
**Automation systems and
integration — Interoperability of
capability units for manufacturing
application solutions —**

**Part 1:
Interoperability criteria of capability
units per application requirements**

*Systèmes d'automatisation et intégration — Interopérabilité des
unités de capacité pour les solutions d'applications industrielles —*

*Partie 1: Critères d'interopérabilité des unités de capacité basés sur
les exigences liées aux applications*



STANDARDSISO.COM : Click to view the full PDF of ISO 16300-1:2018



COPYRIGHT PROTECTED DOCUMENT

© ISO 2018

All rights reserved. Unless otherwise specified, or required in the context of its implementation, 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
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

	Page
Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Interoperability and integration views for automation applications	3
4.1 Interoperability concepts for an automation system.....	3
4.2 Integration concepts for a manufacturing application of an automation system.....	3
4.3 Capability element.....	4
5 Relationship and structure of ISO 16300 series	4
6 Interoperability criteria of a manufacturing application	5
Annex A (informative) Object-Process Methodology overview	7
Annex B (informative) Reference models for application domains	8
Bibliography	11

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. This document was 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 of 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 www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 5, *Interoperability, integration, and architectures for enterprise systems and automation applications*.

A list of all parts in the ISO 16300 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

ISO 16300 addresses requirements of users and suppliers of manufacturing software regarding the interoperability of software in the area of industrial automation.

User interoperability requirements include:

- integrating an automation application system by combining capabilities of a set of software components provided by various sources;
- substituting another software component in a software unit to provide an equivalent capability required by the automation application system;
- integrating the capability of a software unit from one resource system platform to another platform;
- validating and verifying the capability of a software unit to meet the automation application system requirements.

Supplier requirements include:

- representing the set of capabilities provided by a software component used in a software unit;
- verifying software component capability as a part of a required software unit capability;
- cataloguing a software unit in terms of its capability for interoperability in an automation application system to support wide distribution.

ISO 16300 also addresses software interoperability services, which include:

- access to the description of a software capability to enable interoperability assessment;
- enabling the search and location of candidate software units and components, preferably automatically, using search engines;
- representing the dependencies between software components for an automation application hosted on a particular system platform.

Software capability is first defined as a set of functions and services with a set of criteria for evaluating the performance of a capability provider. It is then expressed and represented as facts about the software, how and what it can do. The ISO 16100 series on manufacturing software capability profiling for interoperability was developed with the aim of providing a standardized method to describe capabilities of manufacturing software in terms of the manufacturing software unit (MSU) capability profile. In ISO 16100, the software component is included in the MSU. ISO 16100 also provides a way to exchange an MSU's capability as information by means of a capability profile. Software capability profiling is the basis for providing the above-mentioned software interoperability services. ISO 16100 is used and applied as the foundation for ISO 16300.

To establish ISO 16300, a number of steps were required. The initial step shows what interoperability services are enabled by using software capability profiles. The following steps develop concrete methods and mechanisms to provide these interoperability services.

This document (ISO 16300-1) specifies a framework for describing an automation solution in terms of a set of capabilities provided by a set of MSUs. The framework also defines a set of capability elements and composition rules to represent the interoperability criteria in terms of the automation system capability requirements of an enterprise application.

ISO 16300-2 specifies the template definition to describe the capability of a software unit of an automation solution that can be mapped to the functional requirements of a target manufacturing application. It also specifies mapping rules for composing the contents of a software unit catalogue item in terms of the properties of the capability.

ISO 16300-3 specifies the framework for verifying interoperability of capability units associated with application requirements and system solutions.

ISO 16300-4 specifies the search methodology for acquiring candidate capability units which satisfy the manufacturing application requirements from the software unit catalogues and also describes the structure of the report as an outcome of the search indicating the extent to which the candidates from the software unit catalogues correspond to the manufacturing application requirements.

STANDARDSISO.COM : Click to view the full PDF of ISO 16300-1:2018

Automation systems and integration — Interoperability of capability units for manufacturing application solutions —

Part 1:

Interoperability criteria of capability units per application requirements

1 Scope

This document specifies a framework for describing an automation solution in terms of a set of capabilities provided by a set of manufacturing software units.

