IEC 61499-2

Edition 2.0 2012-11

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

Function blocks –
Part 2: Software tool requirements

Blocs fonctionnels –
Partie 2: Exigences pour les outils logiciels

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2012 IEC, Geneva, Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either IEC or IEC's member National Committee in the country of the requester.
If you have any questions about IEC copyright or have an enquiry about obtaining additional rights to this publication, please contact the address below or your local IEC member National Committee for further information.

Droits de reproduction réservés. Sauf indication contraire, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de la CEI ou du Comité national de la CEI du pays du demandeur.
Si vous avez des questions sur le copyright de la CEI ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de la CEI de votre pays de résidence.

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland

Tel.: +41 22 919 02 11
Fax: +41 22 919 03 00
info@iec.ch
www.iec.ch

About the IEC

The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

About IEC publications

The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigenda or an amendment might have been published.

Useful links:

IEC publications search - www.iec.ch/searchpub

The advanced search enables you to find IEC publications by a variety of criteria (reference number, text, technical committee,…).
It also gives information on projects, replaced and withdrawn publications.

IEC Just Published - webstore.iec.ch/justpublished

Stay up to date on all new IEC publications. Just Published details all new publications released. Available on-line and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing more than 30 000 terms and definitions in English and French, with equivalent terms in additional languages. Also known as the International Electrotechnical Vocabulary (IEV) on-line.

Customer Service Centre - webstore.iec.ch/csc

If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: csc@iec.ch.

A propos de la CEI

La Commission Electrotechnique Internationale (CEI) est la première organisation mondiale qui élabore et publie des Normes internationales pour tout ce qui a trait à l'électricité, à l'électronique et aux technologies apparentées.

A propos des publications CEI

Le contenu technique des publications de la CEI est constamment revu. Veuillez vous assurer que vous possédez l'édition la plus récente, un corrigendum ou amendement peut avoir été publié.

Liens utiles:

Recherche de publications CEI - www.iec.ch/searchpub

La recherche avancée vous permet de trouver des publications CEI en utilisant différents critères (numéro de référence, texte, comité d'études,…).
Elle donne aussi des informations sur les projets et les publications remplacées ou retirées.

Just Published CEI - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications de la CEI. Just Published détaille les nouvelles publications parues. Disponible en ligne et aussi une fois par mois par email.

Electropedia - www.electropedia.org

Le premier dictionnaire en ligne au monde de termes électroniques et électriques. Il contient plus de 30 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans les langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (VEI) en ligne.

Service Clients - webstore.iec.ch/csc

Si vous désirez nous donner des commentaires sur cette publication ou si vous avez des questions contactez-nous: csc@iec.ch.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

IEC 61499-2

Edition 2.0 2012-11

INTERNATIONAL STANDARD

NORME INTERNATIONALE

colour inside

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Function blocks –
Part 2: Software tool requirements

Blocs fonctionnels –
Partie 2: Exigences pour les outils logiciels

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

PRICE CODE
CODE PRIX X

ICS 25.040; 35.240.50

ISBN 978-2-83220-478-8

Warning! Make sure that you obtained this publication from an authorized distributor.
Attention! Veuillez vous assurer que vous avez obtenu cette publication via un distributeur agréé.

® Registered trademark of the International Electrotechnical Commission
Marque déposée de la Commission Electrotechnique Internationale

CONTENTS

FOREWORD 3
INTRODUCTION 5
1 Scope 6
2 Normative references 6
3 Terms and definitions 6
4 Software tool requirements 7
4.1 Information to be provided by the software tool supplier 7
4.2 Exchange of library elements 7
4.3 Information to be provided by the supplier of library elements 7
4.4 Display of declarations 7
4.5 Modification of declarations 8
4.6 Validation of declarations 8
4.7 Implementation of declarations 8
4.8 System operation, testing and maintenance 8
Annex A (normative) Document type definitions (DTDs) 9
Annex B (informative) Graphics model 26
Annex C (informative) Examples 29
Bibliography 47

Figure B.1 – Graphics model 26
Figure B.2 – ECC drawing example 28

Table A.1 – Document type definition (DTD) elements 9
Table A.2 – DataType DTD (1 of 2) 10
Table A.3 – DataType DTD elements (1 of 2) 12
Table A.4 – Library Element DTD (1 of 5) 15
Table A.5 – LibraryElement DTD elements (1 of 5) 20

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

INTERNATIONAL ELECTROTECHNICAL COMMISSION

FUNCTION BLOCKS –

Part 2: Software tool requirements

FOREWORD

1) The International Electrotechnical Commission (IEC) is a worldwide organization for standardization comprising all national electrotechnical committees (IEC National Committees). The object of IEC is to promote international co-operation on all questions concerning standardization in the electrical and electronic fields. To this end and in addition to other activities, IEC publishes International Standards, Technical Specifications, Technical Reports, Publicly Available Specifications (PAS) and Guides (hereafter referred to as "IEC Publication(s)"). Their preparation is entrusted to technical committees; any IEC National Committee interested in the subject dealt with may participate in this preparatory work. International, governmental and non-governmental organizations liaising with the IEC also participate in this preparation. IEC collaborates closely with the International Organization for Standardization (ISO) in accordance with conditions determined by agreement between the two organizations.

2) The formal decisions or agreements of IEC on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC National Committees.

3) IEC Publications have the form of recommendations for international use and are accepted by IEC National Committees in that sense. While all reasonable efforts are made to ensure that the technical content of IEC Publications is accurate, IEC cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.

4) In order to promote international uniformity, IEC National Committees undertake to apply IEC Publications transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC Publication and the corresponding national or regional publication shall be clearly indicated in the latter.

5) IEC itself does not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC marks of conformity. IEC is not responsible for any services carried out by independent certification bodies.

6) All users should ensure that they have the latest edition of this publication.

7) No liability shall attach to IEC or its directors, employees, servants or agents including individual experts and members of its technical committees and IEC National Committees for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this IEC Publication or any other IEC Publications.

8) Attention is drawn to the Normative references cited in this publication. Use of the referenced publications is indispensable for the correct application of this publication.

9) Attention is drawn to the possibility that some of the elements of this IEC Publication may be the subject of patent rights. IEC shall not be held responsible for identifying any or all such patent rights.

International Standard IEC 61499-2, has been prepared by subcommittee 65B: Measurement and control devices, of IEC technical committee 65: Industrial-process measurement, control and automation.

This second edition cancels and replaces the first edition published in 2005. This edition constitutes a technical revision.

This edition includes the following significant technical changes with respect to the previous edition:

- The contents of Annex A have been updated to conform to the technical changes of the second edition of IEC 61499-1.
- CDATA sections are now allowed for the textual contents of algorithms in Tables A.4 and A.5.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

The text of this standard is based on the following documents:

FDIS	Report on voting
65B/846/FDIS	65B/856/RVD

Full information on the voting for the approval of this standard can be found in the report on voting indicated in the above table.

This publication has been drafted in accordance with the ISO/IEC Directives, Part 2.

A list of all parts of the IEC 61499 series can be found, under the general title *Function blocks*, on the IEC website.

Terms used throughout this International Standard that have been defined in Clause 3 of IEC 61499-1:2012 and in this International Standard appear in *italics*.

The committee has decided that the contents of this publication will remain unchanged until the stability date indicated on the IEC web site under "http://webstore.iec.ch" in the data related to the specific publication. At this date, the publication will be

- reconfirmed,
- withdrawn,
- replaced by a revised edition, or
- amended.

IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

INTRODUCTION

IEC 61499 consists of the following parts, under the general title *Function blocks*:

- Part 1: Architecture
- Part 2: Software tool requirements
- Part 3: Tutorial information (withdrawn)
- Part 4: Rules for compliance profiles

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

FUNCTION BLOCKS –

Part 2: Software tool requirements

1 Scope

This part of IEC 61499 defines requirements for *software tools* to support the following systems engineering tasks enumerated in IEC 61499-1:

– the specification of *function block types*;

– the functional specification of *resource types* and *device types*;

– the specification, analysis, and validation of distributed IPMCSs;

– the *configuration*, *implementation*, operation, and maintenance of distributed IPMCSs;

– the exchange of *information* among *software tools*.

