the meaning of these attributes can be unambiguously derived from the target standard. Documentation demonstrating conformance shall clearly show how this is done.

The contextual hierarchy of data and multiplicity shall correspond to that in FSGIM. The multiplicity mapping from FSGIM to a target protocol shall be documented. The mapping shall be bidirectional such that attributes of the conformance block, when translated to a target standard and then translated back to the FSGIM from the target standard, shall retain the original meaning.

NOTE: Example of multiplicity mapping to a target protocol: if FSGIM states that an attribute is 1..*, one protocol might use * = 10000 and another might use * = 10 due to memory limitations.

7.2.3. Enumerations

A target standard shall document all enumerations referenced by its selected conformance blocks. Normative enumerated values of the enumerations defined within FSGIM are immutable; they shall not be deleted or reordered.

7.2.3.1. FSGIM Enumerations

The target standard may elect to extend the set of FSGIM enumerated values. These additions must be fully documented (in the same manner as the FSGIM enumerated values) describing the enumerated value, its description and its use.

7.2.3.2. Target Standard Enumerations

A target standard may define new enumerations referenced by adding target standard attributes. These new enumerations must be fully documented (in the same manner as the FSGIM enumerated values) describing the enumerated value, its description and its use.

7.2.4. Associations

A target standard shall document how each required and optional association in each conformance block is represented in the target standard. This documentation shall clearly state which optional associations in the conformance block are required, optional, or not used in the target standard.

The contextual hierarchy of data and multiplicity shall correspond to that in FSGIM. The multiplicity mapping from FSGIM to a target protocol shall be documented. The mapping shall be bidirectional such that associations of the conformance block, when translated to a target standard and then translated back to the FSGIM from the target standard, shall retain the original meaning.

7.2.5. Behaviors

Documentation for the target standard shall show how the target standard satisfies the behaviors for the conformance blocks to which it conforms, where there is an implied behavior.

7.2.6. Additional Information

These requirements shall not be interpreted to prevent the inclusion of additional information by a target standard when incorporating a conformance block. However, a target standard shall clearly document that this additional information is outside the scope of the FSGIM.

7.2.7. Compliance

The method chosen for ensuring conformance to the target standard shall be specified to ensure that a device implementation can demonstrate compliance with the FSGIM conformance blocks chosen.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

7.3. Conformance Blocks

Conformance blocks are functional groupings of classes, attributes and behaviors that can be selected by target protocol standards to indicate which portions of the FSGIM are supported.

UML Package Location: *Model\Conformance Blocks*

7.3.1. FSGIM Primitive Conformance Block

Conformance blocks in this clause describe the basic set of data type primitives used throughout the FSGIM.

UML Package Location: *Model\Conformance Blocks\FSGIM Primitive Conformance Block*

7.3.1.1. FSGIM Primitive Conformance Block Diagram

This diagram depicts the complete set of primitive blocks from which the target standard may be profiled. These primitives are immutable and cannot be profiled by the target standard. However, since these data types do not contain any processor specific or programming language specific restrictions on their range or resolution; constraints may be imposed on these data types to allow for application specific constraints, provided that the target standard clearly defines these restrictions.

The target standard must select any and all the primitive blocks that are referenced by any class defined within the target standard that references these primitives either directly or indirectly. The target standard may select primitives not presently required but may be referenced in the future.

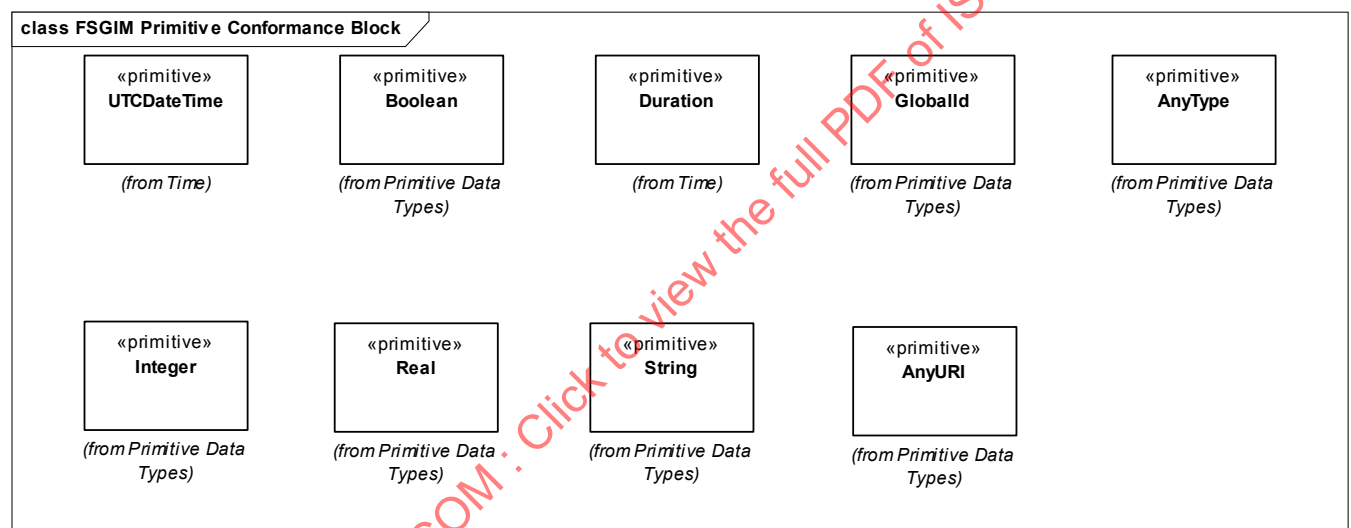


Figure 7.2 - FSGIM Primitive Conformance Block

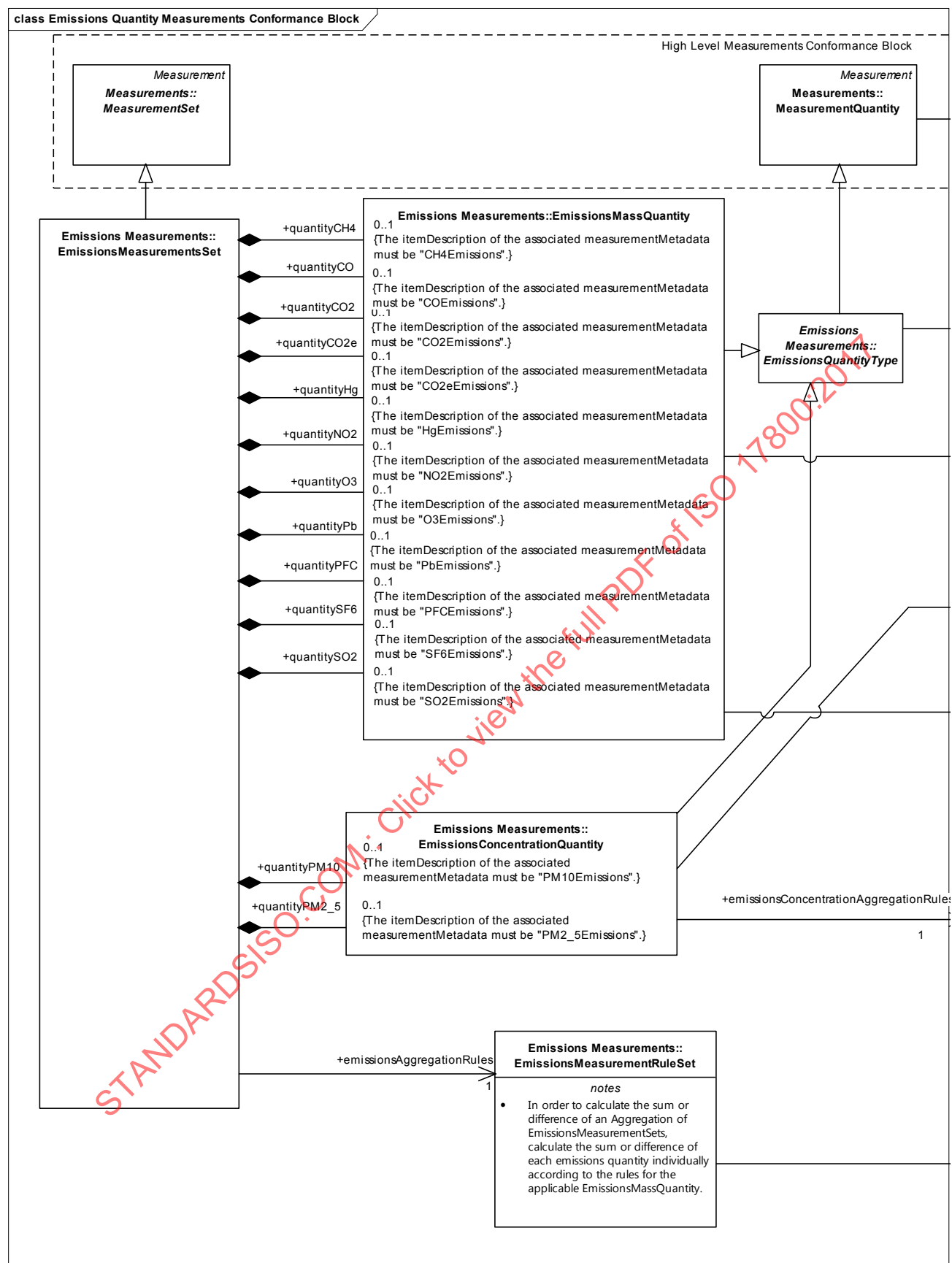
7.3.2. Measurement Conformance Blocks

Conformance blocks in this clause address power, energy, and emissions measurements that are used in the Meter, Load, Generator and Energy Manager Components.

UML Package Location: *Model\Conformance Blocks\Measurement Conformance Blocks*

7.3.2.1. Emissions Quantity Measurements Conformance Block Diagram

This diagram depicts the conformance blocks, classes, and attributes used to define emissions measurements.



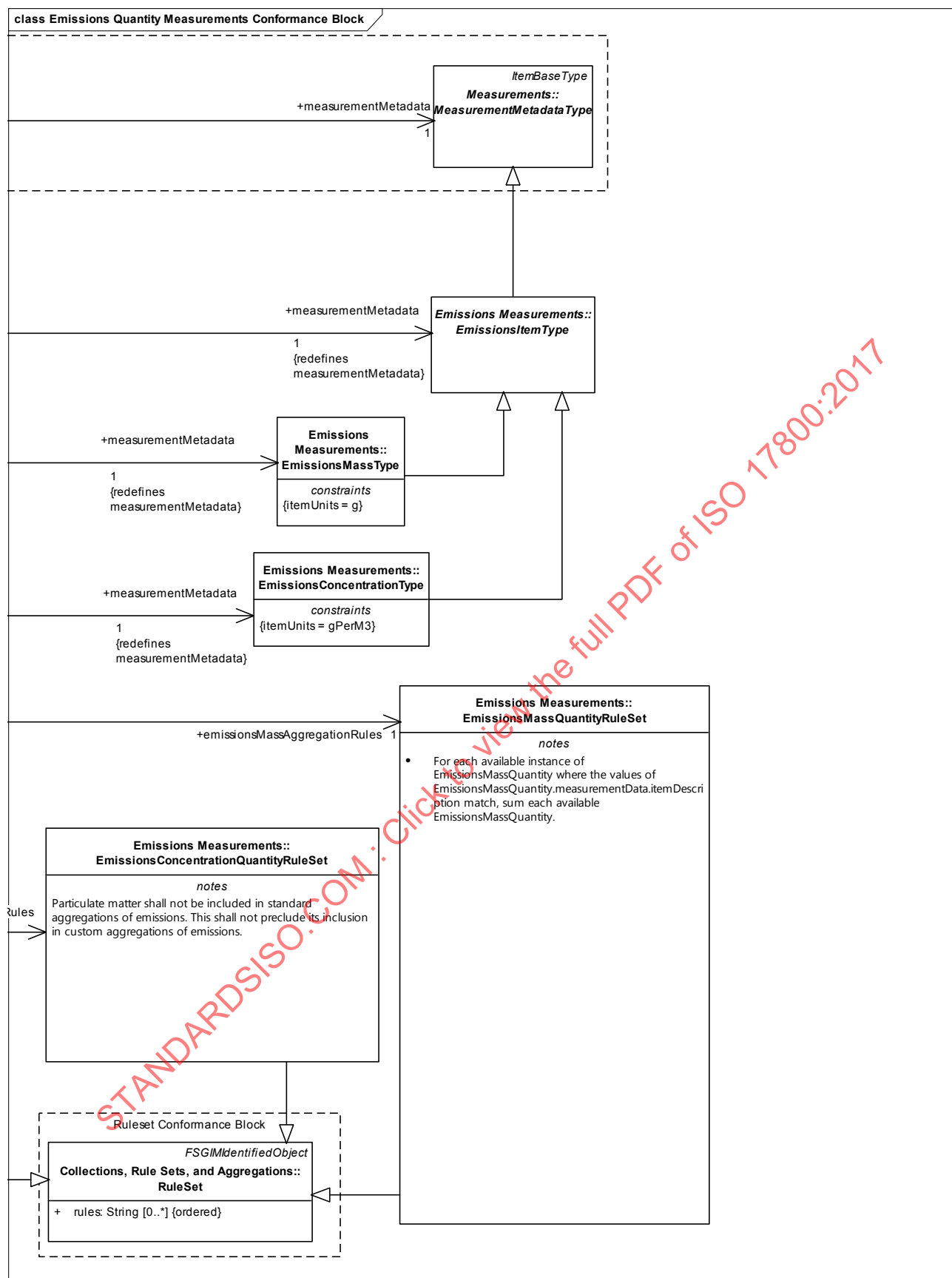


Figure 7.3 - Emissions Quantity Measurements Conformance Block

The Emissions Quantity Measurements Conformance Block Diagram makes use of the following elements:

EmissionsConcentrationQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by concentration. Note: The determination of which instances of EmissionsConcentrationQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationQuantity*. See Clause 6.6.4.8.1 for the complete definition of this element.

EmissionsConcentrationQuantityRuleSet (Class)

Particulate matter shall not be included in standard aggregations of emissions. This shall not preclude its inclusion in custom aggregations of emissions.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationQuantityRuleSet*. See Clause 6.6.4.8.2 for the complete definition of this element.

EmissionsConcentrationType (Class)

Metadata associated with an emissions measurement whose quantity indicates the concentration of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsConcentrationType*. See Clause 6.6.4.8.3 for the complete definition of this element.

EmissionsItemType (Abstract Class)

Metadata associated with an emissions measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsItemType*. See Clause 6.6.4.8.4 for the complete definition of this element.

EmissionsMassQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass. Note: The determination of which instances of EmissionsMassQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassQuantity*. See Clause 6.6.4.8.5 for the complete definition of this element.

EmissionsMassQuantityRuleSet (Class)

- For each available instance of EmissionsMassQuantity where the values of EmissionsMassQuantity.measurementData.itemDescription match, sum each available EmissionsMassQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassQuantityRuleSet*. See Clause 6.6.4.8.6 for the complete definition of this element.

EmissionsMassType (Class)

Metadata associated with an emissions measurement whose quantity indicates the total mass of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMassType*. See Clause 6.6.4.8.10 for the complete definition of this element.

EmissionsMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EmissionsMeasurementSets, calculate the sum or difference of each emissions quantity individually according to the rules for the applicable EmissionsMassQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsMeasurementRuleSet*. See Clause 6.6.4.8.11 for the complete definition of this element.

EmissionsMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions quantity measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurementsSet*. See Clause 6.6.4.8.12 for the complete definition of this element.

EmissionsQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of emissions measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\EmissionsMeasurements\EmissionsQuantityType*. See Clause 6.6.4.8.13 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

7.3.2.2. High Level Measurements Conformance Block Diagram

All measurements of power, energy, and emissions within the FSGIM include both a quantity, which represents the magnitude of the thing being measured, and a set of metadata, which relates to that measurement. The metadata includes information such as the units of the measurement, the accuracy and the precision of the measurement.