The framework also defines a set of capability elements and composition rules to represent the interoperability criteria in terms of the automation system capability requirements of an enterprise application.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 16100-1:2009, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 1: Framework*

ISO 16100-2:2003, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 2: Profiling methodology*

ISO 16100-3, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 3: Interface services, protocols and capability templates*

ISO 16100-4, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 4: Conformance test methods, criteria and reports*

ISO 16100-5, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 5: Methodology for profile matching using multiple capability class structures*

ISO 16100-6, *Industrial automation systems and integration — Manufacturing software capability profiling for interoperability — Part 6: Interface services and protocols for matching profiles based on multiple capability class structures*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 16100-1 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <http://www.electropedia.org/>

3.1

capability

<software> set of functions and services with a set of criteria for evaluating the performance of a capability provider

Note 1 to entry: This definition differs from that given in ISO 15531-1 and ISO 19439, where capability is defined as the quality of being able to perform a given activity. See IEC 62264-1 for a general definition of capability.

[SOURCE: ISO 16100-1:2009, 3.4]

3.2

capability class

element within the *capability profiling* (3.5) method that represents software unit functionality and behaviour with regard to the software unit's role in a manufacturing activity

[SOURCE: ISO 16100-2:2003, 3.3]

3.3

capability element

element used to indicate that a particular *capability* (3.1) is supported by the entity or *manufacturing software unit* (3.9) to which the element belongs

3.4

capability profile

instance of a *capability template* (3.6) filled with the values corresponding to the target *manufacturing software unit* (3.9)

3.5

capability profiling

selection of a set of offered services defined by a particular interface within a software *interoperability* (3.8) framework

[SOURCE: ISO 16100-1:2009, 3.5]

3.6

capability template template

schema representing a *capability class* (3.2)

3.7

capability unit

unit of a type intended to support the execution of a particular task

Note 1 to entry: Examples of types include mechanical, electrical, electronic, hardware and software.

3.8

interoperability

capability (3.1) of two or more entities to exchange items in accordance with a set of rules and mechanisms implemented by an interface in each entity, in order to perform their respective tasks

Note 1 to entry: Examples of entities include devices, equipment, machines, people, processes, applications, software units, systems and enterprises.

Note 2 to entry: Examples of items include information, material, energy, control, assets and ideas.

[SOURCE: ISO 18435-1:2009, 3.12]

3.9 manufacturing software unit MSU

class of software resource, consisting of one or more manufacturing software components, performing a definite function or role within a manufacturing activity while supporting a common information exchange mechanism with other units

Note 1 to entry: A software unit can be modelled using UML as a software object.

[SOURCE: ISO 16100-1:2009, 3.18, modified — The abbreviated term “MSU” has been added.]

4 Interoperability and integration views for automation applications

4.1 Interoperability concepts for an automation system

The concept of interoperability involves two or more entities that can exchange or share certain items in order to perform their respective activities. The entities can be devices, machines, people, software units, systems, applications or even enterprises. The items being exchanged or shared can be material, information, energy or assets. Items are exchanged according to a set of rules and mechanisms implemented by an interface in each entity.

The rules and mechanisms to perform the exchanges or sharing are characterized by interfaces that are enabled by the entities. The entities have a common understanding of the properties of the items exchanged. [Figure 1](#) shows the interoperability of two automation entities.