It is assumed that such software tools may be used in the context of an Engineering Support System (ESS) as described in IEC 61499-1.

It is beyond the scope of this standard to specify the entire life cycle of industrial-process measurement and control systems (IPMCSs), or the entire set of tasks and activities required to support an IPCMS over its life cycle. However, other standards which do specify such tasks and activities may extend or modify the requirements specified in this part of IEC 61499.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

IEC 61131-3:2003, *Programmable controllers – Part 3: Programming languages*

IEC 61499-1:2012, *Function blocks – Part 1: Architecture*

ISO/IEC 8824 (all parts), *Information technology – Abstract Syntax Notation One (ASN.1)*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 61499-1, as well as the following apply.

3.1
library element
collection of *declarations* applying to a *data type, function block type, adapter type, subapplication type, resource type, device type, segment type,* or *system configuration*

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

4 Software tool requirements

4.1 Information to be provided by the software tool supplier

This Clause defines the functional requirements of *software tools* that support the performance of the systems engineering tasks enumerated in Clause 1.

The supplier of a *software tool* shall specify the following information in addition to other information required in this Clause:

- the type or types of *library element* to which the software tool applies;
- the engineering task or tasks supported by the software tool. Task descriptions may be taken from the enumeration of engineering tasks given in Clause 1, or may be defined by the supplier.

4.2 Exchange of library elements

A *software tool* shall be capable of exchanging its *library elements* with other software tools. This exchange shall take the form of *data* in the format defined in Annex A, written on physical media or exchanged over communication links or networks.

4.3 Information to be provided by the supplier of library elements

NOTE The provisions of this subclause are intended to provide the means by which the provider of a library element may achieve protection of intellectual property while still providing sufficient information to permit the effective use of the library element.

The provider of a *library element* may elect to provide an *implementation* of the library element.

EXAMPLE 1 The provider of a *function block type* library element may provide an implementation of the function block type as:

- one or more *instances* of the function block type in a *resource* contained in a *device* of Class 0 or higher as described in IEC 61499-4;
- an instantiable implementation of the function block type in a *resource* contained in a *device* of Class 1 or higher as described in IEC 61499-4;
- a file in an **implementation-dependent** format suitable for installation in a *resource* contained in a *device* of Class 2 as described in IEC 61499-4, for instance using the XML syntax defined in Annex D.

When an implementation of a library element is provided, the provider is not required to provide full details of the implementation. However, the provider shall provide sufficient information to enable the user to fully determine the functionality of the provided library element.

EXAMPLE 2 The requirement of the above paragraph would be met by the provider of an *instance* of a function block *type* in a *resource* through the provision, at a minimum, of the following information:

- a *function block type* library element specifying its *event interfaces, data interfaces* and *services* as defined in IEC 61499-1;
- *resource type* and *device type* library elements showing the occurrence and connections of the function block *instances*.

4.4 Display of declarations

A software tool shall be capable of displaying the *declarations* of its associated *library elements* in a form appropriate to the engineering task. This display may utilize the graphical or textual formats defined in IEC 61499-1, or a format defined by the supplier of the software tool.

NOTE The *declarations* of a library element may define its *interfaces* (event and data inputs and outputs) and internal *variables* as well as its *algorithms* and the control of their *execution*, for example via an *execution control chart* (ECC), etc.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Software tools may provide additional features, beyond those illustrated in IEC 61499-1, in the graphic display of declarations.

EXAMPLE 1 In the display of an Execution Control Chart (ECC), the tool may provide, along with the display of each transition, a cardinal number indicating the order (as defined in IEC 61499-1) in which the transition is evaluated.

EXAMPLE 2 A software tool may provide means of navigating a *mapping* from the display of a function block instance in an *application* to its corresponding display in a *resource*, and vice versa.

4.5 Modification of declarations

A software tool shall enable its user to modify the declarations of its associated library elements as appropriate to the engineering task. Such modifications may include adding, deleting or changing the contents of declarations, and may be performed either graphically or textually or both.

EXAMPLE The software tool may provide convenient means for the user to change the order in which declarations are listed in their textual representation, for instance in a list of transitions in an Execution Control Chart (ECC), without the user having to edit the textual representation by manual means such as "cut and paste".

4.6 Validation of declarations

If required by the associated engineering task, a software tool shall provide facilities for validation of the declarations of its associated library elements. Such facilities may include, but are not limited to:

- Checking the correctness of the syntax of declarations.
- Checking the semantic correctness of declarations, for instance, checking whether all *function block instances* in an *application* and its associated *subapplications* are properly allocated to *resources*, interconnected within resources, and intercommunicating among resources in a *system configuration.*
- Simulation and testing of the operation of an *instance* of a library element *type*, either by itself or in association with other instances of the same or different types.

4.7 Implementation of declarations

If required by the associated engineering task, a software tool shall provide facilities for the *implementation* of the *declarations* of its associated *library elements*. Such facilities may include, but are not limited to:

- the production of an executable code ("firmware") for embedding in instances of resource types and device types;
- the creation and interconnection ("downloading") of function block instances in resources and devices, for instance by using the management facilities defined in subclause 6.3 and Annexes F and G of IEC 61499-1.

4.8 System operation, testing and maintenance

If required by the associated engineering task, a software tool shall provide facilities for the operation, testing and maintenance of an Industrial Process Measurement and Control System (IPMCS) specified by its associated library elements. Such facilities may include, but are not limited to:

- the facilities described in preceding subclauses of this Clause;
- the information exchange facilities defined in IEC 61499-1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Annex A
(normative)

Document type definitions (DTDs)

A.1 General principles

This Annex presents Document Type Definitions (DTDs) for the exchange of IEC 61499 library elements between *software tools*. These DTDs are defined in the syntax defined in the eXtensible Markup Language (XML) specification at www.w3.org/TR/2000/REC-xml-20001006.

The correspondences between the DTD elements given in this annex, the library elements defined in IEC 61499-1, C.2.1, and the textual syntax given in IEC 61499-1, Annex B are given in Table A.1.

Table A.1 – Document type definition (DTD) elements

DTD element	LibraryElement	Textual syntax
`DataType`	DataTypeDeclaration	`data_type_declaration` (IEC 61131-3, B.1.3)
`FBType`	FBTypeDeclaration	`fb_type_declaration`
`SubapplicationType`	SubapplicationTypeDeclaration	`subapplication_type_declaration`
`AdapterType`	AdapterTypeDeclaration	`adapter_type_declaration`
`ResourceType`	ResourceTypeDeclaration	`resource_type_specification`
`DeviceType`	DeviceTypeDeclaration	`device_type_specification`
`System`	SystemConfiguration	`system_configuration`

The first table of each subclause of this Annex contains the DTD for the corresponding library element. The second table of each subclause provides a reference to the textual syntax (if any) plus an explanation for the major elements and attributes in the DTD. Following this, examples are given of the resulting XML files for typical library elements.

NOTE 1 If there is a conflict between the provisions of this Annex and the provisions of Annex B of IEC 61499-1, the provisions of the latter prevail.

NOTE 2 The examples given in this Annex provide a representative, but not exhaustive, sample of the features of the associated DTDs. In particular, these examples are not intended to be used as a test suite for compliance to the provisions of this standard.