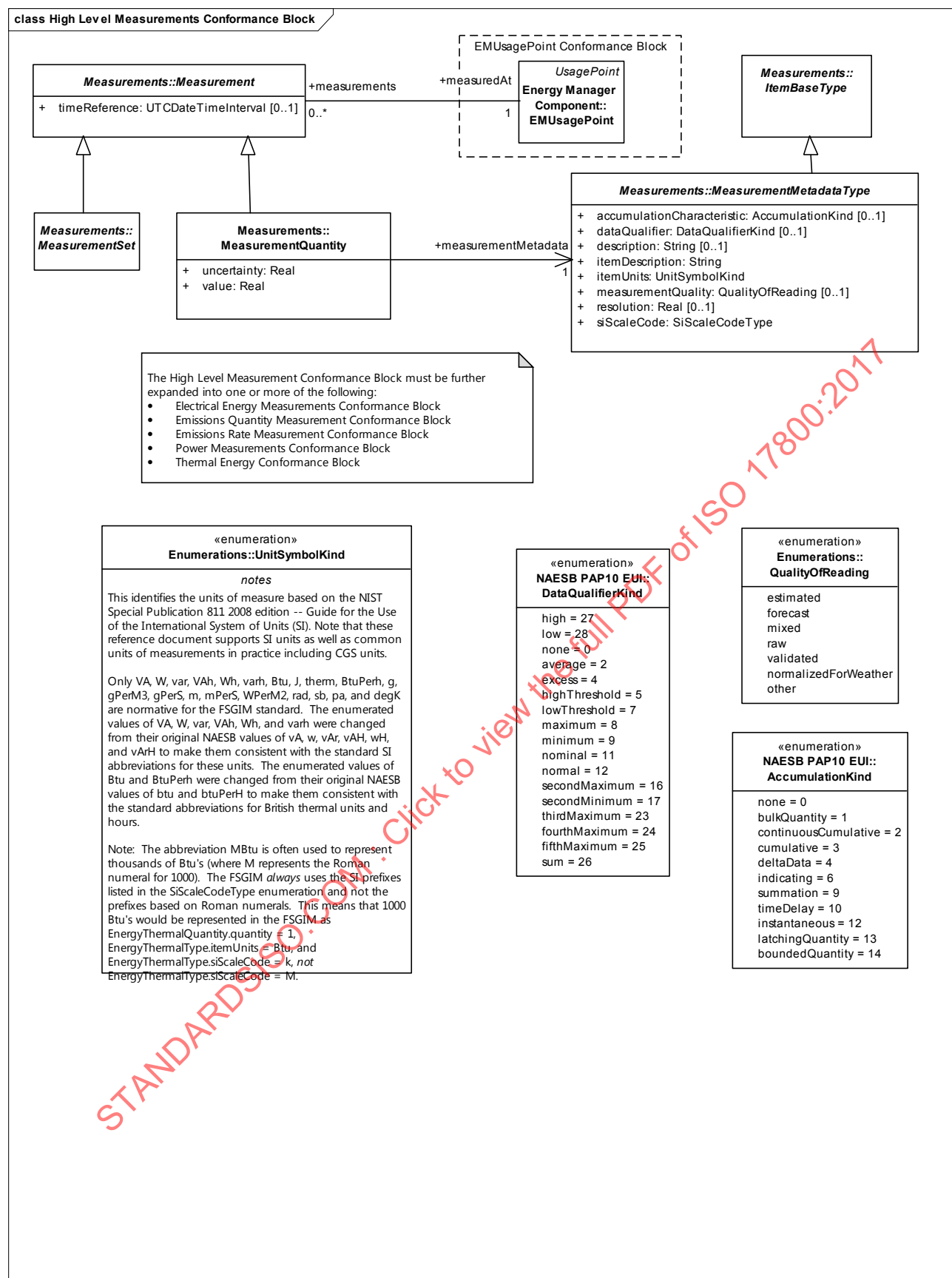


Figure 7.4 - High Level Measurements Conformance Block

The High Level Measurements Conformance Block Diagram makes use of the following elements:

AccumulationKind (Enumeration)

This attribute provides metadata about how the measurements described are made. These enumerated values refer to the common present and historic methods by which physical measurements are made in revenue-quality metering instruments and are based on that terminology.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\AccumulationKind. See Clause 5.7.5.2.1.35 for the complete definition of this element.

DataQualifierKind (Enumeration)

This value qualifies the measurement as to what it represents -- for example a minimum or maximum value or a nominal or nameplate value.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\DataQualifierKind. See Clause 5.7.5.2.1.37 for the complete definition of this element.

EMUsagePoint (Class)

Logical point on a network at which consumption or production is either measured (e.g., metered) or estimated (e.g., unmetered street lights).

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMUsagePoint*. See Clause 5.6.7.8 for the complete definition of this element.

ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType*. See Clause 6.6.4.3 for the complete definition of this element.

Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*. See Clause 6.6.4.1 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and*

Enumerations\Measurements\MeasurementMetadataType. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and*

Enumerations\Measurements\MeasurementQuantity. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

QualityOfReading (Enumeration)

List of codes indicating the quality of the reading.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Enumerations\QualityOfReading*. See Clause 6.6.2.5 for the complete definition of this element.

UnitSymbolKind (Enumeration)

This identifies the units of measure based on the NIST Special Publication 811 2008 edition -- Guide for the Use of the International System of Units (SI). Note that these reference document supports SI units as well as common units of measurements in practice including CGS units.

Only VA, W, var, VAh, Wh, varh, Btu, J, therm, BtuPerh, g, gPerM3, gPerS, m, mPerS, WPerM2, rad, sb, pa, and degK are normative for the FSGIM standard. The enumerated values of VA, W, var, VAh, Wh, and varh were changed from their original NAESB values of vA, w, vAr, vAH, wH, and vArH to make them consistent with the standard SI abbreviations for these units. The enumerated values of Btu and BtuPerh were changed from their original NAESB values of btu and btuPerH to make them consistent with the standard abbreviations for British thermal units and hours.

Note: The abbreviation MBtu is often used to represent thousands of Btu's (where M represents the Roman numeral for 1000). The FSGIM *always* uses the SI prefixes listed in the SiScaleCodeType enumeration and not the prefixes based on Roman numerals. This means that 1000 Btu's would be represented in the FSGIM as EnergyThermalQuantity.quantity = 1, EnergyThermalType.itemUnits = Btu, and EnergyThermalType.siScaleCode = k, *not* EnergyThermalType.siScaleCode = M.

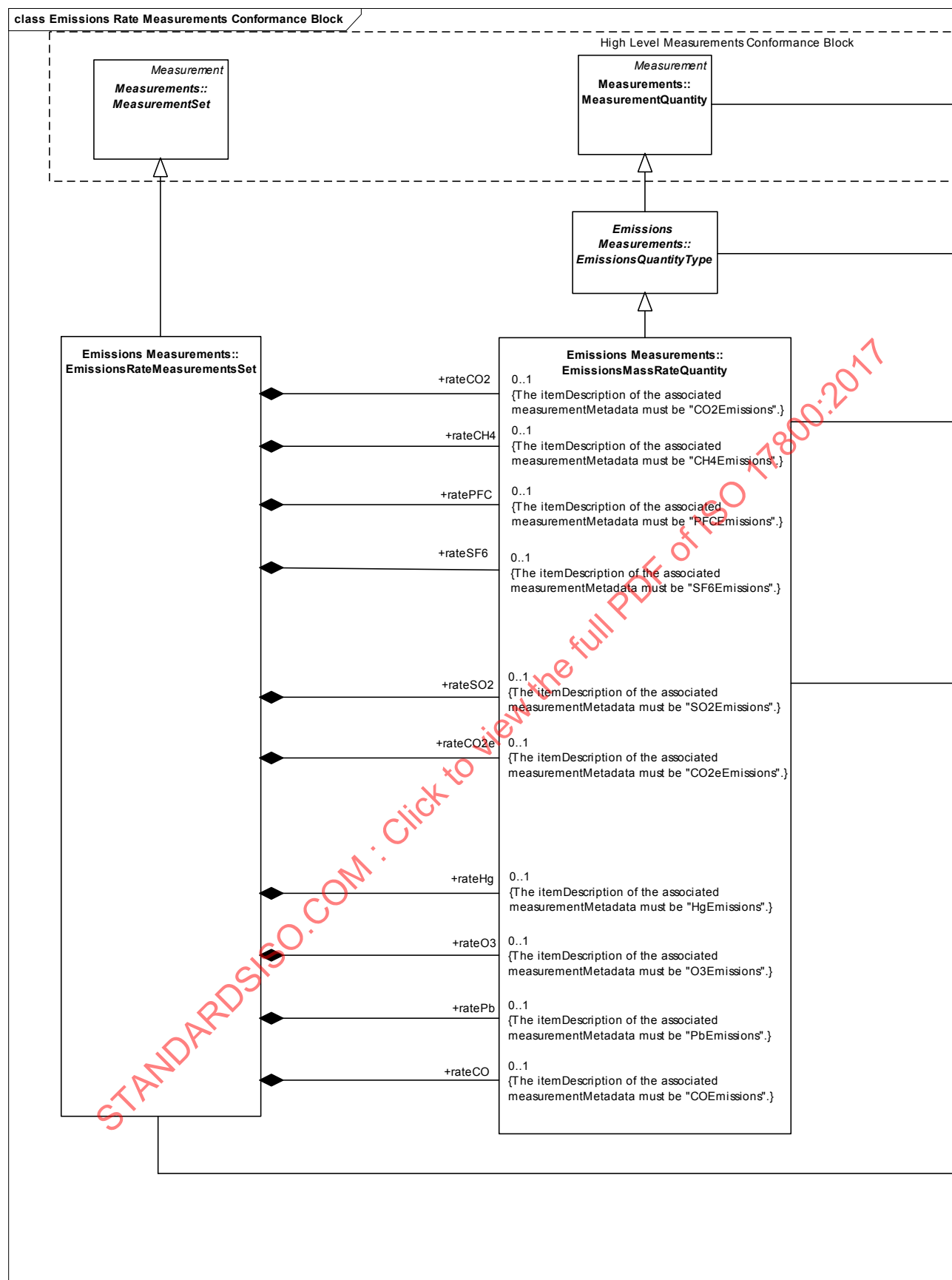
UML Element Location: *Model/Common Primitive Types, Classes, and Enumerations/Enumerations/UnitSymbolKind*. See Clause 6.6.2.6 for the complete definition of this element.

7.3.2.3. Emissions Rate Measurements Conformance Block Diagram

It is sometimes necessary to know more than one type of emissions rate measurement for a given location and time period. For example, it may be necessary to measure the rate of sulfur dioxide, nitrogen dioxide, and mercury emissions for a given point during a given time period. When this need arises, an EmissionsRateMeasurementSet class is used. Each of the individual emissions rates that can be represented using a subclass of the abstract EmissionsQuantityType is optionally included within this EmissionsRateMeasurementSet.

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



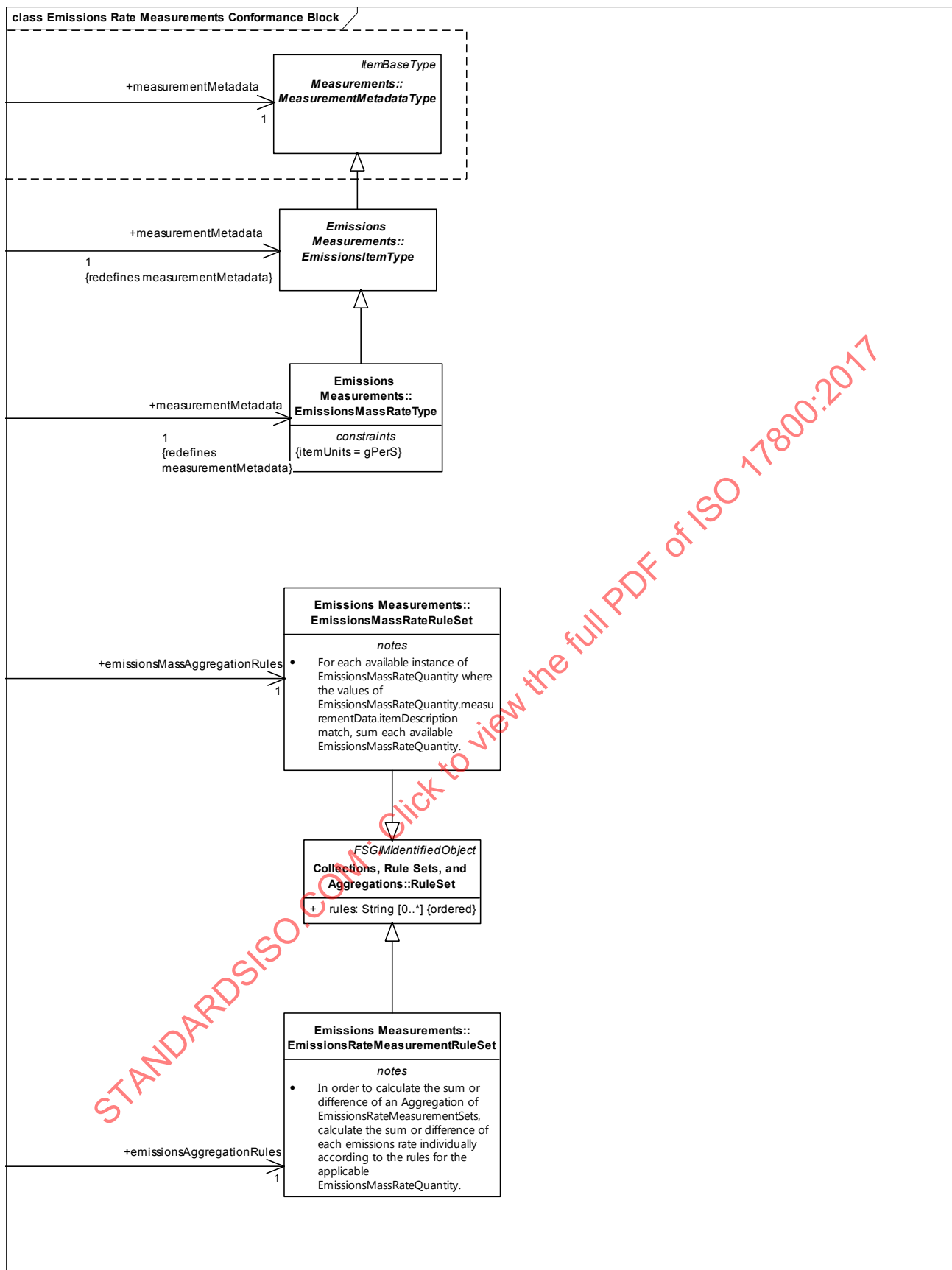


Figure 7.5 - Emissions Rate Measurements Conformance Block

The Emissions Rate Measurements Conformance Block Diagram makes use of the following elements:

EmissionsItemType (Abstract Class)

Metadata associated with an emissions measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsItemType*. See Clause 6.6.4.8.4 for the complete definition of this element.

EmissionsMassRateQuantity (Class)

Quantity/metadata pair used to represent the measurement of a specific type of emissions that is measured by mass rate.

Note: The determination of which instances of EmissionsMassRateQuantity are required in a given application is a local matter that may be within the purview of a local regulatory authority.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassRateQuantity*. See Clause 6.6.4.8.7 for the complete definition of this element.

EmissionsMassRateRuleSet (Class)

- For each available instance of EmissionsMassRateQuantity where the values of EmissionsMassRateQuantity.measurementData.itemDescription match, sum each available EmissionsMassRateQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassRateRuleSet*. See Clause 6.6.4.8.8 for the complete definition of this element.

EmissionsMassRateType (Class)

Metadata associated with an emissions measurement whose quantity indicates the rate of mass of the emission.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsMassRateType*. See Clause 6.6.4.8.9 for the complete definition of this element.

EmissionsQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of emissions measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsQuantityType*. See Clause 6.6.4.8.13 for the complete definition of this element.

EmissionsRateMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EmissionsRateMeasurementSets, calculate the sum or difference of each emissions rate individually according to the rules for the applicable EmissionsMassRateQuantity.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsRateMeasurementRuleSet*. See Clause 6.6.4.8.14 for the complete definition of this element.

EmissionsRateMeasurementsSet (Class)

A container class used to hold zero to one of each type of emissions rate measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Emissions Measurements\EmissionsRateMeasurementsSet*. See Clause 6.6.4.8.15 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

RuleSet (Class)

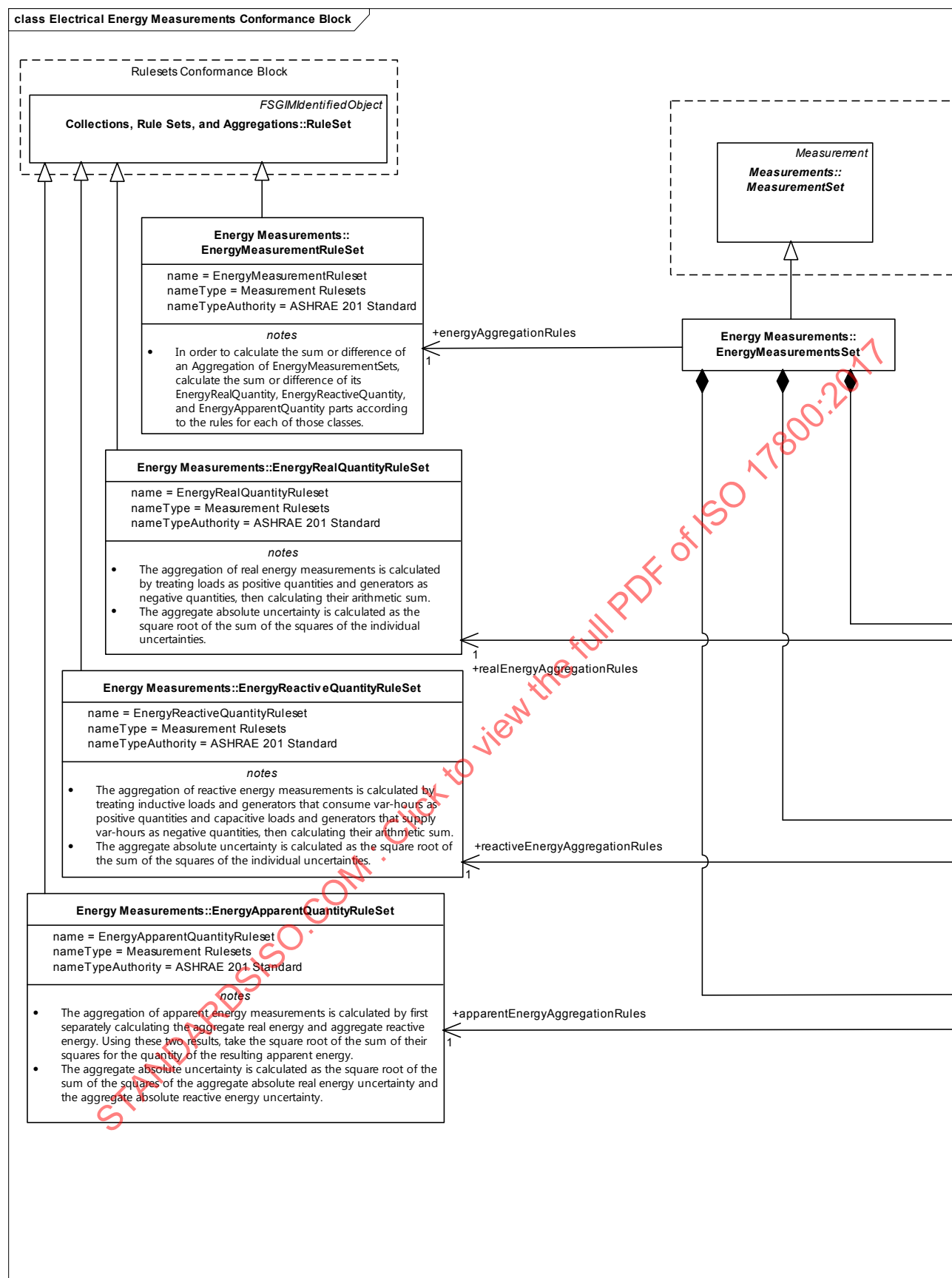
A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

7.3.2.4. Electrical Energy Measurements Conformance Block Diagram

This diagram depicts the conformance blocks, classes, and attributes used to define electrical energy measurements.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



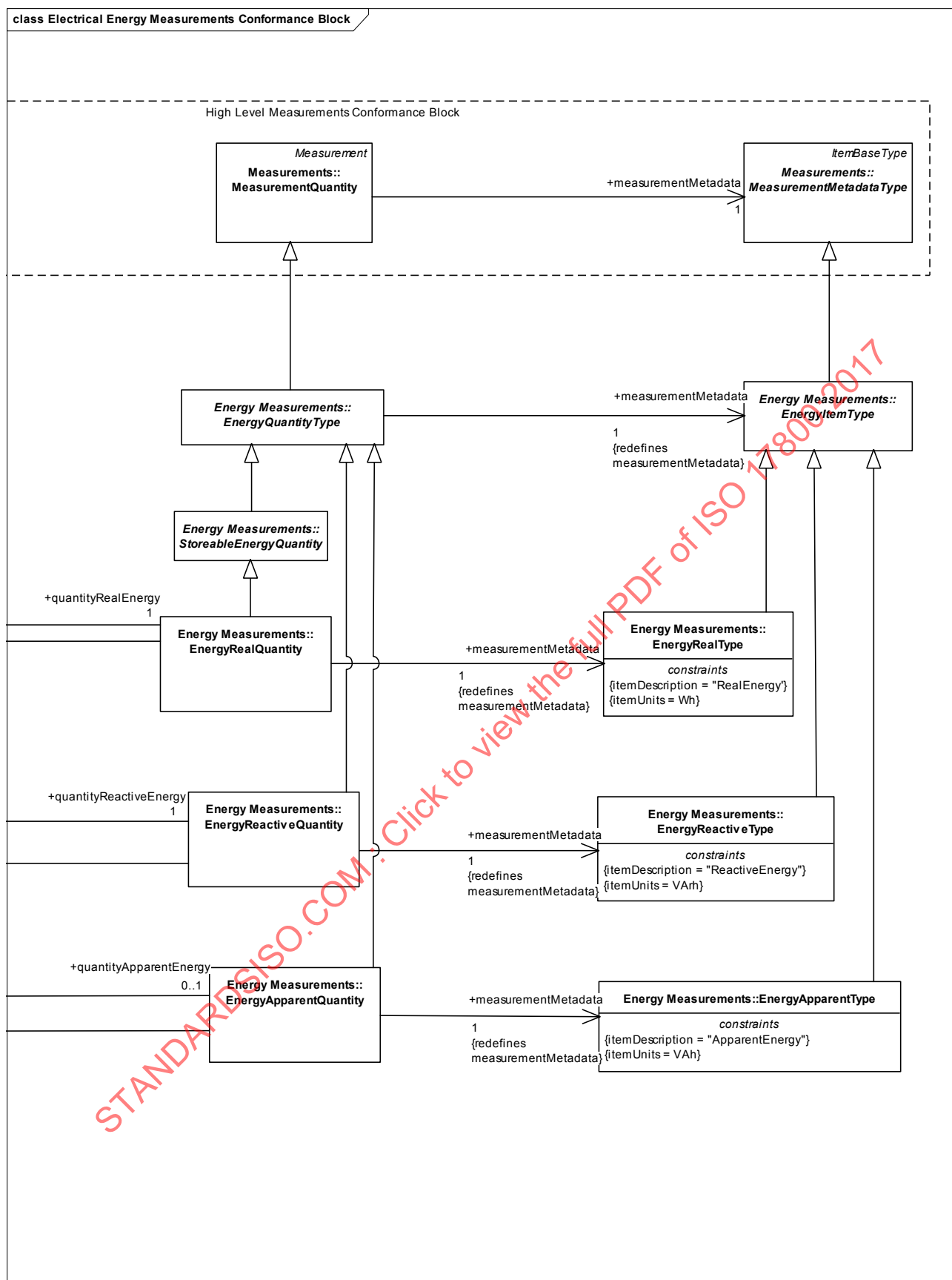


Figure 7.6 - Electrical Energy Measurements Conformance Block

The Electrical Energy Measurements Conformance Block Diagram makes use of the following elements:

EnergyApparentQuantity (Class)

Quantity/metadata pair used to represent apparent energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentQuantity*. See Clause 6.6.4.7.6 for the complete definition of this element.

EnergyApparentQuantityRuleSet (Class)

- The aggregation of apparent energy measurements is calculated by first separately calculating the aggregate real energy and aggregate reactive energy. Using these two results, take the square root of the sum of their squares for the quantity of the resulting apparent energy.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the aggregate absolute real energy uncertainty and the aggregate absolute reactive energy uncertainty.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentQuantityRuleSet*. See Clause 6.6.4.7.2 for the complete definition of this element.

EnergyApparentType (Class)

Metadata associated with measurements of apparent energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyApparentType*. See Clause 6.6.4.7.7 for the complete definition of this element.

EnergyItemType (Abstract Class)

Metadata associated with an energy measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyItemType*. See Clause 6.6.4.7.8 for the complete definition of this element.

EnergyMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of EnergyMeasurementSets, calculate the sum or difference of its EnergyRealQuantity, EnergyReactiveQuantity, and EnergyApparentQuantity parts according to the rules for each of those classes.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementRuleSet*. See Clause 6.6.4.7.3 for the complete definition of this element.

EnergyMeasurementsSet (Class)

A container class used to hold zero to one of each type of energy measurement that could be made for a given location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyMeasurementsSet*. See Clause 6.6.4.7.9 for the complete definition of this element.

EnergyQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyQuantityType*. See Clause 6.6.4.7.10 for the complete definition of this element.

EnergyReactiveQuantity (Class)

Quantity/metadata pair used to represent reactive energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveQuantity*. See Clause 6.6.4.7.11 for the complete definition of this element.

EnergyReactiveQuantityRuleSet (Class)

- The aggregation of reactive energy measurements is calculated by treating inductive loads and generators that consume var-hours as positive quantities and capacitive loads and generators that supply var-hours as negative quantities, then calculating their arithmetic sum.

- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveQuantityRuleSet*. See Clause 6.6.4.7.4 for the complete definition of this element.

EnergyReactiveType (Class)

Metadata associated with measurements of reactive energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyReactiveType*. See Clause 6.6.4.7.12 for the complete definition of this element.

EnergyRealQuantity (Class)

Quantity/metadata pair used to represent real energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantity*. See Clause 6.6.4.7.13 for the complete definition of this element.

EnergyRealQuantityRuleSet (Class)

- The aggregation of real energy measurements is calculated by treating loads as positive quantities and generators as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealQuantityRuleSet*. See Clause 6.6.4.7.5 for the complete definition of this element.

EnergyRealType (Class)

Metadata associated with measurements of real energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyRealType*. See Clause 6.6.4.7.14 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*. See Clause 6.6.4.2 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

StoreableEnergyQuantity (Abstract Class)

Quantity used to represent real electrical or thermal energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\StoreableEnergyQuantity*. See Clause 6.6.4.7.17 for the complete definition of this element.

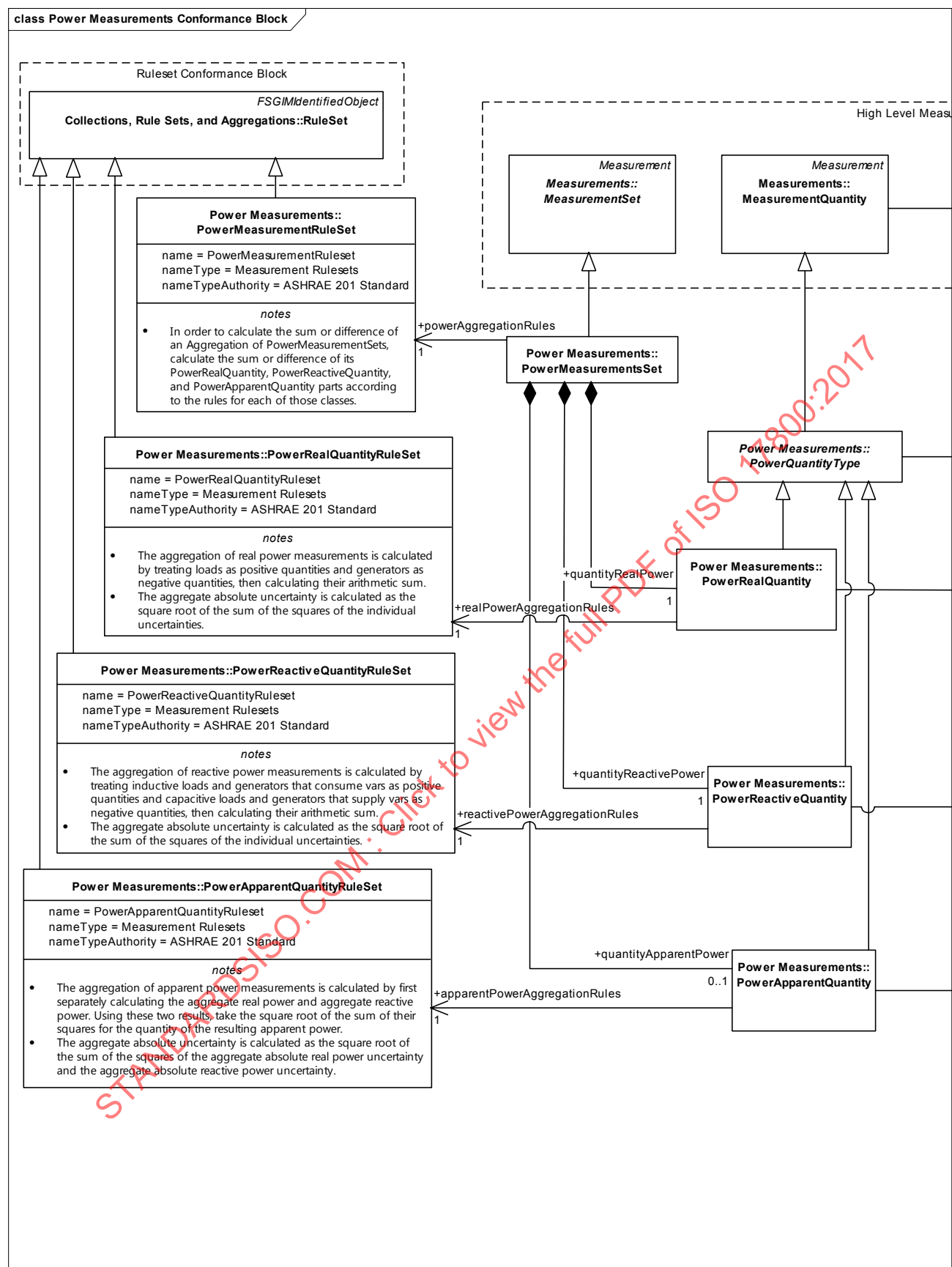
7.3.2.5. Power Measurements Conformance Block Diagram

This diagram depicts the conformance blocks, classes, and attributes used to define power measurements.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