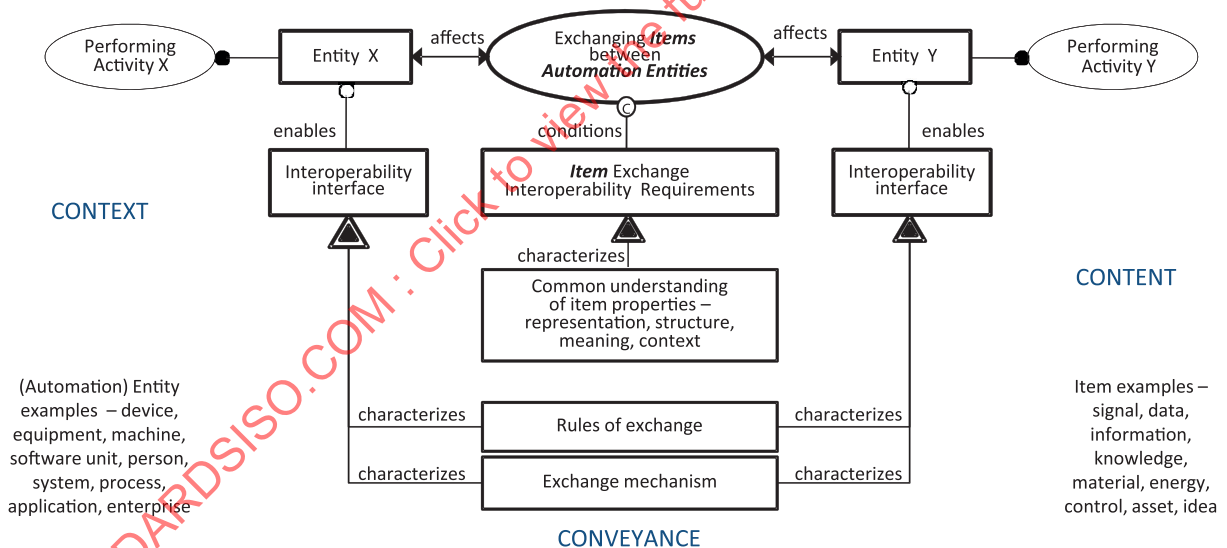


Figure 1 — Interoperability of two automation entities

EXAMPLE Three historian servers each have a different operating system: Windows, Mac OS X, Linux. All can exchange data items among their respective historian servers using a messaging interface (on some TCP/IP/Ethernet link) based on Open Platform Communications Unified Architecture (OPC-UA) as a conveyance. The historian servers are interoperable and non-interfering.

This document focuses on the cases where MSUs are used to represent entities.

4.2 Integration concepts for a manufacturing application of an automation system

The concept of integration involves two or more component entities that form a system or a single entity performing a certain activity. Each component entity has a structure, a behaviour and a boundary that is distinct from the structure, behaviour and boundary exhibited by the system. An integrated system is perceived to perform an activity as a single entity; however, its component entities collaborate,

coordinate, and interoperate via exchanges of items to accomplish the system's task. These relationships regarding entities and activities can be described using Object-Process Methodology (OPM). [Annex A](#) gives an overview of OPM. [Annex B](#) describes the application integration and related interoperability concepts using OPM diagrams.

EXAMPLE From an external client point of view, a data item request at a user interface is processed without the client's knowledge of three different historian servers that collaborate and coordinate to service the external request. The three historian servers are considered to be well-integrated if the response to the request is complete and timely.

Clearly, interoperability is a pre-requisite to integration. These concepts are applied iteratively as systems are composed of components that can be considered as systems at a more detailed level.

[Figure 2](#) illustrates the integration of automation entities in a system.

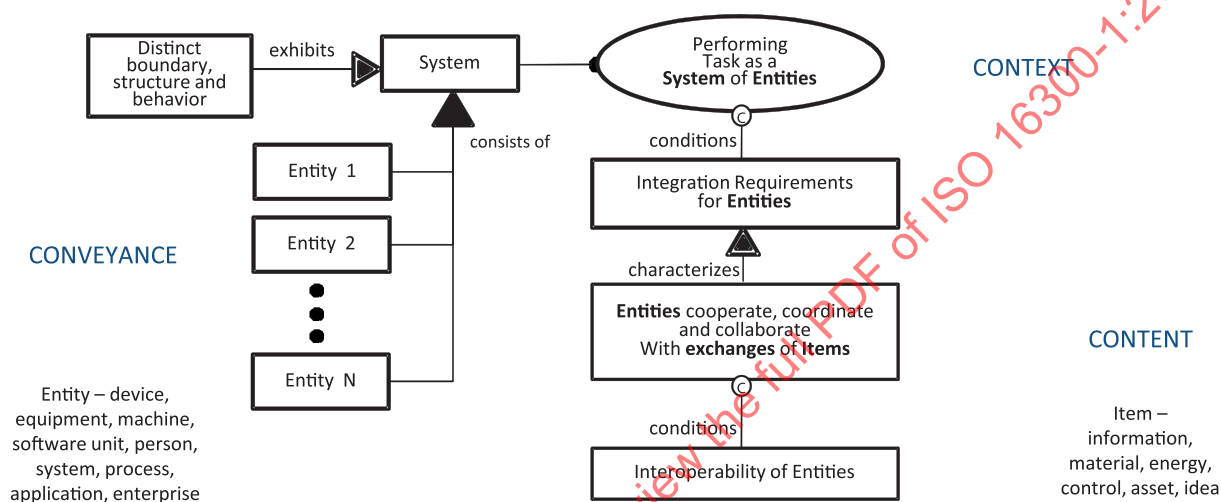


Figure 2 — Integration of automation entities in a system

4.3 Capability element

The capability element indicates a particular capability supported by the entities, or MSUs, to which the element belongs. These elements are utilized in the automation solutions or in the automation system capability requirements. The values of the capability element are the names of the capabilities to be supported or required by the capability units.