A.2 DataType DTD

An XML document complying with the DTD in Table A.2 represents a **DataTypeDeclaration** object as described in C.2.2 of IEC 61499-1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.2 – DataType DTD (1 of 2)

```
<?xml version="1.0" encoding="UTF-8"?>

<!ELEMENT DataType (Identification?, VersionInfo+, CompilerInfo?,
ASN1Tag?, (DirectlyDerivedType | EnumeratedType | SubrangeType |
ArrayType | StructuredType))>

<!ATTLIST DataType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT Identification EMPTY>
<!ATTLIST Identification
 Standard CDATA #IMPLIED
 Classification CDATA #IMPLIED
 ApplicationDomain CDATA #IMPLIED
 Function CDATA #IMPLIED
 Type CDATA #IMPLIED
 Description CDATA #IMPLIED>
```

```
<!ELEMENT VersionInfo EMPTY>
<!ATTLIST VersionInfo
 Organization CDATA #REQUIRED
 Version CDATA #REQUIRED
 Author CDATA #REQUIRED
 Date CDATA #REQUIRED
 Remarks CDATA #IMPLIED>
```

```
<!ELEMENT ASN1Tag EMPTY>
<!ATTLIST ASN1Tag
 Class (UNIVERSAL | APPLICATION | CONTEXT | PRIVATE) #IMPLIED
 Number CDATA #REQUIRED>
```

```
<!ELEMENT CompilerInfo (Compiler*)>
<!ATTLIST CompilerInfo
 header CDATA #IMPLIED
 classdef CDATA #IMPLIED>
```

```
<!ELEMENT Compiler EMPTY>
<!ATTLIST Compiler
 Language (Java | Cpp | C | Other) #REQUIRED
 Vendor CDATA #REQUIRED
 Product CDATA #REQUIRED
 Version CDATA #REQUIRED>
```

```
<!ELEMENT DirectlyDerivedType EMPTY>
<!ATTLIST DirectlyDerivedType
 BaseType (BOOL | SINT | INT | DINT | LINT | USINT | UINT | UDINT
| ULINT | REAL | LREAL | TIME | DATE | TIME_OF_DAY | TOD
| DATE_AND_TIME | DT | STRING | BYTE |WORD | DWORD |LWORD |
WSTRING) #REQUIRED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT EnumeratedType (EnumeratedValue+)>
<!ATTLIST EnumeratedType
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT EnumeratedValue EMPTY>
<!ATTLIST EnumeratedValue
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.2 (2 of 2)

```
<!ELEMENT SubrangeType (Subrange)>
<!ATTLIST SubrangeType
 BaseType (SINT|INT|DINT|LINT|USINT|UINT|UDINT|ULINT) #REQUIRED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT Subrange EMPTY>
<!ATTLIST Subrange
 LowerLimit CDATA #REQUIRED
 UpperLimit CDATA #REQUIRED>
```

```
<!ELEMENT ArrayType (Subrange+)>
<!ATTLIST ArrayType
 BaseType CDATA #REQUIRED
 InitialValues CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT StructuredType (VarDeclaration|SubrangeVarDeclaration)+>
<!ATTLIST StructuredType
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT VarDeclaration EMPTY >
<!ATTLIST VarDeclaration
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 ArraySize CDATA #IMPLIED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT SubrangeVarDeclaration (Subrange+) >
<!ATTLIST SubrangeVarDeclaration
 Name CDATA #REQUIRED
 Type (SINT|INT|DINT|LINT|USINT|UINT|UDINT|ULINT) #REQUIRED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

Explanations of the elements of the above DTD, and (where applicable) references to the formal syntax for their attributes, are given in Table A.3.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.3 – DataType DTD elements (1 of 2)

Element attributes	Textual syntax (IEC 61131-3, Annex B)	Explanation
DataType		See IEC 61131-3
Name	data_type_name	
Comment	--	A comment per IEC 61131-3 without (* and *) delimiters
Identification		Information for data base retrieval
Standard		Primary reference standard in number-part-subclause format
Classification		Classification code as defined in reference standard
ApplicationDomain		Application domain as defined in reference standard
Function		Function of this element as defined in reference standard
Type		Element type (e.g., device type) as defined in reference standard
Description		Descriptive phrase as defined in reference standard
VersionInfo	--	Possibly one of several entries: First entry – Most recent version Second entry – Immediately preceding released version… Last entry – First released version
Organization	--	The organization supplying this library element
Version	digit [digit] '.' digit [digit] [letter]	The version identification for this library element
Author	--	The author of this library element
Date	date_literal ['-' daytime]	The release date of this version
Remarks	--	Comments relating to this version
ASN1Tag		ASN.1 tag per ISO/IEC 8824
Class		ASN.1 tag class per ISO/IEC 8824
Number		ASN.1 tag number per ISO/IEC 8824
CompilerInfo	--	Information for and about compilers used with this class
header	--	Header information such as package, imports, etc.
classdef	--	The class definition information such as superclass and implemented interfaces. If none is given, a default abstract superclass is used.
Compiler	--	Possibly one of several compilers used with this version
Language	--	The source language of this compiler
Vendor	--	The vendor of this compiler
Product	--	The product name of this compiler
Version	--	The version of this compiler
DirectlyDerivedType		See IEC 61131-3, Tables 12 and 14, item 1
BaseType	elementary_type_name	
InitialValue	constant	
Comment		A comment per IEC 61131-3 without (* and *) delimiters

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.3 (2 of 2)

Element Attributes	Textual Syntax (IEC 61131-3, Annex B)	Explanation
EnumeratedType	See IEC 61131-3 Tables 12 and 14, item 2	
InitialValue	enumerated_value	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
EnumeratedValue	See IEC 61131-3 Table 14, item 2	
Name	enumerated_value	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
SubrangeType	--	See IEC 61131-3 Tables 12 and 14, item 3
BaseType	integer_type_name	
InitialValue	signed_integer	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
Subrange	See IEC 61131-3 Tables 12 and 14, item 3	
LowerLimit	signed_integer	
UpperLimit	signed_integer	
ArrayType	See IEC 61131-3 Tables 12 and 14, item 4	
BaseType	non_generic_type_name	
InitialValues	array_initialization	
StructuredType	See IEC 61131-3 Tables 12, item 5 and 14, item 5 and item 6	
VarDeclaration		
Name	structure_element_name	
Type	non_generic_type_name	
ArraySize	a	
InitialValue	b	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
SubrangeVarDeclaration	See IEC 61131-3, 2.3.3.	
Name	structure_element_name	
Type	integer_type_name	
InitialValue	signed_integer	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	

a The syntax of this attribute when present shall be equivalent to the syntactic expression (subrange {','subrange}) | integer {','integer} where the non-terminals subrange and integer are as defined in Annex B of IEC 61131-3. Each term of the second form is equivalent to the subrange 0..n-1, where n is the value of the corresponding integer syntactic element. If this attribute is missing, the structure component is not an anonymously defined array.

b The syntax of this attribute is the syntax for initialization of the corresponding variable type as defined in Annex B.1.4.3 of IEC 61131-3.

EXAMPLE The structured data type ANALOG_CHANNEL_CONFIGURATIONI example is expressed in IEC 61131-3, Table 14 as follows:

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

```
TYPE ANALOG_CHANNEL_CONFIGURATIONI:
    STRUCT
      RANGE: ANALOG_SIGNAL_RANGE;
      MIN_SCALE: ANALOG_DATA:= -4095;
      MAX_SCALE: ANALOG_DATA:=  4095;
    END_STRUCT;
END_TYPE
```

A corresponding XML document could be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM "DataType.dtd"
>
<DataType
   Name="ANALOG_CHANNEL_CONFIGURATIONI"
   Comment="IEC 61131-3, Table 14#5">
<Identification
   Function="Configuration Data"
   Standard="61131-3-2.3.3.2"
   ApplicationDomain="Any"
   Classification="Data type"
   Type="Analog"
   Description="Table 14, #5"/>
<VersionInfo
   Organization="IEC SC65B/WG7/TF3"
   Version="2.0"
   Author="JHC"
   Date="2000-01-31"/>
<StructuredType>
  <VarDeclaration Name="SIGNAL_RANGE"
     Type="ANALOG_SIGNAL_RANGE"/>
  <VarDeclaration Name="MIN_SCALE"
     Type="ANALOG_DATA"
     InitialValue="-4095"/>
  <VarDeclaration Name="MAX_SCALE"
     Type="ANALOG_DATA"
     InitialValue="4095"/>
</StructuredType>
</DataType>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

A.3 LibraryElement DTD

An XML document complying with the DTD in Table A.4 represents a **LibraryElement** object as described in Annex C of IEC 61499-1. Possible root elements of such a document are `FBType`, `AdapterType`, `ResourceType`, `DeviceType`, `System`, and `SubappType`, representing the concrete subclasses **FBTypeDeclaration**, **AdapterTypeDeclaration**, **ResourceTypeDeclaration**, **DeviceTypeDeclaration**, **SystemConfiguration**, and **SubapplicationTypeDeclaration** of the abstract superclass **LibraryElement**, respectively. The **DataTypeDeclaration** subclass is represented separately by the DTD given in Clause A.2.