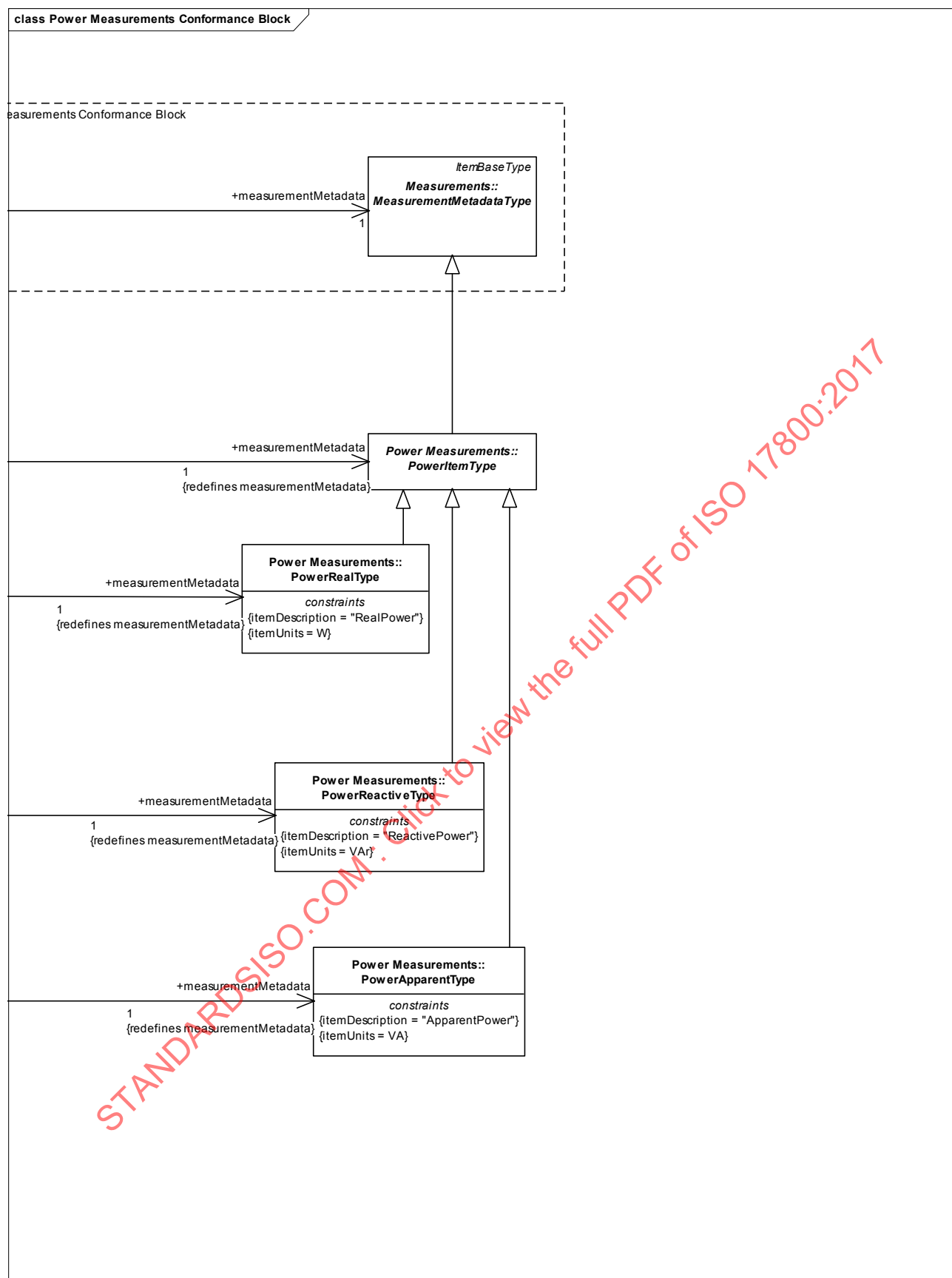


Figure 7.7 - Power Measurements Conformance Block

The Power Measurements Conformance Block Diagram makes use of the following elements:

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and*

Enumerations\Measurements\MeasurementMetadataType. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and*

Enumerations\Measurements\MeasurementQuantity. See Clause 6.6.4.5 for the complete definition of this element.

MeasurementSet (Abstract Class)

An abstract container class used to hold either the power measurements, energy measurements, or emissions measurements that could be made for a given EMUsagePoint and time reference.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementSet*.

See Clause 6.6.4.2 for the complete definition of this element.

PowerApparentQuantity (Class)

Quantity and pair used to represent apparent power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerApparentQuantity. See Clause 6.6.4.6.8 for the complete definition of this element.

PowerApparentQuantityRuleSet (Class)

- The aggregation of apparent power measurements is calculated by first separately calculating the aggregate real power and aggregate reactive power. Using these two results, take the square root of the sum of their squares for the quantity of the resulting apparent power.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the aggregate absolute real power uncertainty and the aggregate absolute reactive power uncertainty.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerApparentQuantityRuleSet. See Clause 6.6.4.6.1 for the complete definition of this element.

PowerApparentType (Class)

Metadata associated with measurements of apparent power.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerApparentType. See Clause 6.6.4.6.9 for the complete definition of this element.

PowerItemType (Abstract Class)

Metadata associated with a power measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerItemType. See Clause 6.6.4.6.10 for the complete definition of this element.

PowerMeasurementRuleSet (Class)

- In order to calculate the sum or difference of an Aggregation of PowerMeasurementSets, calculate the sum or difference of its PowerRealQuantity, PowerReactiveQuantity, and PowerApparentQuantity parts according to the rules for each of those classes.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerMeasurementRuleSet. See Clause 6.6.4.6.3 for the complete definition of this element.

PowerMeasurementsSet (Class)

A container class used to hold one of each type of power measurement made at a given physical location and time.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power*

Measurements\PowerMeasurementsSet. See Clause 6.6.4.6.11 for the complete definition of this element.

PowerQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerQuantityType*. See Clause 6.6.4.6.12 for the complete definition of this element.

PowerReactiveQuantity (Class)

Quantity and pair used to represent reactive power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveQuantity*. See Clause 6.6.4.6.13 for the complete definition of this element.

PowerReactiveQuantityRuleSet (Class)

- The aggregation of reactive power measurements is calculated by treating inductive loads and generators that consume vars as positive quantities and capacitive loads and generators that supply vars as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveQuantityRuleSet*. See Clause 6.6.4.6.5 for the complete definition of this element.

PowerReactiveType (Class)

Metadata associated with measurements of reactive power.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerReactiveType*. See Clause 6.6.4.6.14 for the complete definition of this element.

PowerRealQuantity (Class)

Quantity and metadata pair used to represent real power measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealQuantity*. See Clause 6.6.4.6.15 for the complete definition of this element.

PowerRealQuantityRuleSet (Class)

- The aggregation of real power measurements is calculated by treating loads as positive quantities and generators as negative quantities, then calculating their arithmetic sum.
- The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealQuantityRuleSet*. See Clause 6.6.4.6.6 for the complete definition of this element.

PowerRealType (Class)

Metadata associated with measurements of real power.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Power Measurements\PowerRealType*. See Clause 6.6.4.6.16 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

7.3.2.6. Thermal Energy Measurements Conformance Block Diagram

This diagram depicts the conformance blocks, classes, and attributes used to define thermal energy measurements.

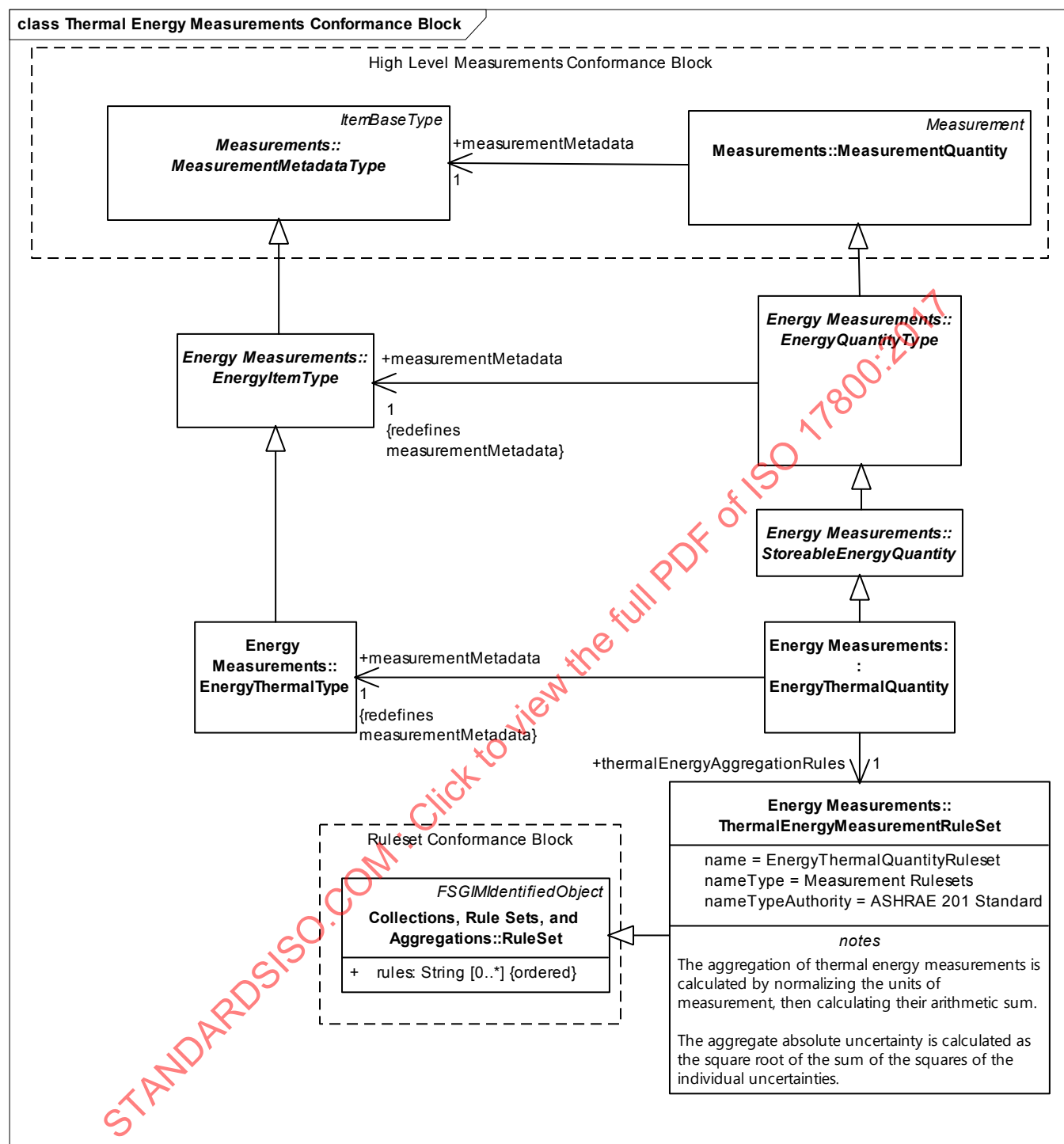


Figure 7.8 - Thermal Energy Measurements Conformance Block

The Thermal Energy Measurements Conformance Block Diagram makes use of the following elements:

EnergyItemType (Abstract Class)

Metadata associated with an energy measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyItemType*. See Clause 6.6.4.7.8 for the complete definition of this element.

EnergyQuantityType (Abstract Class)

Quantity/metadata pair used to represent types of energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyQuantityType*. See Clause 6.6.4.7.10 for the complete definition of this element.

EnergyThermalQuantity (Class)

Quantity/metadata pair used to represent thermal energy measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyThermalQuantity*. See Clause 6.6.4.7.15 for the complete definition of this element.

EnergyThermalType (Class)

Metadata associated with measurements of thermal energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\EnergyThermalType*. See Clause 6.6.4.7.16 for the complete definition of this element.

MeasurementMetadataType (Abstract Class)

Metadata associated with a measurement.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementMetadataType*. See Clause 6.6.4.4 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

StoreableEnergyQuantity (Abstract Class)

Quantity used to represent real electrical or thermal energy.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\StoreableEnergyQuantity*. See Clause 6.6.4.7.17 for the complete definition of this element.

ThermalEnergyMeasurementRuleSet (Class)

The aggregation of thermal energy measurements is calculated by normalizing the units of measurement, then calculating their arithmetic sum.

The aggregate absolute uncertainty is calculated as the square root of the sum of the squares of the individual uncertainties.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Energy Measurements\ThermalEnergyMeasurementRuleSet*. See Clause 6.6.4.7.1 for the complete definition of this element.

7.3.3. Component Conformance Blocks

Conformance blocks in this clause address Devices, Energy Managers, Generators, Loads, and Meters Components. The Collection, Aggregation and Standard Aggregation Conformance Blocks are included as part of the Energy Manager.

UML Package Location: *Model\Conformance Blocks\Component Conformance Blocks*

7.3.3.1. Device Conformance Block

Conformance blocks in this clause address the portion of the FSGIM that represents a physical entity that is then modeled using the remaining classes within the FSGIM. These conformance blocks allow devices to be addressed, and tagged. Nameplate and other metadata about the device can also be defined.

UML Package Location: *Model\Conformance Blocks\Component Conformance Blocks\Device Conformance Block*

7.3.3.1.1. Device Conformance Block Diagram

This diagram depicts the conformance blocks, classes and attributes used to define nameplate and other metadata about the device. A device status may also be defined. Semantic tags may be applied to the device. The scope of the tags are managed by the name type authority.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

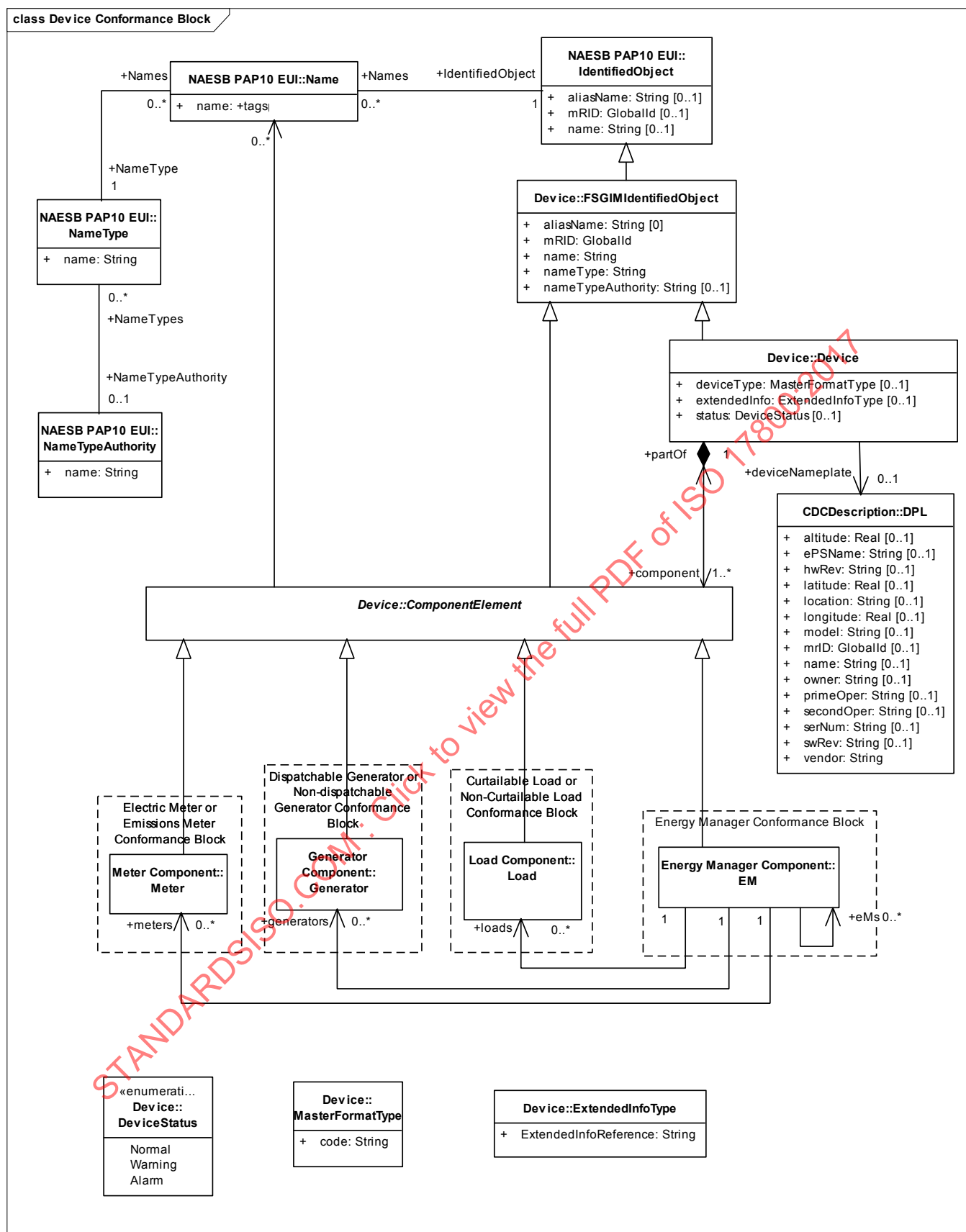


Figure 7.9 - Device Conformance Block

The Device Conformance Block Diagram makes use of the following elements:

ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*. See Clause 5.2.3.1 for the complete definition of this element.