Each capability unit shall contain the necessary capability elements to express the capabilities. This set of capability elements includes the interoperability criteria, which are based on the semantics, syntax and contexts of the exchanges among capability units. Capability elements shall be described using the MDDs in the specific parts of the capability profiles defined in the ISO 16100 series.

5 Relationship and structure of ISO 16300 series

Manufacturing software is constructed by the combination of MSUs. The ISO 16300 series focuses on how to develop manufacturing applications through MSU interoperability. A traditional V-model of the development procedure for an application system is shown in [Figure 3](#). The ISO 16100 series provides the methodology to implement an application system using MSU's capability profiling.

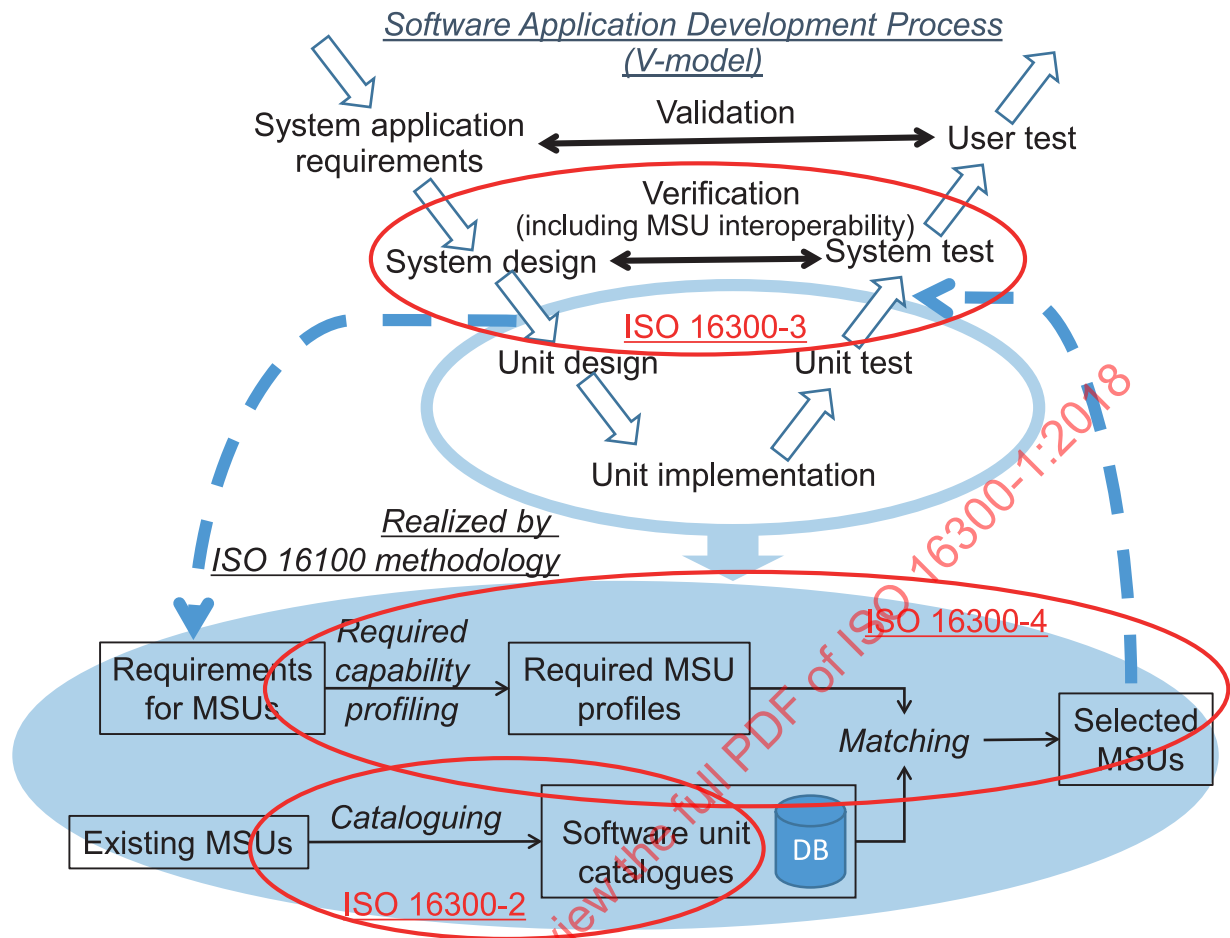


Figure 3 — Manufacturing application development procedure

In the ISO 16100 series, the existing MSU capability profiles are registered in an appropriate database. Required MSU capability profiles for development of new applications are compared to existing profiles in the database. The matching procedure searches for adequate MSUs through the catalogues which include MSU capability profiles. When a match occurs, the profiled MSU is considered as a candidate. By repeating these matching procedures, all MSUs for an application are selected. The assessment procedure is performed to select candidate MSUs resulting from the matching procedure.

Figure 3 also shows the relationship of ISO 16300-2, ISO 16300-3 and 16300-4 in developing manufacturing applications. Before the unit test stage, a matching procedure performs the verification of MSUs. At the system test stage, the application constructed by the combination of MSUs is verified, including MSU interoperability. At the user test stage, the developed manufacturing application is validated by referring to the automation system capability requirements.

6 Interoperability criteria of a manufacturing application

Manufacturing processes in manufacturing software are realized by manufacturing resources in the particular manufacturing software. In real industrial settings, manufacturing processes and manufacturing resources depend on each other, and deciding on the optimum processes and resources requires balancing (or conducting trade-offs between) conflicting objectives.

This decision making process is outside the scope of this document, which only points out that the relation exists between process and resource. Product and material are also outside the scope of this document.

To realize a manufacturing process by a particular manufacturing resource, activities shall be derived from the manufacturing application requirements for the desired automation solution. An automation solution is a collection of manufacturing software which satisfies the application requirements. An automation solution shall be modelled as an activity tree structure, where each activity can be associated with a capability class that is provided by an MSU. Each capability class is represented by the MSU with appropriate capability elements.