Table A.4 – Library Element DTD (1 of 5)

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Common elements -->
<!ELEMENT Identification EMPTY>
<!ATTLIST Identification
 Standard CDATA #IMPLIED
 Classification CDATA #IMPLIED
 ApplicationDomain CDATA #IMPLIED
 Function CDATA #IMPLIED
 Type CDATA #IMPLIED
 Description CDATA #IMPLIED>
```

```
<!ELEMENT VersionInfo EMPTY>
<!ATTLIST VersionInfo
 Organization CDATA #REQUIRED
 Version CDATA #REQUIRED
 Author CDATA #REQUIRED
 Date CDATA #REQUIRED
 Remarks CDATA #IMPLIED>
```

```
<!ELEMENT CompilerInfo (Compiler*)>
<!ATTLIST CompilerInfo
 header CDATA #IMPLIED
 classdef CDATA #IMPLIED>

<!ELEMENT Compiler EMPTY>
<!ATTLIST Compiler
 Language (Java | Cpp | C | Other) #REQUIRED
 Vendor CDATA #REQUIRED
 Product CDATA #REQUIRED
 Version CDATA #REQUIRED>
```

```
<!ELEMENT FBNetwork
(FB*,EventConnections?,DataConnections?,AdapterConnections?)>

<!ELEMENT FB (Parameter*)>
<!ATTLIST FB
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
  x CDATA #IMPLIED
  y CDATA #IMPLIED>
```

```
<!ELEMENT EventConnections (Connection+)>

<!ELEMENT DataConnections (Connection+)>
<!ELEMENT AdapterConnections (Connection+)>
<!ELEMENT Connection EMPTY>
<!ATTLIST Connection
 Source CDATA #REQUIRED
 Destination CDATA #REQUIRED
 Comment CDATA #IMPLIED
 dx1 CDATA #IMPLIED
 dx2 CDATA #IMPLIED
 dy CDATA #IMPLIED
>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.4 (2 of 5)

```
<!-- FBType elements -->
<!ELEMENT FBType
(Identification?,VersionInfo+,CompilerInfo?,InterfaceList,
 (BasicFB | FBNetwork)?, Service?) >
<!ATTLIST FBType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED
>
```

```
<!ELEMENT InterfaceList
(EventInputs?,EventOutputs?,InputVars?,OutputVars?,
 Sockets?, Plugs?)>
<!ELEMENT EventInputs (Event+)>
<!ELEMENT EventOutputs (Event+)>
<!ELEMENT InputVars (VarDeclaration+)>
<!ELEMENT OutputVars (VarDeclaration+)>
<!ELEMENT Sockets (AdapterDeclaration+)>
<!ELEMENT Plugs (AdapterDeclaration+)>
```

```
<!ELEMENT Event (With*)>
<!ATTLIST Event
 Name CDATA #REQUIRED
 Type CDATA #IMPLIED
 Comment CDATA #IMPLIED>

<!ELEMENT With EMPTY>
<!ATTLIST With
 Var CDATA #REQUIRED>
```

```
<!ELEMENT VarDeclaration EMPTY>
<!ATTLIST VarDeclaration
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 ArraySize CDATA #IMPLIED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT AdapterDeclaration (Parameter*)>
<!ATTLIST AdapterDeclaration
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

```
<!ELEMENT BasicFB (InternalVars?,ECC?,Algorithm*)>

<!ELEMENT InternalVars (VarDeclaration+)>

<!ELEMENT ECC (ECState+,ECTransition+)>
```

```
<!ELEMENT ECState (ECAction*)>
<!ATTLIST ECState
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

```
<!ELEMENT ECTransition EMPTY>
<!ATTLIST ECTransition
 Source CDATA #REQUIRED
 Destination CDATA #REQUIRED
 Condition CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.4 (3 of 5)

```
<!ELEMENT ECAction EMPTY>
<!ATTLIST ECAction
 Algorithm CDATA #IMPLIED
 Output CDATA #IMPLIED>
```

```
<!ELEMENT Algorithm (VarDeclaration*, (FBD | ST | LD | Other))>
<!ATTLIST Algorithm
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT FBD (FB+,DataConnections)>
```

```
<!ELEMENT ST (#PCDATA)>
<!ATTLIST ST
 Text CDATA #IMPLIED >
```

```
<!ELEMENT LD (Rung+)>
<!ELEMENT Rung (#PCDATA)>
<!ATTLIST Rung
 Output CDATA #REQUIRED
 Expression CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT Other  (#PCDATA)>
<!ATTLIST Other
 Language CDATA #REQUIRED >
```

```
<!ELEMENT Service (ServiceSequence+)>
<!ATTLIST Service
 RightInterface CDATA #REQUIRED
 LeftInterface CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT ServiceSequence (ServiceTransaction*)>
<!ATTLIST ServiceSequence
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT ServiceTransaction (InputPrimitive?, OutputPrimitive*)>

<!ELEMENT InputPrimitive EMPTY>
<!ATTLIST InputPrimitive
 Interface CDATA #REQUIRED
 Event CDATA #REQUIRED
 Parameters CDATA #IMPLIED>

<!ELEMENT OutputPrimitive EMPTY>
<!ATTLIST OutputPrimitive
 Interface CDATA #REQUIRED
 Event CDATA #REQUIRED
 Parameters CDATA #IMPLIED>
```

```
<!-- AdapterType elements -->
<!ELEMENT AdapterType
(Identification?,VersionInfo+,CompilerInfo?,InterfaceList,Service?)>
<!ATTLIST AdapterType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.4 (4 of 5)

```
<!-- ResourceType elements -->
<!ELEMENT ResourceType (Identification?,VersionInfo+, CompilerInfo?,
FBTypeName*, VarDeclaration*, FBNetwork?)>
<!ATTLIST ResourceType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT FBTypeName EMPTY>
<!ATTLIST FBTypeName
 Name CDATA #REQUIRED>
```

```
<!-- DeviceType elements -->
<!ELEMENT DeviceType (Identification?, VersionInfo+, CompilerInfo?,
VarDeclaration*, ResourceTypeName*, Resource*, FBNetwork?)>
<!ATTLIST DeviceType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT ResourceTypeName EMPTY>
<!ATTLIST ResourceTypeName
 Name CDATA #REQUIRED >

<!ELEMENT Resource (Parameter*,FBNetwork?)>
<!ATTLIST Resource
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

```
<!-- System elements -->
<!ELEMENT System (Identification?, VersionInfo+, CompilerInfo?,
Application*, Device+,  Mapping*, Segment*, Link*)>
<!ATTLIST System
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT Application (SubAppNetwork)>
<!ATTLIST Application
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT Mapping EMPTY>
<!ATTLIST Mapping
 From CDATA #REQUIRED
 To CDATA #REQUIRED>