Device (Class)

This is the class that represents a physical device in the facility

UML Element Location: *Model\Device and Model Components\Device\Device*. See Clause 5.2.3.2 for the complete definition of this element.

DeviceStatus (Enumeration)

The general indication of the health of a Device.

UML Element Location: *Model\Device and Model Components\Device\DeviceStatus*. See Clause 5.2.3.6 for the complete definition of this element.

DPL (Class)

This common data class is used to identify entities like primary equipment or physical devices.

UML Element Location: *Model\Model Elements from External Sources\IEC61850\IEC61850_7_3\CommonDataClasses\CDCDescription\DPL*. See Clause 5.7.4.5.3.1.18 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

ExtendedInfoType (Class)

This class defines an reference external from this model that can be used to gain additional information about the device.

UML Element Location: *Model\Device and Model Components\Device\ExtendedInfoType*. See Clause 5.2.3.3 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject*. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

MasterFormatType (Class)

This class defines a string attribute with specific constraints to map into the Master Format 2012 numbering system.

UML Element Location: *Model\Device and Model Components\Device\MasterFormatType*. See Clause 5.2.3.5 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Name*. See Clause 5.7.5.2.1.14 for the complete definition of this element.

NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameType*. See Clause 5.7.5.2.1.15 for the complete definition of this element.

NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameTypeAuthority*. See Clause 5.7.5.2.1.16 for the complete definition of this element.

7.3.3.2. Energy Manager Conformance Blocks

Conformance blocks in this clause address the portion of the FSGIM that represents all logical and physical devices that make decisions regarding the consumption, generation and measurement of energy. Collection and Aggregation Conformance Blocks are also included in this clause.

UML Package Location: *Model\Conformance Blocks\Component Conformance Blocks\Energy Manager Conformance Blocks*

7.3.3.2.1. AllResourcesInEMDomain Conformance Block Diagram

This diagram depicts the classes and attributes used to define the standard collection consisting of all resources directly managed by a given energy manager.

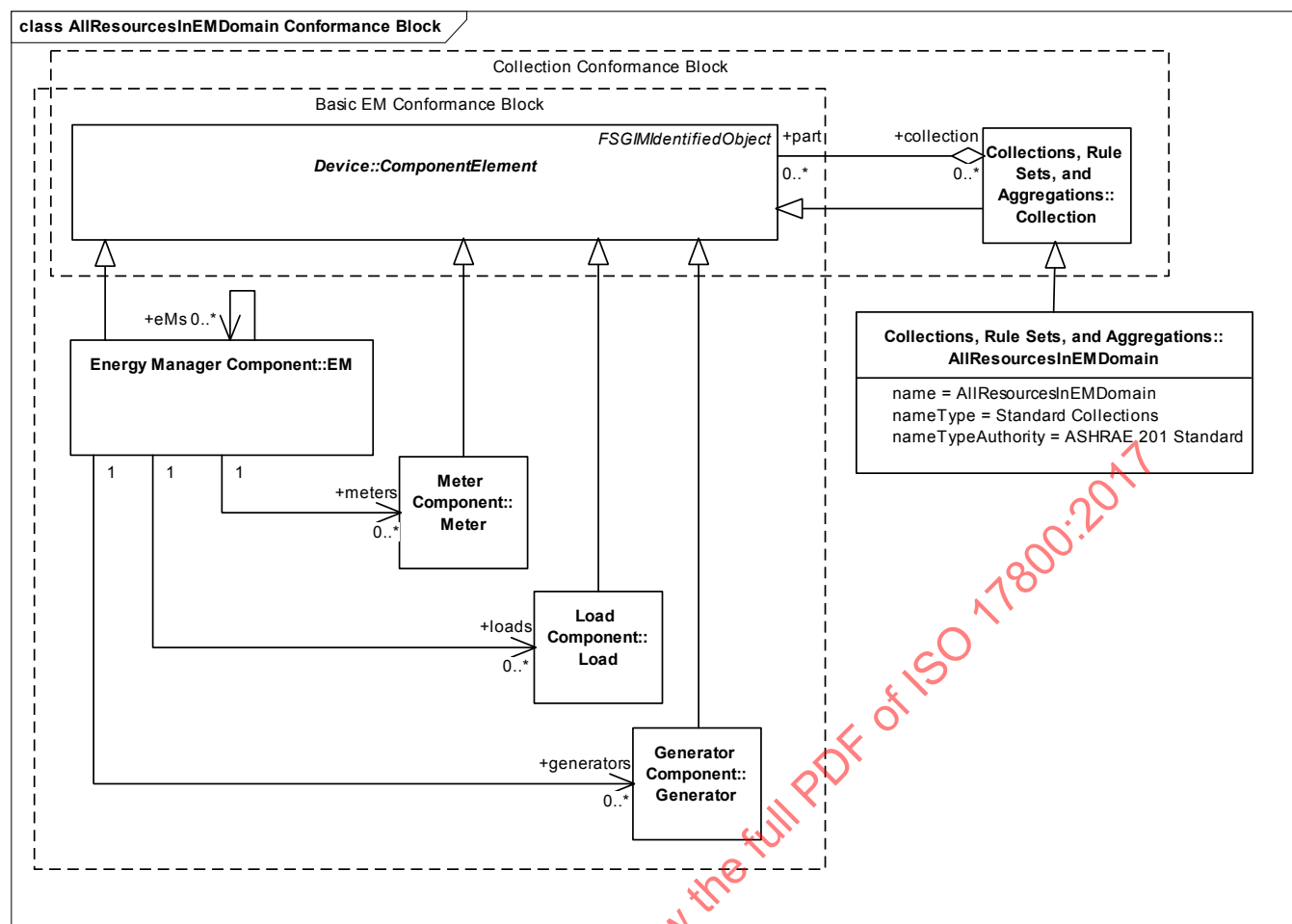


Figure 7.10 - AllResourcesInEMDomain Conformance Block

The AllResourcesInEMDomain Conformance Block Diagram makes use of the following elements:

AllResourcesInEMDomain (Class)

The AllResourcesInEMDomain collection describes the set of all of the loads, generators, meters, and EMs that are directly managed by the energy manager. Contributions of loads, generators, meters, and EMs that are indirect subordinates of a given energy manager are proxied by the standard aggregations that are contained in the direct subordinate energy managers.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\AllResourcesInEMDomain*. See Clause 5.6.7.11.2 for the complete definition of this element.

Collection (Class)

This class defines a set of entities that is identified as a group because the members share some trait. A Collection can contain other Collections.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Collection*. See Clause 5.6.7.11.3 for the complete definition of this element.

ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*. See Clause 5.2.3.1 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

7.3.3.2.2. Aggregation Conformance Block Diagram

This diagram depicts the conformance blocks, classes and attributes used to define aggregations. This diagram only defines the generic mechanism used to define an aggregation. Standard Aggregations are defined in other sections of this clause.

The section of the diagram labeled "High Level Measurements Conformance Block" indicates that a conforming protocol may choose to support any or all of the conformance blocks noted in that diagram when supporting the Aggregation Conformance Block. The conforming protocol shall specify which of these conformance blocks are supported.

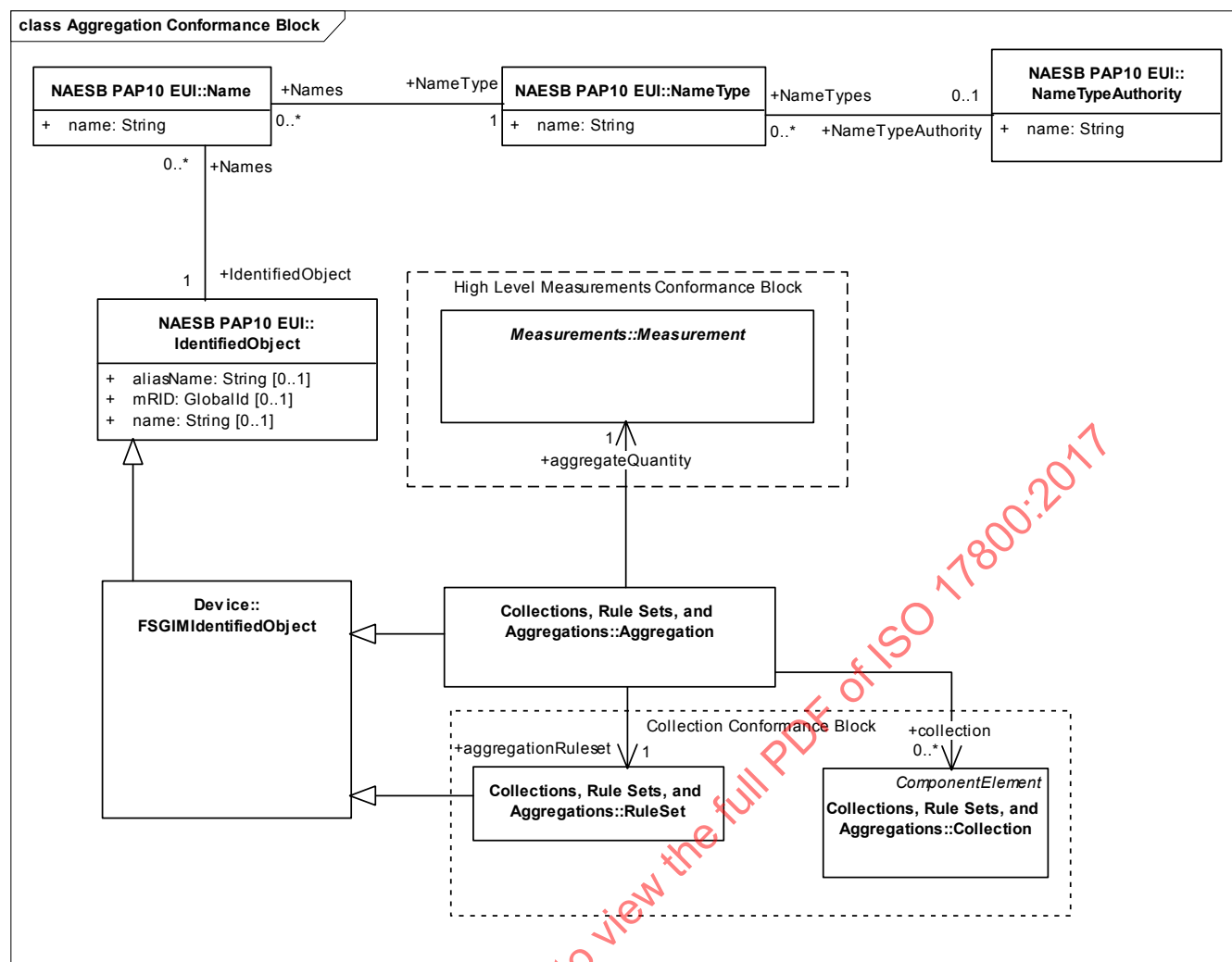


Figure 7.11 - Aggregation Conformance Block

The Aggregation Conformance Block Diagram makes use of the following elements:

Aggregation (Class)

An aggregation allows the assembly of statistics of interest arrived at through defined algorithms applied to other parts of the model. The description of an aggregation includes all of the inputs that comprise the algorithms as well as the outputs of the algorithm. An aggregation instance represents one or more aggregateQuantities that are computed by applying aggregation rules for each quantity against a Collection or FilteredCollection of Components.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Aggregation*. See Clause 5.6.7.11.1 for the complete definition of this element.

Collection (Class)

This class defines a set of entities that is identified as a group because the members share some trait. A Collection can contain other Collections.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Collection*. See Clause 5.6.7.11.3 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Measurement (Abstract Class)

Abstraction of a measured value or related set of values.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\Measurement*. See Clause 6.6.4.1 for the complete definition of this element.

Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Name. See Clause 5.7.5.2.1.14 for the complete definition of this element.

NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameType. See Clause 5.7.5.2.1.15 for the complete definition of this element.

NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\Model Elements from External*

Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameTypeAuthority. See Clause 5.7.5.2.1.16 for the complete definition of this element.

RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

7.3.3.2.3. Basic EM Conformance Block Diagram

This diagram depicts the conformance blocks, classes and attributes used to define and name an energy manager and its associations with other ComponentElements.

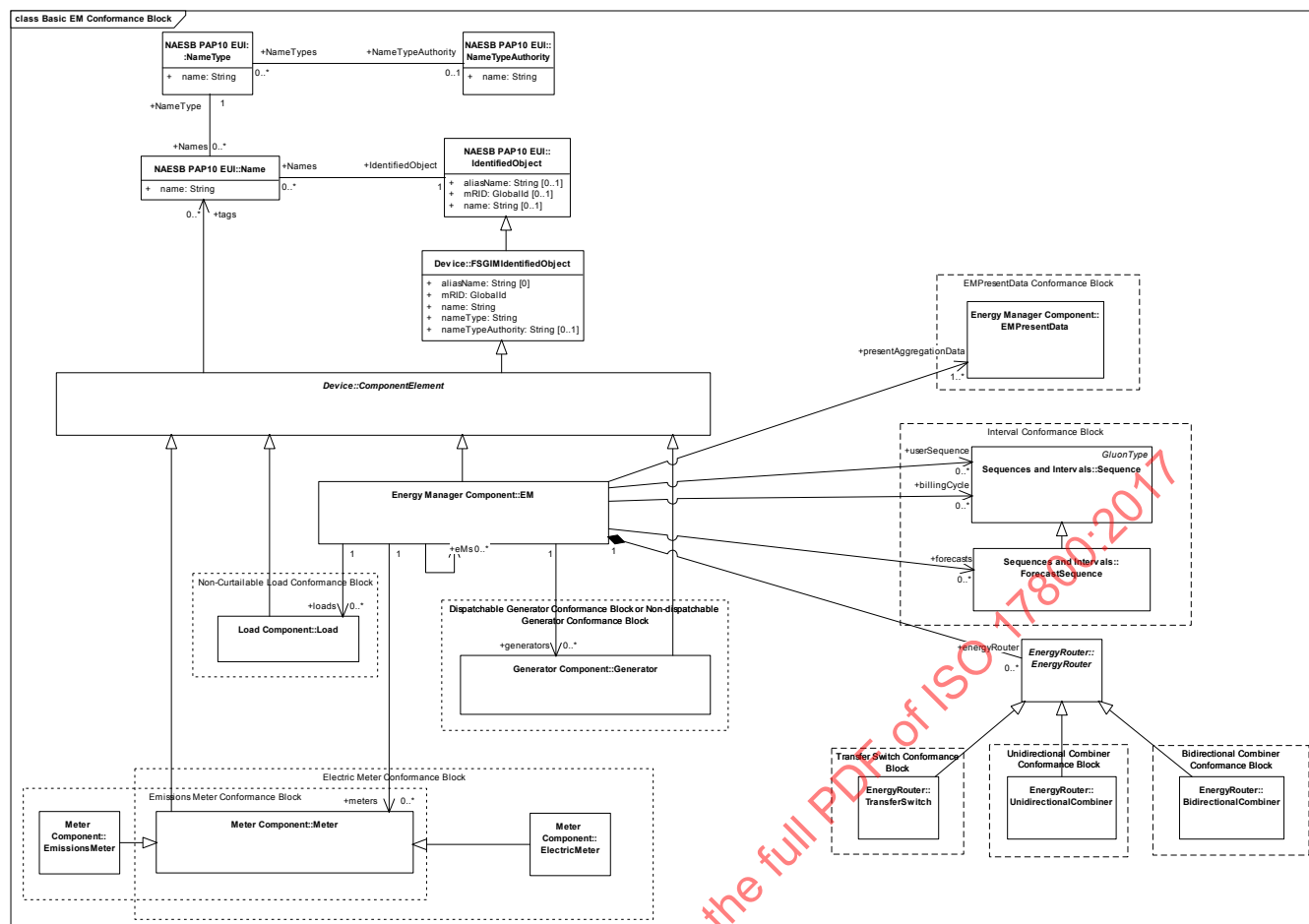


Figure 7.12 - Basic EM Conformance Block

The Basic EM Conformance Block Diagram makes use of the following elements:

BidirectionalCombiner (Class)

This class is used to model bidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that allows energy to flow from the onsite generation source to the utility grid but not from the utility grid to the onsite generation source. The effect of the constraints that are placed on this class results in the following behavior for the bidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.
- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power sent back to the grid is the amount of power available from the onsite generation side of the combiner box minus the amount of power drawn from load side of the combiner box.