An MSU is modelled as a type of manufacturing resource that can satisfy a set of interoperability criteria. These criteria are determined by the specific set of manufacturing functions that need to be accomplished by an MSU and the information exchanges that it needs to support.

A particular class of MSU may be used in different activities. Each MSU shall provide a set of interfaces for supporting these activities. The interoperability criteria between MSUs shall be determined only by the requirements of the interoperable activities.

When two or more MSUs provide the required manufacturing function within a manufacturing activity, these MSUs shall satisfy a set of interoperability criteria. The required interface(s) for interoperability of a set of MSUs within a particular activity shall be described using a set of capability elements.

Interoperability criteria are represented by reusability in ISO 16300-3. These interoperability criteria are determined by matching and assessing capability profiles, which are defined in ISO 16300-2 and ISO 16300-4. Reusability includes data exchangeability, interchangeability, requirement adequacy and platform agnosticism.

STANDARDSISO.COM : Click to view the full PDF of ISO 16300-1:2018

Annex A

(informative)

Object-Process Methodology overview

Object-Process Methodology (OPM) is a minimal-length conceptual approach, language, and methodology for modelling and knowledge representation of complex multidisciplinary dynamic systems, suitable for implementation and support by information and computer technology facilitated tools.

OPM provides a graphical representation and a textual representation for the same model. A set of interrelated diagrams of a single kind, called Object-Process Diagram (OPD), is the visual expression. Each OPD consists of OPM elements depicted as graphic symbols. The OPD syntax specifies the consistent and correct ways by which those elements can be managed graphically.

A set of automatically-generated sentences in an Object-Process Language (OPL) constitute the textual model. The corresponding textual model in OPL is generated in a subset of English in a manner that retains the constraints of the graphical model. Since OPL's syntax and semantics are a subset of English natural language, the textual model is easily understood by domain experts, who need not learn any special language nor decipher cryptic code.

OPM is a systems modelling paradigm that represents its structure and behaviour. OPM is a formal and intuitive paradigm for systems architecting, engineering, development, life cycle support, communication, and evolution. OPM notation supports conceptual modelling of systems. OPM's top-down approach includes refinement mechanisms of in-zooming and unfolding.