<!ELEMENT Device (Parameter*,Resource*,FBNetwork?)>
<!ATTLIST Device
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.4 (5 of 5)

```
<!-- SubAppType elements -->
<!ELEMENT SubAppType
(Identification?,VersionInfo+,CompilerInfo?,SubAppInterfaceList,
SubAppNetwork?)>
<!ATTLIST SubAppType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT SubAppInterfaceList
(SubAppEventInputs?,SubAppEventOutputs?,InputVars?,OutputVars?)>
<!ELEMENT SubAppEventInputs (SubAppEvent+)>
<!ELEMENT SubAppEventOutputs (SubAppEvent+)>

<!ELEMENT SubAppEvent EMPTY>
<!ATTLIST SubAppEvent
 Name CDATA #REQUIRED
 Type CDATA #IMPLIED
 Comment CDATA #IMPLIED>

<!ELEMENT SubAppNetwork
(SubApp*,FB*,EventConnections?,DataConnections?, AdapterConnections?)>

<!ELEMENT SubApp EMPTY>
<!ATTLIST SubApp
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
  x CDATA #IMPLIED
  y CDATA #IMPLIED>
```

```
<!-- Network elements -->
<!ELEMENT SegmentType (Identification?,VersionInfo+, CompilerInfo?,
VarDeclaration*)>
<!ATTLIST SegmentType
  Name CDATA #REQUIRED
  Comment CDATA #IMPLIED >

<!ELEMENT Segment (Parameter*)>
<!ATTLIST Segment
  Name CDATA #REQUIRED
  Type CDATA #REQUIRED
  Comment CDATA #IMPLIED
  x CDATA #IMPLIED
  y CDATA #IMPLIED
  dx1 CDATA #IMPLIED>

<!ELEMENT Parameter EMPTY>
<!ATTLIST Parameter
  Name CDATA #REQUIRED
  Value CDATA #REQUIRED
  Comment CDATA #IMPLIED>

<!ELEMENT Link (Parameter*)>
<!ATTLIST Link
  SegmentName CDATA #REQUIRED
  CommResource CDATA #REQUIRED
  Comment CDATA #IMPLIED >
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Explanations of some of the elements of the above DTD, and (where applicable) references to the formal syntax for their attributes, are given in Table A.5.

Table A.5 – LibraryElement DTD elements (1 of 5)

Element attributes	Syntax (IEC 61499-1, Annex B)	Explanation
Identification	See Table A.3	
VersionInfo		
CompilerInfo		
Compiler		
FBNetwork	A *function block network* as defined in IEC 61499-1.	
FB	A *function block instance* as defined in IEC 61499-1.	
Name	fb_instance_name	
Type	fb_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
x, y	See Annex B.	
Connection	An *event connection, data connection* or *adapter connection* as defined in IEC 61499-1.	
Source	c) [1]	
Destination	c)	
dx1, dx2, dy	See Annex B.	
FBType	A *function block type* as described in IEC 61499-1.	
Name	fb_type_name	
Comment	A comment per IEC 61131 without (* and *) delimiters	
Event	A *declaration* of an *event interface.*	
Name	event_input_name | event_output_name	d)
Type	event_type	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
With	A *declaration* of an association between an *event* and a *variable.*	
Var	input_variable_name | output_ variable _name	d)
VarDeclaration	A declaration of a variable.	
Name	input_variable_name | output_variable_name | internal_variable_name	e)
Type	identifier	
ArraySize	f)	
InitialValue	g)	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	

[1] See Table footnote at the end of Table A.1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.5 (2 of 5)

Element Attributes	Syntax (IEC 61499-1, Annex B)	Explanation
AdapterDeclaration	A declaration of a plug or socket interface of a function block type.	
Name	plug_name | socket_name	h) [1]
Type	adapter_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
x, y	Location (See Annex B) of plug or socket in the internal function block network of a *composite function block type*.	
ECState	An *EC state* as defined in IEC 61499-1.	
Name	ec_state_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
x, y	See Annex B.	
ECTransition	An *EC transition* as defined in IEC 61499-1.	
Source	ec_state_name	
Destination	ec_state_name	
Condition	ec_transition_condition	
x, y	See Annex B.	
ECAction	An *EC action* as defined in IEC 61499-1.	
Algorithm	algorithm_name	
Output	event_output_name	
Algorithm	An *algorithm* in a specified language a)	
Name	algorithm_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
ST	An *algorithm* in the IEC 61131-3 ST language.	
Text	Algorithm contents in the syntax of a statement_list per IEC 61131-3, Annex B.	l)
Rung	A rung of an algorithm in the LD language	
Output	b)	
Expression	See b) and l).	
Other	An algorithm in a language other than FBD, ST or LD.	
Language	The name of the programming language	a)
Text	Content of the algorithm in the syntax defined for the particular language	See a) and l).
Service	A *declaration* of a *service* per IEC 61499-1	
RightInterface	service_interface_name	
LeftInterface	service_interface_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	

[1] See Table footnote at the end of Table A.1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.5 (3 of 5)

Element Attributes	Syntax (IEC 61499-1, Annex B)	Explanation
ServiceSequence	A declaration of a service sequence per IEC 61499-1	
Name	sequence_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
InputPrimitive	An input *service primitive* per IEC 61499-1	
Interface	service_interface_name	
Event	(([plug_name '.'] event_input_name) | (socket_name '.' event_output_name)) ['+' | '-']	
Parameters	input_variable_name {',' input_variable_name}	
OutputPrimitive	An output *service primitive* per IEC 61499-1	
Interface	service_interface_name	
Event	('NULL' | ([plug_name '.'] event_output_name) | (socket_name '.' event_input_name)) ['+' | '-']	
Parameters	output_variable_name {',' output_variable_name}	
AdapterType	A declaration of an adapter interface type per IEC 61499-1	
Name	adapter_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
ResourceType	A *declaration* of a *resource type* per IEC 61499-1	
Name	resource_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
FBTypeName	The name of a function block type supported by all instances of a resource type	
Name	fb_type_name	
DeviceType	A *declaration* of a *device type* per IEC 61499-1	
Name	device_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
ResourceTypeName	The name of a *resource type* supported by all *instances* of a *device type*	
Name	resource_type_name	
Resource	A resource instance present in all instances of a device type	
Name	resource_instance_name	
Type	resource_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.5 (4 of 5)

Element Attributes	Syntax (IEC 61499-1, Annex B)	Explanation
System	A declaration of a system configuration per IEC 61499-1	
Name	system_name	
Comment	A comment per IEC 61131-3, 2.1.5 without (* and *) delimiters	
Application	A *declaration* of an *application* per IEC 61499-1	
Name	application_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
Mapping	Mapping of a *function block instance* from an *application* onto a function block instance in a *resource*.	
From	fb_instance_reference	Hierarchical *function block instance name* in its *application*, e.g., APP1.SUBAPP2.FB2
To	fb_resource_reference	Hierarchical *function block instance name* in the physical system j) [1].
Device	A declaration of a device configuration per IEC 61499-1	
Name	device_instance_name	
Type	device_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
SubAppType	A declaration of a subapplication type per IEC 61499-1	
Name	subapp_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
SubAppEvent	A declaration of an event interface of a subapplication type.	
Name	event_input_name | event_output_name	i)
Type	event_type	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
SubApp	A *subapplication instance* as defined in IEC 61499-1.	
Name	subapp_instance_name	
Type	subapp_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
x, y	See Annex B.	
SegmentType	A *declaration* of a *segment type* per IEC 61499-1	
Name	segment_type_name	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	

[1] See Table footnote at the end of Table A.1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Table A.5 (5 of 5)

Element Attributes	Syntax (IEC 61499-1, Annex B)	Explanation
Segment	A segment of a communication *network*	
Name	identifier	
Type	identifier	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
x, y, dx1	See Annex B.	
Link	A link between a Segment element and a Device element	
CommResource	resource_hierarchy	k)
SegmentName	identifier	The segment to be linked
Comment	A comment per IEC 61131-3 without (* and *) delimiters	
Parameter	A *parameter* of an element, e.g. a Segment or Link element.	
Name	identifier	
Value	A character string in an appropriate format to express the value of the associated parameter.	
Comment	A comment per IEC 61131-3 without (* and *) delimiters	

a) The specification of algorithms in languages other than FBD, ST and LD is beyond the scope of this standard.

b) Since the FBD and ST languages are available for the specification of complex algorithms, it is recommended that the usage of the LD language in the context of this standard be limited to rungs performing the evaluation of assignment statements of the form <output>:= <expression>. For portability between software tools, it is further recommended that the XML Expression element have the following simple postfix-operator textual syntax with whitespace_separated terms:

```
expression::= and_expression
and_expression::= (variable_name ['!']) | or_expression
      | and_expression and_expression '&'
or_expression::= and_expression | or_expression or_expression '|'
```

See EXAMPLE 1 in Clause C.1 for an illustration of this recommended usage.

c) Depending on the context, the syntax of a Source or Destination element should correspond to the syntax of the respective element in one of the productions event_conn, data_conn, adapter_conn, subapp_event_conn, subapp_data_conn, config_event_conn, config_data_conn, config_adapter_conn, devtype_event_conn, devtype_data_conn, or devtype_adapter_conn given in Annex B of IEC 61499-1.

d) The productions event_input_name and input_variable_name apply when the Event element is part of an EventInputs element, and event_output_name and output_variable_name apply when it is part of an EventOutputs element.

e) The productions input_variable_name, output_variable_name and internal_variable_name apply when the associated VarDeclaration element is part of an InputVars, OutputVars or InternalVars element, respectively.

f) The syntax of this element when present shall be equivalent to the syntactic expression (subrange {',' subrange}) | integer {',' integer} where the non-terminals subrange and integer are as defined in Annex B of IEC 61131-3. Each term of the second form is equivalent to the subrange 0..n-1, where n is the value of the corresponding integer syntactic element. If this element is missing, the variable is not an array of anonymous type, although it could still be an instance of a previously defined array type.

g) The syntax of this element is the syntax for initialization of the corresponding variable type as defined in Annex B of IEC 61131-3.

h) The productions plug_name and socket_name apply when the associated

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

`AdapterDeclaration` element is part of a `Plugs` or `Sockets` element, respectively.

i) The productions `event_input_name` and `event_output_name` apply when the `SubAppEvent` element is part of a `SubAppEventInputs` or `SubAppEventOutputs` element, respectively.

j) This element may show a full device/resource/FB name hierarchy, e.g. `DEV1.RES2.FB2`; a device/resource hierarchy, e.g., `DEV1.RES2`; or (in the case where the device itself is a single resource) simply a device name, e.g., `DEV1`. In the latter two cases, the FB instance shall have the same name as in the application, e.g., if the source is `APP1.FB3` then the resulting mapping shall be to `DEV1.RES2.FB3` and `DEV1.FB3` respectively.

k) This attribute references the communication resource linked to the network segment. It may show a full device/resource name sequence, e.g. `DEV1.RES2`; or (in the case where the device itself provides the communication interface) simply a device name, e.g., `DEV1`.

l) To increase readability and reduce the probability of errors in encoding and parsing data, the use of CDATA sections instead of textual attributes is recommended for these elements.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Annex B
(informative)

Graphics model

B.1 Coordinate system

This Annex presents a simple graphics model which permits the approximate reconstruction of the graphical appearance of *communication networks, function block networks* and *execution control charts (ECCs)* between software tools, utilizing the data defined for the **FBNetwork, ECC, ResourceType, DeviceType, SubAppType** and **System** elements in the **LibraryElement** DTD in Annex A.

NOTE The graphical model in this Annex is intended to allow passing of information among software tools with a common semantic, but exact reproduction of graphics may not be achieved among tools with different layout and drawing algorithms.

Since the main direction of event and data flow in function block networks is from left to right, and secondarily from top to bottom, the coordinate system of the graphics model has its origin (x=y=0) at the upper left hand corner with x-coordinate values increasing in the left-to-right direction and y-coordinate values increasing in the top-to-bottom direction, as illustrated in Figure B.1.

To achieve independence from font sizes and display resolutions, both x and y coordinates are expressed as multiples of 1 % of the line height (shown as **h** in Figure B.1) used for the layout of function block inputs and outputs.

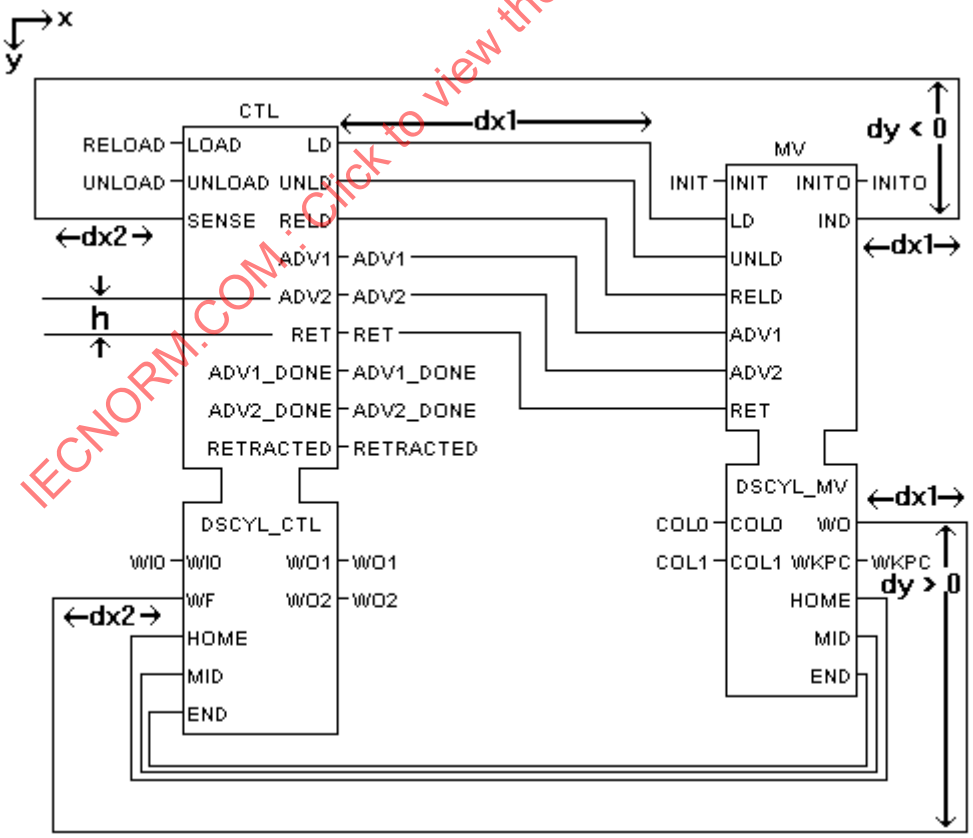


Figure B.1 – Graphics model

EXAMPLE The upper left-hand corner of the DSCYL_CTL instance named CTL in Figure B.1 is located approximately 10h units from the left-hand edge of the diagram and 5h units from the top of the diagram; hence the

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

values of the x and y attributes of the corresponding FB sub-element in an FBNetwork element of an XML document defined according to one of the DTDs listed above would be 1 000 and 500, respectively.

B.2 Location of graphical elements

The location of a *function block instance* is determined by the location of the upper left corner of its graphical outline.