UML Element Location: *Model\Device and Model Components\Energy Manager*

Component\EnergyRouter\BidirectionalCombiner. See Clause 5.6.7.14.1 for the complete definition of this element.

ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*. See Clause 5.2.3.1 for the complete definition of this element.

ElectricMeter (Class)

The ElectricMeter Class is a subclass of the Meter Class. It is an abstract representation of devices that measure real and reactive power or energy. These devices optionally measure other electrical parameters such as apparent power, apparent energy, present subinterval demand or power quality.

UML Element Location: *Model\Device and Model Components\Meter Component\ElectricMeter*. See Clause 5.3.2.2 for the complete definition of this element.

EM (Class)

The EM Class is the central class of the Energy Manager Component. It contains indirectly through its associations the information elements that support the functionality of an energy manager. An energy manager reports on and makes decisions involving meters, loads, generators, storage devices, other energy managers, and external information.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EM*. See Clause 5.6.7.2 for the complete definition of this element.

EmissionsMeter (Class)

The EmissionsMeter Class is a subclass of the Meter Class which is used to represent devices that are capable of measuring pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\EmissionsMeter*. See Clause 5.3.2.3 for the complete definition of this element.

EMPresentData (Class)

The container that directly holds all of the aggregated present data and indirectly holds the associated present context.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EMPresentData*. See Clause 5.6.7.7 for the complete definition of this element.

EnergyRouter (Abstract Class)

This class is used to model the various devices that are capable of connecting a load to more than one source of energy.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\EnergyRouter*. See Clause 5.6.7.14.3 for the complete definition of this element.

ForecastSequence (Class)

The Sequence that is used for storing all forecasts, including forecasts for time periods where the originally forecasted time period has now passed.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\ForecastSequence*. See Clause 5.6.7.10.2 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

Generator (Class)

The Generator Class is the central class of the Generator Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a generator. A generator produces energy or retrieves it from storage.

UML Element Location: *Model\Device and Model Components\Generator Component\Generator*. See Clause 5.5.6.2 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject*. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Load (Class)

The Load Class is the central class of the Load Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a load.

UML Element Location: *Model\Device and Model Components\Load Component\Load*. See Clause 5.4.2.5 for the complete definition of this element.

Meter (Class)

The Meter Class is the central class of the Meter Component. It contains, directly through its attributes and indirectly through its associations, the information elements that support the functionality of a meter. A meter measures electricity or pollution emissions.

UML Element Location: *Model\Device and Model Components\Meter Component\Meter*. See Clause 5.3.2.1 for the complete definition of this element.

Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Name*. See Clause 5.7.5.2.1.14 for the complete definition of this element.

NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameType*. See Clause 5.7.5.2.1.15 for the complete definition of this element.

NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameTypeAuthority*. See Clause 5.7.5.2.1.16 for the complete definition of this element.

Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be surmised by monitoring the time references.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*. See Clause 5.6.7.10.4 for the complete definition of this element.

TransferSwitch (Class)

This class is used to model transfer switches, which are devices that allow a load or set of loads to be switched between two power sources without allowing the two power sources to be connected together.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\EnergyRouter\TransferSwitch*. See Clause 5.6.7.14.7 for the complete definition of this element.

UnidirectionalCombiner (Class)

This class is used to model unidirectional combiner boxes, which are devices that allow two sources of energy to be connected to one or more loads in a way that does not allow energy to flow from one source to another. The effect of the constraints that are placed on this class results in the following behavior for the unidirectional combiner box:

- If the amount of power available from the onsite generation side of the combiner box is less than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn

from utility grid side of the combiner box is the difference between the amount of power drawn from load side of the combiner box and the amount of power injected into the onsite generation side of the combiner box.

- If the amount of power available from the onsite generation side of the combiner box is greater than the amount of power consumed by the equipment connected to load side of the combiner box, then the amount of power drawn from utility grid side of the combiner box is zero and the amount of power actually injected into the onsite generation side of the combiner box is equal to the amount of power drawn from load side of the combiner box.

UML Element Location: *Model\Device and Model Components\Energy Manager*

Component\EnergyRouter\UnidirectionalCombiner. See Clause 5.6.7.14.9 for the complete definition of this element.

7.3.3.2.4. Circuit Conformance Block Diagram

This diagram depicts the conformance blocks, classes and attributes used to define and name an electrical circuit.

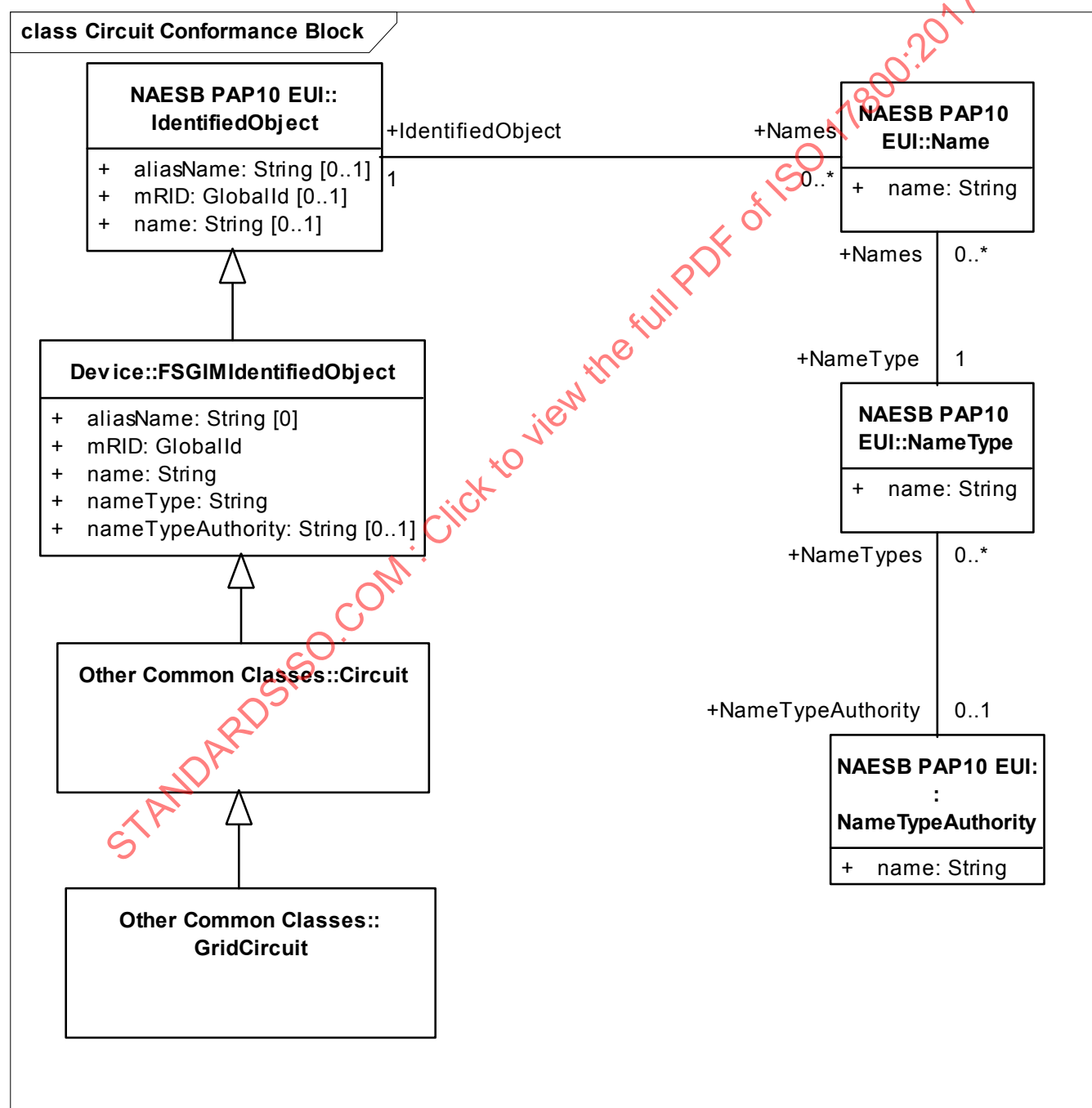


Figure 7.13 - Circuit Conformance Block

The Circuit Conformance Block Diagram makes use of the following elements:

Circuit (Class)

This class is used to model electrical circuits. All ConnectionPoints that are electrically tied together will point to the same instance of this class.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\Circuit*. See Clause 6.6.5.1 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

GridCircuit (Class)

This class is used to model electrical circuits that are part of the utility grid.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\GridCircuit*. See Clause 6.6.5.3 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject*. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Name*. See Clause 5.7.5.2.1.14 for the complete definition of this element.

NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameType*. See Clause 5.7.5.2.1.15 for the complete definition of this element.

NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameTypeAuthority*. See Clause 5.7.5.2.1.16 for the complete definition of this element.

7.3.3.2.5. Collection Conformance Block Diagram

This diagram depicts the conformance blocks, classes and attributes used to define collections. This diagram only defines the generic mechanism used to define collections. Standard Collections are defined in other sections of this clause.

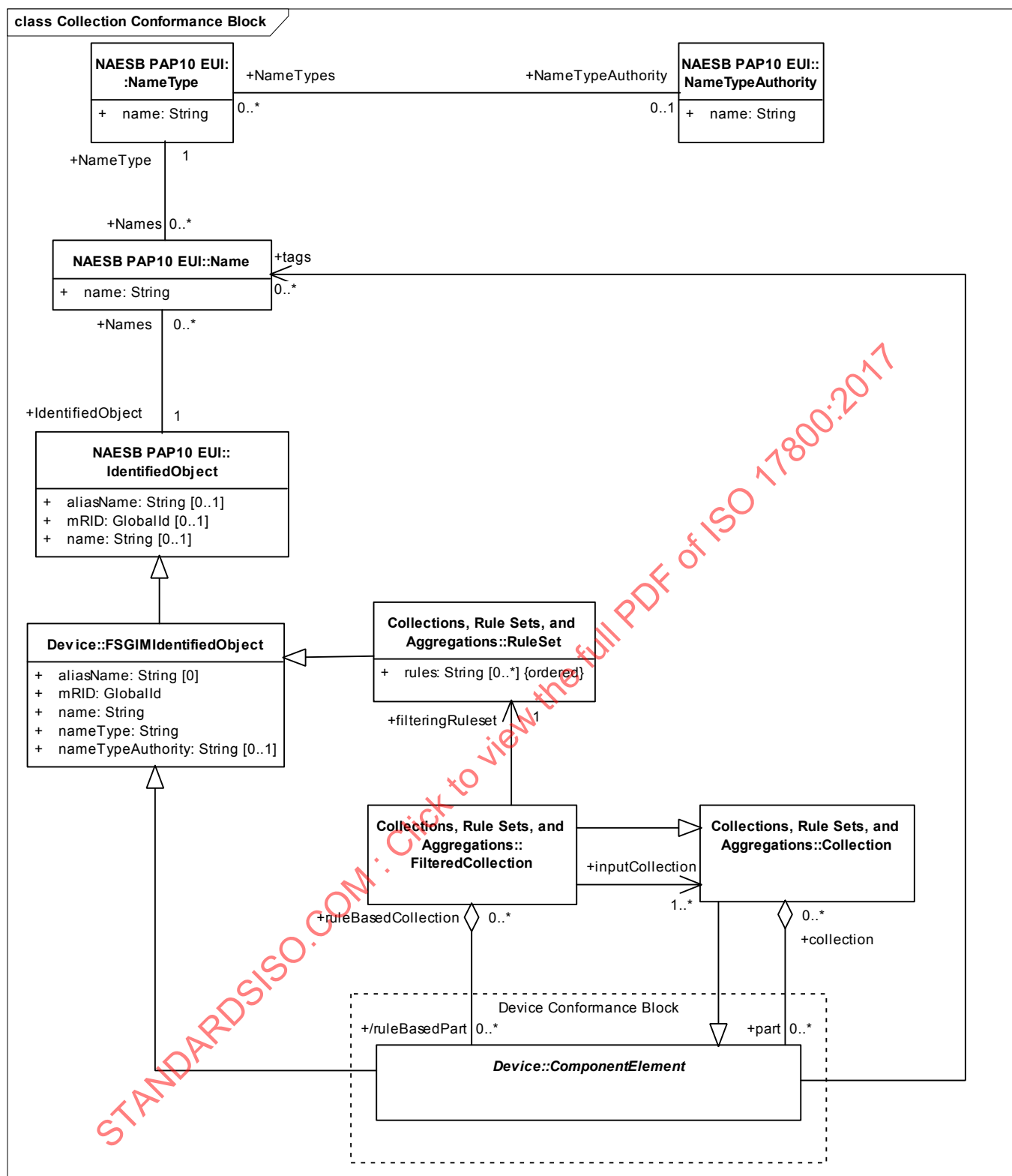


Figure 7.14 - Collection Conformance Block

The Collection Conformance Block Diagram makes use of the following elements:

Collection (Class)

This class defines a set of entities that is identified as a group because the members share some trait. A Collection can contain other Collections.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\Collection*. See Clause 5.6.7.11.3 for the complete definition of this element.

ComponentElement (Abstract Class)

This class is an abstract class representing the top level class of one of the four FSGIM model components or a Collection of these top level classes. Note that the parts of a Collection instance inherit the tag attribute values of its parent collection ComponentElement instance.

UML Element Location: *Model\Device and Model Components\Device\ComponentElement*. See Clause 5.2.3.1 for the complete definition of this element.

FilteredCollection (Class)

A FilteredCollection "filters" one or more inputCollections according to a set of filteringRules to produce a new instance of a Collection with the subset defined by the Ruleset.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\FilteredCollection*. See Clause 5.6.7.11.4 for the complete definition of this element.

FSGIMIdentifiedObject (Class)

A standard representation of the identifier used to distinguish key object instances in the FSGIM.

UML Element Location: *Model\Device and Model Components\Device\FSGIMIdentifiedObject*. See Clause 5.2.3.4 for the complete definition of this element.

IdentifiedObject (Class)

This is a root class to provide common identification for all classes needing identification and naming attributes

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\IdentifiedObject*. See Clause 5.7.5.2.1.8 for the complete definition of this element.

Name (Class)

The Name class provides the means to define any number of human readable names for an object.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\Name*. See Clause 5.7.5.2.1.14 for the complete definition of this element.

NameType (Class)

Type of name. Possible values for attribute 'name' are implementation dependent but standard profiles may specify types. An enterprise may have multiple information technology systems each having its own local name for the same object, e.g., a planning system may have different names from an energy management system. An object may also have different names within the same system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameType*. See Clause 5.7.5.2.1.15 for the complete definition of this element.

NameTypeAuthority (Class)

Authority responsible for creation and management of names of a given type; typically an organization or an enterprise system.

UML Element Location: *Model\Model Elements from External Sources\iec_cim_naesb_eui_model_20101111Update_20120802\NAESB PAP10 EUI\NameTypeAuthority*. See Clause 5.7.5.2.1.16 for the complete definition of this element.