OPM uses a single kind of OPD to describe the functional, structural and behavioural aspects of the system. A set of OPDs simplifies the expression of a complex system. Each OPD is obtained by in-zooming or unfolding of a thing (object or process) in its root OPD.

Copies of an existing thing can be placed in any diagram, where some or all the details, such as object states or links to other things, which are unimportant in the context of the diagram, can be omitted. It is sufficient for some detail to appear once in some OPD for it to be true for the system in general even though it is not shown in any other OPD.

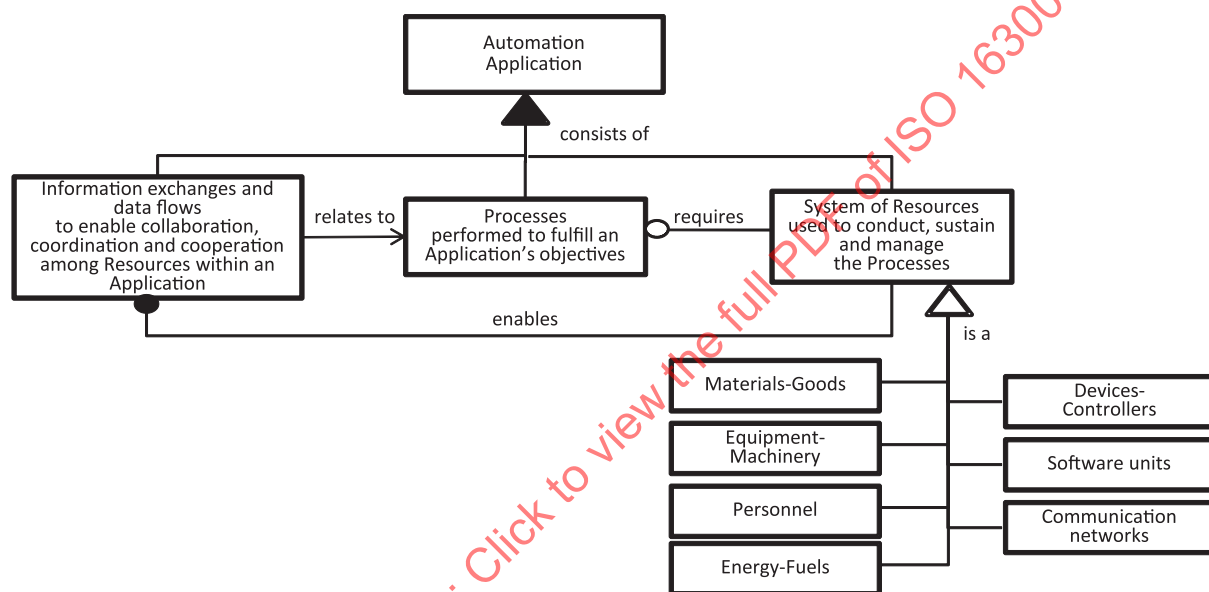
OPM has a formal syntax and semantics that serve as the basis for model-based systems engineering in general and model-based document authoring and evolution in particular. OPM can be used to sketch the implementation of a standard, and this sketch can identify weaknesses in the standard itself and correct these weaknesses to achieve consistency and completeness. OPM enables the model based standard document to become of sufficient quality to allow uniform compliance.

Annex B (informative)

Reference models for application domains

B.1 General

The entities for a specific application and the relationships between the entities and resources are shown in [Figure B.1](#). Each component of the resource is constructed using the automation system interfaces between the resources identified in each of the associated resource models.



References: ISO 15745, ISO 19450

Figure B.1 — Entities within an automation application

B.2 Related interoperability concepts representation

Following IEC 62264 definitions and terms, the Manufacturing Operations Management (MOM) activity domain at Level 3 can be represented as a group of OPM processes that exchange information with the business systems at Level 4. Each manufacturing operation category denoted by the green OPM objects in [Figure B.2](#) has a corresponding process diagram.

In the OPM diagram, there are eight generic processes, ranging from defining the operation's objective through tracking and analysing the performance of the operation through the completion of its objective. These processes are the main areas where improvements are expected to be provided by automation productivity. Of course, the interoperability requirements of the applications within the activity domain can be further modelled. In principle, the applications in one MOM category can exchange items with applications in another MOM category.

[Figure B.2](#) shows the OPM process model of managing a particular manufacturing operations category.