The location of an *EC state* is determined by the center point of the bounding box containing the state name.

The location of an *EC transition condition* is determined by the center point of the invisible bounding box containing the transition condition.

NOTE `(x,y)` coordinates may be used in the transfer syntax of *device* and *resource instances*. However, software tools that use default graphical or tree notations for these elements are not required to use or produce these attributes.

B.3 Routing of connections

As illustrated in Figure B.1, *data connections, event connections* and *adapter connections* may be drawn as an odd number of line segments according to the following guidelines.

a) When the source of the connection is to the left of its destination, the line may be drawn as a single straight line proceeding from the right edge of the function block which provides the source of the connection to the left edge of the function block which provides the destination of the connection.

b) When the source of the connection is to the left of its destination, the connection can be drawn as three contiguous line segments proceeding rightward at a distance **dx1** from the right edge of the function block which provides the source of the connection; thence, vertically, an appropriate distance to proceed horizontally to the left edge of the function block which provides the destination of the connection.

c) When the source of the connection is to the left or right of its destination, the connection can be drawn as five contiguous line segments proceeding rightward at a distance **dx1** from the right edge of the function block which provides the source of the connection; thence vertically at a distance **dy**; thence horizontally to an x-coordinate at a distance **dx2** left of the left edge of the function block which provides the destination of the connection; thence vertically at an appropriate distance to proceed horizontally to the left edge of the function block which provides the destination of the connection.

d) *EC transitions* are drawn as two straight lines from the location of the source *EC action* to the location of the EC transition condition and thence to the location of the destination EC action, where these locations are as defined in Clause B.2. The portions of these lines within the bounding boxes of EC state names and transition conditions are hidden, as illustrated in Figure B.2. Arrowheads or other graphic means may be used to indicate the direction of transitions.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

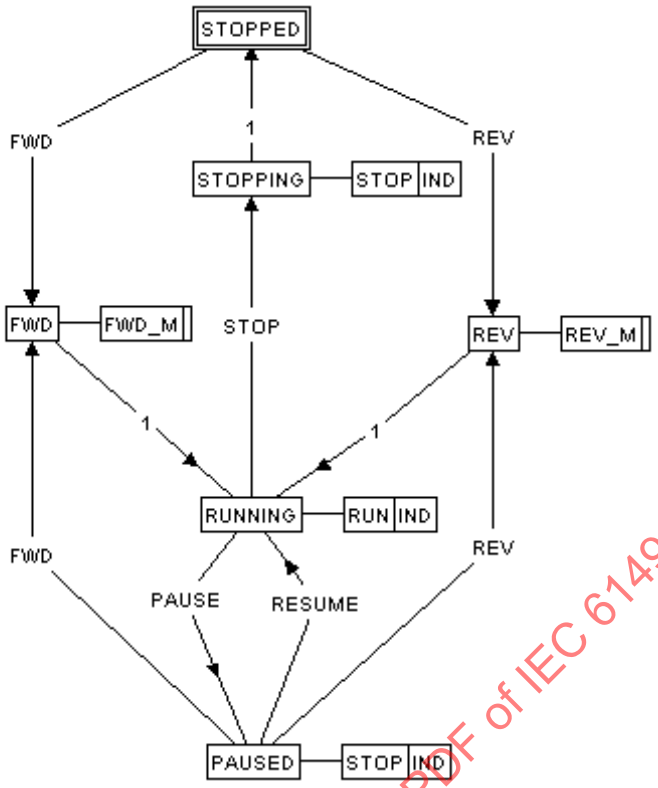


Figure B.2 – ECC drawing example

B.4 Default layouts

Suppliers of software tools shall specify the means employed to obtain default layouts of *ECCs* and *function block networks* when the necessary graphic information is not supplied in the XML transfer document of the associated *library element*, or when the information supplied in the document is inconsistent with the drawing algorithms employed by the particular tool.

NOTE An example of such inconsistency is when a software tool uses a different algorithm to determine the width and height of graphical elements from the algorithm employed by the tool producing the document, which may cause graphical interferences among elements such as overlapping of connecting lines with element outlines.

B.5 Graphical representation of system configurations

A *system configuration* as defined in IEC 61499-1, whose XML representation is given in the `System` element notation defined in Tables A.4 and A.5, can be laid out in the coordinate system defined in Clause B.1, according to the following rules.

a) A *device* is represented as a generally rectangular block containing the device's type name, with its instance name at the top of the block. The (x,y) coordinates of the center of the block are given by the `x` and `y` attributes of the XML `Device` element, respectively.

b) A *network segment* is represented as a horizontal line segment, which may be thick enough vertically to contain textual information such as the segment instance name and type name and may have other implementation-dependent features such as arrowheads. The origin and length of the line segment, and the vertical location of the center of the line, are given by the `x`, `dx1` and `y` parameters of the XML `Segment` element, respectively.

c) A *network link* is represented as a vertical line segment from the center of a device (or other implementation-dependent position) to the horizontal center line of the corresponding network link. Portions of the link may be overlaid by the graphical representations of segments and devices.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Annex C
(informative)

Examples

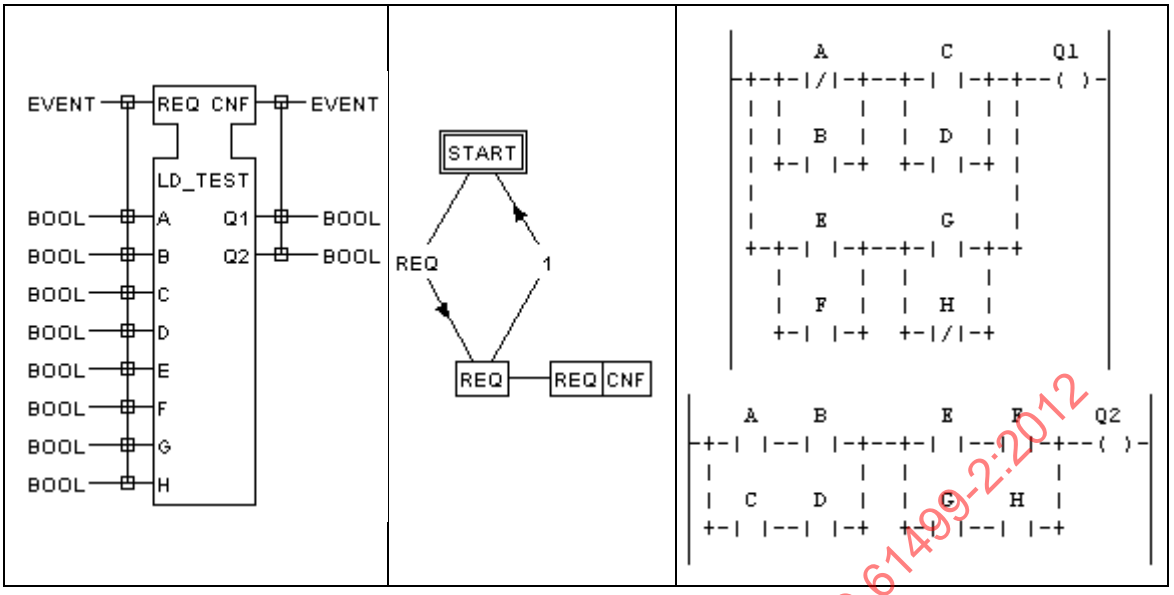
C.1 Basic function block types

EXAMPLE 1 A basic function block type containing a Ladder Diagram (LD) algorithm according to NOTE 2 of Table A.5 could be expressed textually as follows.