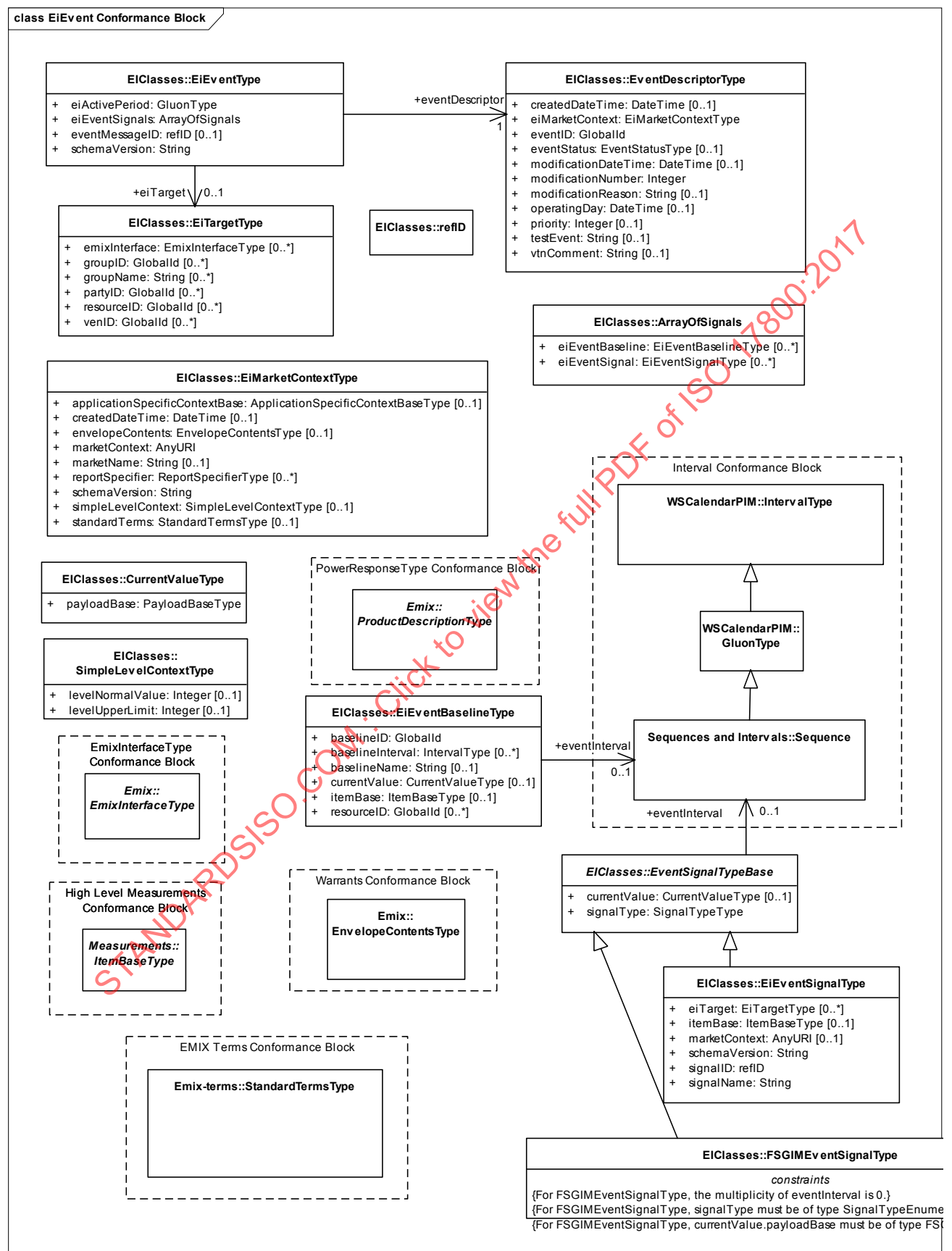
RuleSet (Class)

A Ruleset contains a set of ordered rules that defines how the value of an AggregateQuantity is derived or how the members of a FilteredCollection are selected. The text in Ruleset.rules is meant to convey the intent of the set of rules. It is not meant to imply requirements for a parsable Ruleset language. The exact format used for the rules in the Ruleset is a local matter.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Collections, Rule Sets, and Aggregations\RuleSet*. See Clause 5.6.7.11.5 for the complete definition of this element.

7.3.3.2.6. EiEvent Conformance Block Diagram

The EI Events Diagram illustrates the UML model that is used to call for performance under a transaction. The service parameters and event information distinguish different types of events. Event types include reliability events, emergency events, and others.



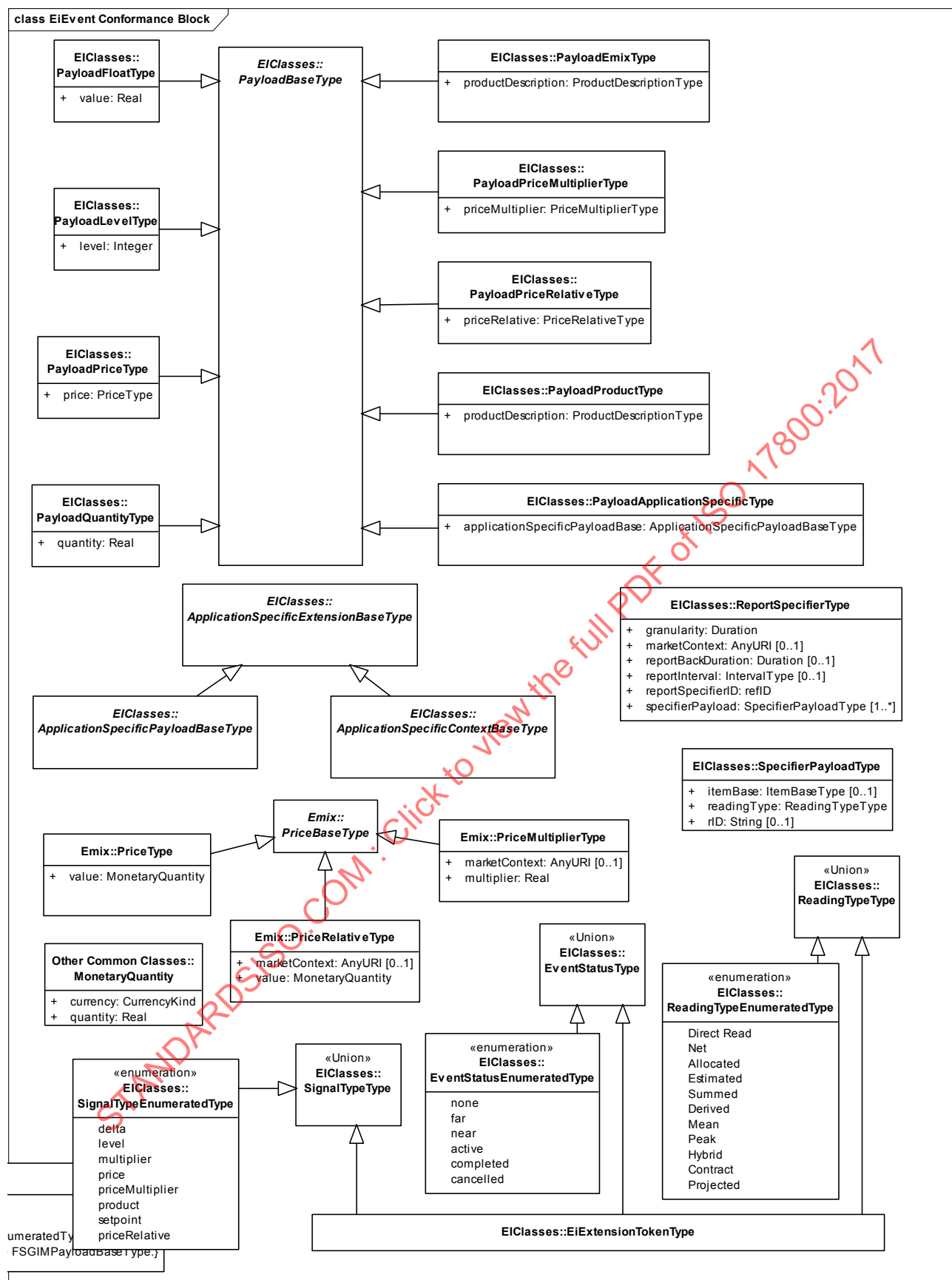


Figure 7.15 - EiEvent Conformance Block

The EiEvent Conformance Block Diagram makes use of the following elements:

ApplicationSpecificContextBaseType (Abstract Class)

The Application Specific Context Base is an abstract class to exchange invariant or setup information with an Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificContextBaseType*. See Clause 5.7.1.2.1.1 for the complete definition of this element.

ApplicationSpecificExtensionBaseType (Abstract Class)

Application Extensions are used to provide hints to or interactions with Applications running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificExtensionBaseType*. See Clause 5.7.1.2.1.2 for the complete definition of this element.

ApplicationSpecificPayloadBaseType (Abstract Class)

The Application Specific Payload Base is an abstract class to exchange feedback with an Application running on the other side of an interaction. They are not defined in Energy Interoperation, although there are specific conformance rules that must be followed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ApplicationSpecificPayloadBaseType*. See Clause 5.7.1.2.1.3 for the complete definition of this element.

ArrayOfSignals (Class)

Collection of Signal Base derived elements

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ArrayOfSignals*. See Clause 5.7.1.2.1.4 for the complete definition of this element.

CurrentValueType (Class)

This class contains the common information, at the time of an event, in an Energy Interoperation signal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\CurrentValueType*. See Clause 5.7.1.2.1.5 for the complete definition of this element.

EiEventBaselineType (Class)

This type is used for describing baseline for the event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiEventBaselineType*. See Clause 5.7.1.2.1.6 for the complete definition of this element.

EiEventSignalType (Class)

This type is used for describing the signal type communications.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiEventSignalType*. See Clause 5.7.1.2.1.7 for the complete definition of this element.

EiEventType (Class)

Core element of event-based demand response that specifies the duration and applicability of an Event. Some deadlines, time periods, and transitions may not be applicable to all products or services.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiEventType*. See Clause 5.7.1.2.1.8 for the complete definition of this element.

EiExtensionTokenType (Class)

Pattern used for extending string enumeration, where allowed

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiExtensionTokenType*. See Clause 5.7.1.2.1.9 for the complete definition of this element.

EiMarketContextType (Class)

Terms that define the market context of this transaction. The Market Context is the overall market or program rules that govern this event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiMarketContextType*. See Clause 5.7.1.2.1.10 for the complete definition of this element.

EiTargetType (Class)

A set of elements to that collectively name who is participating or should participate in an EI interactions

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EiTargetType*. See Clause 5.7.1.2.1.11 for the complete definition of this element.

EmixInterfaceType (Abstract Class)

Abstract base class for the interfaces for EMIX Product delivery, measurement, and/or pricing.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EmixInterfaceType*. See Clause 5.7.2.13.1.3 for the complete definition of this element.

EnvelopeContentsType (Class)

EMIX Warrants are delivered as Envelope Contents. Assertions that distinguish the commodity product are called EMIX Warrants. A common definition of a warrant is a written assurance that some product or service will be provided or will meet certain specifications.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\EnvelopeContentsType*. See Clause 5.7.2.13.1.5 for the complete definition of this element.

EventDescriptorType (Class)

A description of the type of event being requested, indicating the market context and priority of the event as well as when the event was created, modified, and is scheduled to become active.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventDescriptorType*. See Clause 5.7.1.2.1.12 for the complete definition of this element.

EventSignalTypeBase (Abstract Class)

This class conveys the kind of demand response event being signaled and the desired resulting load or generation.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventSignalTypeBase*. See Clause 5.7.1.2.1.13 for the complete definition of this element.

EventStatusEnumeratedType (Enumeration)

Indicates the present status of an event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventStatusEnumeratedType*. See Clause 5.7.1.2.1.32 for the complete definition of this element.

EventStatusType (Class)

Indicates the present status of an event.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\EventStatusType*. See Clause 5.7.1.2.1.14 for the complete definition of this element.

FSGIMEventSignalType (Class)

This class conveys the kind of demand response event being signaled from an energy manager to a curtailable load or a dispatchable generator. The signal indicates the desired resulting load or generation.

For FSGIMEventSignalType, the multiplicity of eventInterval is 0.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\FSGIMEventSignalType*. See Clause 5.7.1.2.1.15 for the complete definition of this element.

GluonType (Class)

A GluonType influences the serialization of IntervalTypes in a sequence. A GluonType is often used as a pointer to the beginning interval in a sequence, although in general it can point to any interval in a sequence, or simply describe an isolated interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\GluonType*. See Clause 5.7.3.2.6 for the complete definition of this element.

IntervalType (Class)

This class represents an interval of time that attaches to a collection of data pertinent to the interval.

UML Element Location: *Model\Model Elements from External Sources\OASIS\WSCalendarPIM\IntervalType*. See Clause 5.7.3.2.7 for the complete definition of this element.

ItemBaseType (Abstract Class)

Abstract base type for units for EMIX Product delivery, measurement, and warrants. Item as in PO Item, Requisition Item, Invoice Item, Lading Item. Item does not include Quantity or Price, because a single product description or transaction may have multiple quantities or prices associated with a single item.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\ItemBaseType*. See Clause 6.6.4.3 for the complete definition of this element.

MonetaryQuantity (Class)

A monetary quantity used to express the cost or value of something in some currency.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\MonetaryQuantity*. See Clause 6.6.5.4 for the complete definition of this element.

PayloadApplicationSpecificType (Class)

This Payload contains the information that changes in each Stream payload.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadApplicationSpecificType*. See Clause 5.7.1.2.1.16 for the complete definition of this element.

PayloadBaseType (Abstract Class)

base for information in Signal / Baseline / Report Payloads

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadBaseType*. See Clause 5.7.1.2.1.17 for the complete definition of this element.

PayloadEmixType (Class)

This is the Payload for Signals that require an EMIX Product Description.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadEmixType*. See Clause 5.7.1.2.1.18 for the complete definition of this element.

PayloadFloatType (Class)

This is the Payload for Signals that require a Quantity.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadFloatType*. See Clause 5.7.1.2.1.19 for the complete definition of this element.

PayloadLevelType (Class)

This is the Payload for Signals that convey Simple Levels.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadLevelType*. See Clause 5.7.1.2.1.20 for the complete definition of this element.

PayloadPriceMultiplierType (Class)

A Price Multiplier that when applied to a reference price produces the actual price.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadPriceMultiplierType*. See Clause 5.7.1.2.1.21 for the complete definition of this element.

PayloadPriceRelativeType (Class)

A Price Relative is added to a reference price to compute the actual price. Price Relative may be positive or negative.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadPriceRelativeType*. See Clause 5.7.1.2.1.22 for the complete definition of this element.

PayloadPriceType (Class)

This is the Payload for Signals that require a Quantity.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadPriceType*. See Clause 5.7.1.2.1.23 for the complete definition of this element.

PayloadProductType (Class)

This Payload contains the information that changes in each Stream payload.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadProductType*. See Clause 5.7.1.2.1.24 for the complete definition of this element.

PayloadQuantityType (Class)

This is the Payload for Signals that require a Quantity.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\PayloadQuantityType*. See Clause 5.7.1.2.1.25 for the complete definition of this element.

PriceBaseType (Abstract Class)

Type of Abstract base for EMIX Prices

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceBaseType*. See Clause 5.7.2.13.1.8 for the complete definition of this element.

PriceMultiplierType (Class)

Multiplier times market price, 1 for same as market

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceMultiplierType*. See Clause 5.7.2.13.1.9 for the complete definition of this element.

PriceRelativeType (Class)

Price Relative is a fixed charge (positive or negative) applied to base price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceRelativeType*. See Clause 5.7.2.13.1.10 for the complete definition of this element.

PriceType (Class)

Simple Price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceType*. See Clause 5.7.2.13.1.11 for the complete definition of this element.

ProductDescriptionType (Abstract Class)

In EMIX, the Product Description is placed in the Interval or Gluon attachment. The respective product schemas extend this abstract class.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\ProductDescriptionType*. See Clause 5.7.2.13.1.12 for the complete definition of this element.

ReadingTypeEnumeratedType (Enumeration)

Indicates metadata about the Readings, i.e., direct measurement or computation.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ReadingTypeEnumeratedType*. See Clause 5.7.1.2.1.33 for the complete definition of this element.

ReadingTypeType (Class)

Type of Reading.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ReadingTypeType*. See Clause 5.7.1.2.1.26 for the complete definition of this element.

refID (Class)

Reference ID for a particular instance, transmittal, or artifact. Note: not the same as the native ID of the object being transmitted or shared.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\refID*. See Clause 5.7.1.2.1.27 for the complete definition of this element.

ReportSpecifierType (Class)

Parameters that define the content of a Report Stream

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\ReportSpecifierType*. See Clause 5.7.1.2.1.28 for the complete definition of this element.

Sequence (Class)

A sequence is a set of time related ordered intervals. These can represent a schedule of events or just a set of data related by time. Each interval of a sequence includes relative start time. The concatenation of these intervals forms a time-series. Missing data, for periodic time-series, can be surmised by monitoring the time references.

UML Element Location: *Model\Device and Model Components\Energy Manager Component\Sequences and Intervals\Sequence*. See Clause 5.6.7.10.4 for the complete definition of this element.

SignalTypeEnumeratedType (Enumeration)

SignalTypeEnumerated lists the pre-defined Types used to specify the Payload Types and conformance in a Stream

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SignalTypeEnumeratedType*. See Clause 5.7.1.2.1.34 for the complete definition of this element.

SignalTypeType (Class)

SignalType is used in EventSignals to specify the Payload Types in a Signal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SignalTypeType*. See Clause 5.7.1.2.1.29 for the complete definition of this element.

SimpleLevelContextType (Class)

Simple Levels are a set of simple indicators about scarcity and value, in which an ordered set of values indicate energy scarcity is above normal, normal, or below normal.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SimpleLevelContextType*. See Clause 5.7.1.2.1.30 for the complete definition of this element.

SpecifierPayloadType (Class)

Payload for use in Report Specifiers.

UML Element Location: *Model\Model Elements from External Sources\OASIS\Energy Interoperation wd36/PR03\EIClasses\SpecifierPayloadType*. See Clause 5.7.1.2.1.31 for the complete definition of this element.

StandardTermsType (Class)

Standard Terms express overarching terms for a market context

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\StandardTermsType*. See Clause 5.7.2.13.2.23 for the complete definition of this element.

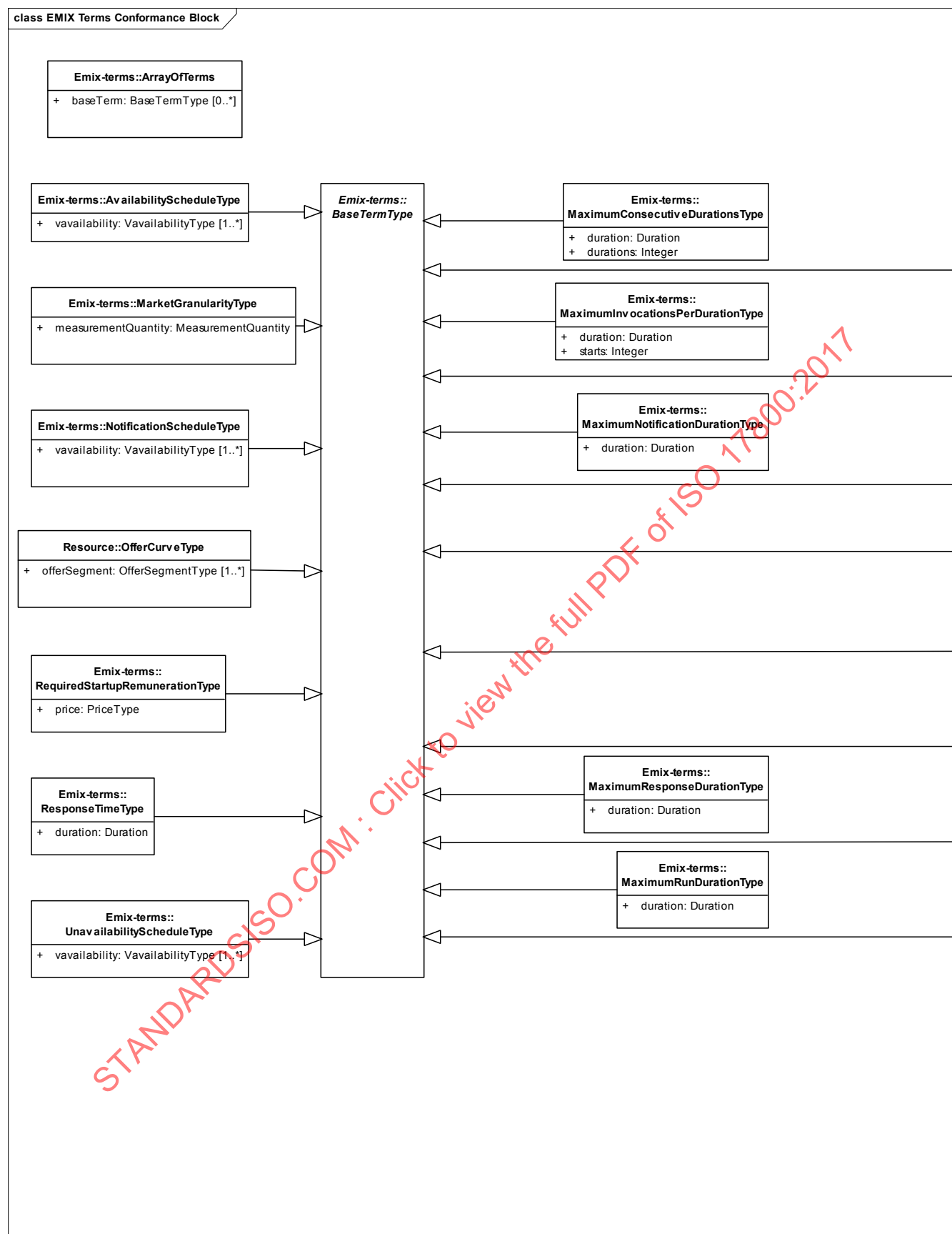
7.3.3.2.7. EMIX Terms Conformance Block Diagram

This diagram shows how the terms of sale for power and energy are modeled in EMIX.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017

This blank page has been inserted to improve the readability of the multi-page diagram that begins on the next page.

STANDARDSISO.COM : Click to view the full PDF of ISO 17800:2017



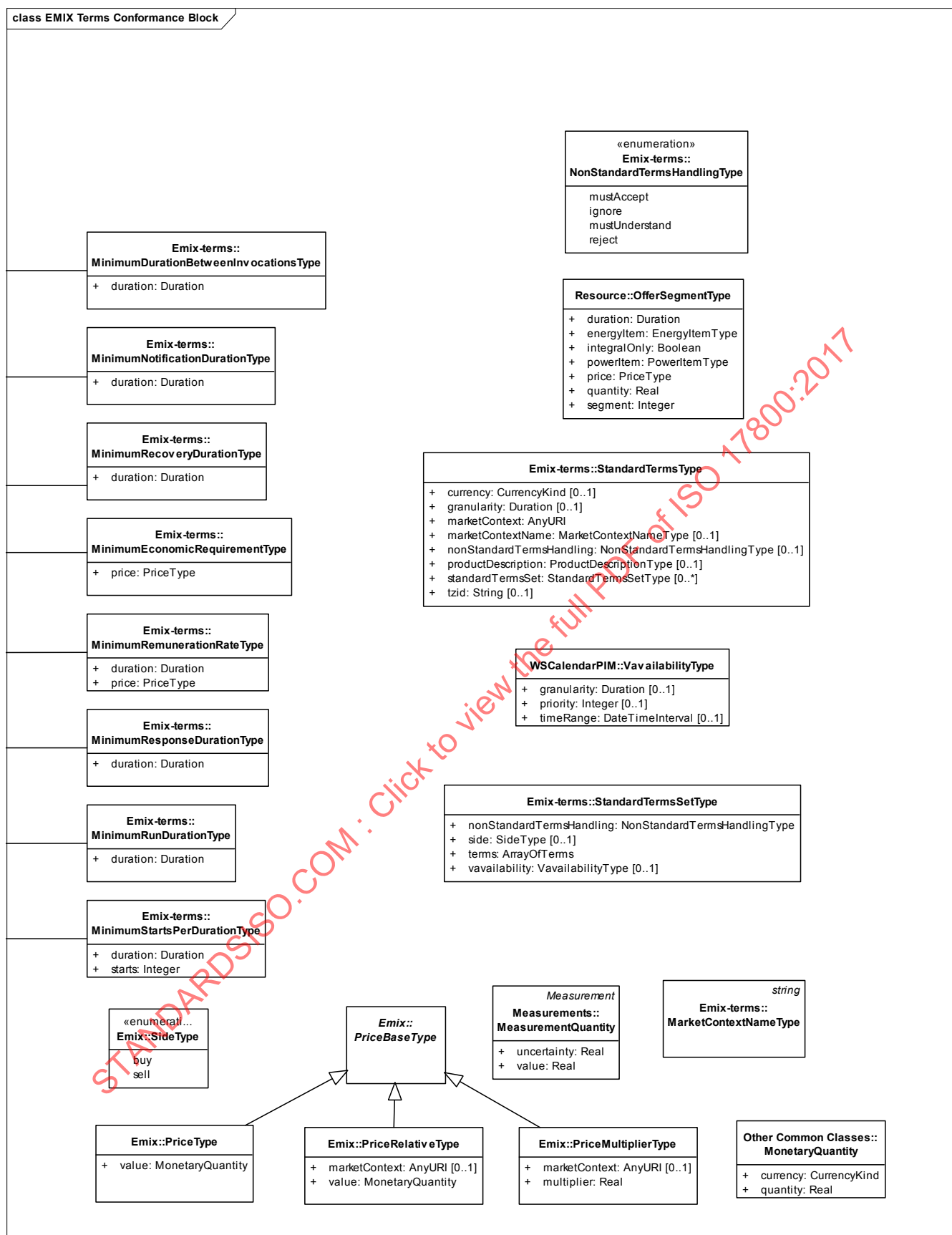


Figure 7.16 - EMIX Terms Conformance Block

The EMIX Terms Conformance Block Diagram makes use of the following elements:

ArrayOfTerms (Class)

Collection of Emix Terms

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\ArrayOfTerms*. See Clause 5.7.2.13.2.1 for the complete definition of this element.

AvailabilityScheduleType (Class)

A Schedule of time windows during which a response may be scheduled. A scheduled Duration must be entirely within a single instance of an availability window.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\AvailabilityScheduleType*. See Clause 5.7.2.13.2.2 for the complete definition of this element.

BaseTermType (Abstract Class)

Type of Abstract extension object for Emix Terms

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\BaseTermType*. See Clause 5.7.2.13.2.3 for the complete definition of this element.

MarketContextNameType (Class)

A descriptive name applied to any market context through the associated Standard Terms.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MarketContextNameType*. See Clause 5.7.2.13.2.4 for the complete definition of this element.

MarketGranularityType (Class)

Granularity defines the size of a market "bundle". For example, a Market with a granularity of 10 MW, will only accept tenders, process transactions, and pay for delivery of Power in multiples of 10 MW.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MarketGranularityType*. See Clause 5.7.2.13.2.5 for the complete definition of this element.

MaximumConsecutiveDurationsType (Class)

Maximum consecutive Durations in which a notification will be accepted, e.g., it will not accept requests on more than three consecutive days.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumConsecutiveDurationsType*. See Clause 5.7.2.13.2.6 for the complete definition of this element.

MaximumInvocationsPerDurationType (Class)

Maximum number of requests for response that will be accepted during a Duration.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumInvocationsPerDurationType*. See Clause 5.7.2.13.2.7 for the complete definition of this element.

MaximumNotificationDurationType (Class)

The maximum Duration in advance of a proposed response that a notification will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumNotificationDurationType*. See Clause 5.7.2.13.2.8 for the complete definition of this element.

MaximumResponseDurationType (Class)

The longest Duration for which a request will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumResponseDurationType*. See Clause 5.7.2.13.2.9 for the complete definition of this element.

MaximumRunDurationType (Class)

Maximum acceptable Duration for a proposed response

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MaximumRunDurationType*. See Clause 5.7.2.13.2.10 for the complete definition of this element.

MeasurementQuantity (Class)

The quantity portion of a generic quantity/metadata pair used to represent types of measurements.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Measurements\MeasurementQuantity*. See Clause 6.6.4.5 for the complete definition of this element.

MinimumDurationBetweenInvocationsType (Class)

The minimum Duration between successive responses that will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumDurationBetweenInvocationsType*. See Clause 5.7.2.13.2.11 for the complete definition of this element.

MinimumEconomicRequirementType (Class)

Minimum net remuneration for any single response.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumEconomicRequirementType*. See Clause 5.7.2.13.2.12 for the complete definition of this element.

MinimumNotificationDurationType (Class)

The minimum Duration in advance of a proposed response that a notification will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumNotificationDurationType*. See Clause 5.7.2.13.2.13 for the complete definition of this element.

MinimumRecoveryDurationType (Class)

The minimum Duration after completion of a response before a new response can be begun.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRecoveryDurationType*. See Clause 5.7.2.13.2.14 for the complete definition of this element.

MinimumRemunerationRateType (Class)

Minimum remuneration acceptable per stated Duration of response, e.g., a minimum remuneration of \$100 / hour.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRemunerationRateType*. See Clause 5.7.2.13.2.15 for the complete definition of this element.

MinimumResponseDurationType (Class)

The shortest Duration for which a request will be accepted.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumResponseDurationType*. See Clause 5.7.2.13.2.16 for the complete definition of this element.

MinimumRunDurationType (Class)

Minimum acceptable Duration for a proposed response.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumRunDurationType*. See Clause 5.7.2.13.2.17 for the complete definition of this element.

MinimumStartsPerDurationType (Class)

The fewest requests that the resource will accept during any Duration.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\MinimumStartsPerDurationType*. See Clause 5.7.2.13.2.18 for the complete definition of this element.

MonetaryQuantity (Class)

A monetary quantity used to express the cost or value of something in some currency.

UML Element Location: *Model\Common Primitive Types, Classes, and Enumerations\Other Common Classes\MonetaryQuantity*. See Clause 6.6.5.4 for the complete definition of this element.

NonStandardTermsHandlingType (Enumeration)

Defines how participants should handle terms not listed in the RuleSet

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\NonStandardTermsHandlingType*. See Clause 5.7.2.13.2.25 for the complete definition of this element.

NotificationScheduleType (Class)

A schedule of time windows during which requests can be made.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\NotificationScheduleType*. See Clause 5.7.2.13.2.19 for the complete definition of this element.

OfferCurveType (Class)

A collection of Offer Segments used to compute cost requirements across a range of power.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\OfferCurveType*. See Clause 5.7.2.13.6.4 for the complete definition of this element.

OfferSegmentType (Class)

Marginal offer for Power within a range. Marginal costs must be computed within the context of a range of segments as conformed by the Offer Type.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Resource\OfferSegmentType*. See Clause 5.7.2.13.6.5 for the complete definition of this element.

PriceBaseType (Abstract Class)

Type of Abstract base for EMIX Prices

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceBaseType*. See Clause 5.7.2.13.1.8 for the complete definition of this element.

PriceMultiplierType (Class)

Multiplier times market price, 1 for same as market

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceMultiplierType*. See Clause 5.7.2.13.1.9 for the complete definition of this element.

PriceRelativeType (Class)

Price Relative is a fixed charge (positive or negative) applied to base price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceRelativeType*. See Clause 5.7.2.13.1.10 for the complete definition of this element.

PriceType (Class)

Simple Price

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix\PriceType*. See Clause 5.7.2.13.1.11 for the complete definition of this element.

RequiredStartupRemunerationType (Class)

Minimum remuneration required for initiating a response.

UML Element Location: *Model\Model Elements from External Sources\OASIS\EMIX PR04\Emix-terms\RequiredStartupRemunerationType*. See Clause 5.7.2.13.2.20 for the complete definition of this element.