```
FUNCTION_BLOCK LD_TEST (* LD Algorithm Example *)
EVENT_INPUT
    REQ WITH A, B, C, D, E, F, G, H;
END_EVENT
EVENT_OUTPUT
    CNF WITH Q1, Q2; (* Execution Confirmation *)
END_EVENT
VAR_INPUT
    A: BOOL;
    B: BOOL;
    C: BOOL;
    D: BOOL;
    E: BOOL;
    F: BOOL;
    G: BOOL;
    H: BOOL;
END_VAR
VAR_OUTPUT
    Q1: BOOL;
    Q2: BOOL;
END_VAR
EC_STATES
    START; (* Initial State *)
    REQ: REQ -> CNF; (* Normal execution *)
END_STATES
EC_TRANSITIONS
    START TO REQ:= REQ;
    REQ TO START:= 1;
END_TRANSITIONS
ALGORITHM REQ IN LD:
    Q1:= ((!A|B)&(C|D))|((E|F)&(G|!H));
    Q2:= ((A&B)|(C&D))&((E&F)|(G&H));
END_ALGORITHM
END_FUNCTION_BLOCK
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

The interface, ECC, and REQ algorithm could appear graphically as follows.



IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

A corresponding XML document would be:

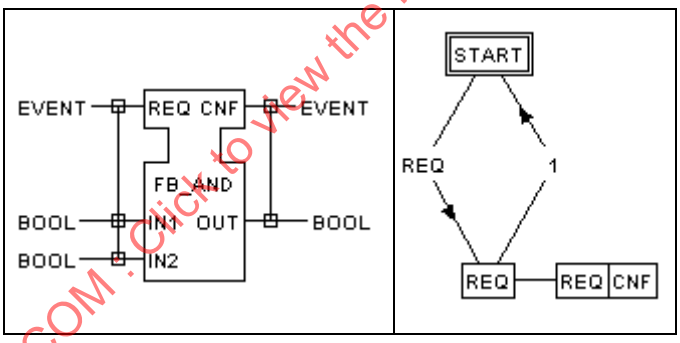
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "../LibraryElement.dtd" >
<FBType Name="LD_TEST" Comment="LD Algorithm Example" >
  <Identification Standard="61499-2-C.1" Description="LD Algorithm Example" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.2" Author="JHC" Date="2000-11-
16" Remarks="Corrected Identification" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC" Date="2000-06-
20" Remarks="Tested Sun compiler" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC" Date="2000-02-
01" />
  <CompilerInfo header="package fb.rt.part2;" >
    <Compiler Language="Java" Vendor="IBM" Product="VisualAge" Version="3.0" />
    <Compiler Language="Java" Vendor="Sun" Product="JDK" Version="1.1.8" />
  </CompilerInfo>
  <InterfaceList>
    <EventInputs>
      <Event Name="REQ" >
        <With Var="A" />
        <With Var="B" />
        <With Var="C" />
        <With Var="D" />
        <With Var="E" />
        <With Var="F" />
        <With Var="G" />
        <With Var="H" />
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Name="CNF" Comment="Execution Confirmation" >
        <With Var="Q1" />
        <With Var="Q2" />
      </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Name="A" Type="BOOL" />
      <VarDeclaration Name="B" Type="BOOL" />
      <VarDeclaration Name="C" Type="BOOL" />
      <VarDeclaration Name="D" Type="BOOL" />
      <VarDeclaration Name="E" Type="BOOL" />
      <VarDeclaration Name="F" Type="BOOL" />
      <VarDeclaration Name="G" Type="BOOL" />
      <VarDeclaration Name="H" Type="BOOL" />
    </InputVars>
    <OutputVars>
      <VarDeclaration Name="Q1" Type="BOOL" />
      <VarDeclaration Name="Q2" Type="BOOL" />
    </OutputVars>
  </InterfaceList>
  <BasicFB>
    <ECC >
      <ECState Name="START" Comment="Initial State" x="341.1765" y="105.8824" >
      </ECState>
      <ECState Name="REQ" Comment="Normal execution" x="358.8235" y="858.8235" >
        <ECAction Algorithm="REQ" Output="CNF" />
      </ECState>
      <ECTransition Source="START" Destination="REQ" Condition="REQ" x="170.5882"
y="494.1176" />
      <ECTransition Source="REQ" Destination="START" Condition="1" x="564.7059"
y="500" />
    </ECC>
  <Algorithm Name="REQ" Comment="Normally executed algorithm" >
    <LD >
      <Rung Output="Q1" Expression="A ! B | C D | &#38; E F | G H ! | &#38; | " />
      <Rung Output="Q2" Expression="A B &#38; C D &#38; | E F &#38; G H &#38; |
&#38; " />
    </LD>
  </Algorithm>
  </BasicFB>
</FBType>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

EXAMPLE 2 A basic function block type containing a ST algorithm could be expressed textually as follows.

```
FUNCTION_BLOCK FB_AND (* Boolean AND *)
EVENT_INPUT
    REQ WITH IN1, IN2;
END_EVENT
EVENT_OUTPUT
    CNF WITH OUT;
END_EVENT
VAR_INPUT
    IN1: BOOL;
    IN2: BOOL;
END_VAR
VAR_OUTPUT
    OUT: BOOL; (* IN1&IN2 *)
END_VAR
EC_STATES
    START; (* Initial State *)
    REQ: REQ -> CNF; (* Normal execution *)
END_STATES
EC_TRANSITIONS
    START TO REQ:= REQ;
    REQ TO START:= 1;
END_TRANSITIONS
ALGORITHM REQ IN ST:
  OUT:= (IN1 & IN2);
END_ALGORITHM
END_FUNCTION_BLOCK
```

The interface and ECC would appear graphically as follows.



IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

A corresponding XML document would be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "../LibraryElement.dtd" >
<FBType Name="FB_AND" Comment="Boolean AND" >
  <Identification Standard="61499-1-D.1" Classification="Math"
ApplicationDomain="Any" Function="AND" Type="Boolean" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC"
Date="2000-06-10" Remarks="Tested Sun compiler." />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC"
Date="2000-01-29" Remarks="Simple Boolean AND" />
  <CompilerInfo header="package fb.rt.part2;" >
    <Compiler Language="Java" Vendor="Sun" Product="JDK" Version="1.1.8"
/>
    <Compiler Language="Java" Vendor="IBM" Product="VisualAge"
Version="3.0" />
  </CompilerInfo>
  <InterfaceList>
    <EventInputs>
      <Event Name="REQ" >
        <With Var="IN1" />
        <With Var="IN2" />
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Name="CNF" >
        <With Var="OUT" />
      </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Name="IN1" Type="BOOL" />
      <VarDeclaration Name="IN2" Type="BOOL" />
    </InputVars>
    <OutputVars>
      <VarDeclaration Name="OUT" Type="BOOL" Comment="IN1&#38;IN2" />
    </OutputVars>
  </InterfaceList>
  <BasicFB>
    <ECC >
      <ECState Name="START" Comment="Initial State" x="200" y="105.8824" >
      </ECState>
      <ECState Name="REQ" Comment="Normal execution" x="205.8824"
y="676.4706" >
        <ECAction Algorithm="REQ" Output="CNF" />
      </ECState>
      <ECTransition Source="START" Destination="REQ" Condition="REQ"
x="370.5882" y="405.8824" />
      <ECTransition Source="REQ" Destination="START" Condition="1"
x="52.9412" y="429.4117" />
    </ECC>
  <Algorithm Name="REQ" >
    <ST Text="  OUT:= (IN1 &#38; IN2);&#10;" />
  </Algorithm>
  </BasicFB>
</FBType>
```

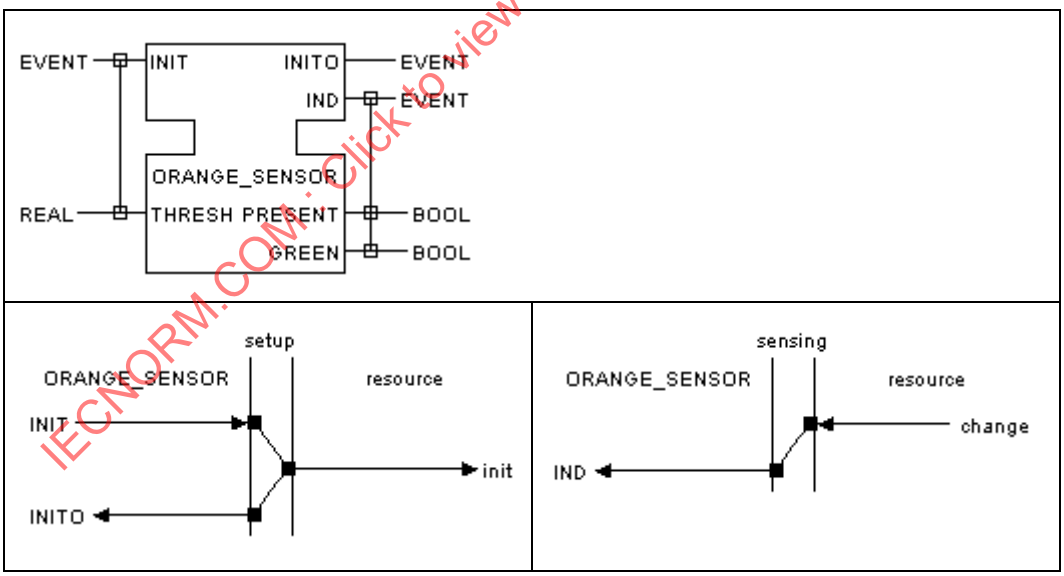
IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

C.2 Service interface function block types

EXAMPLE 1 A service interface function block type for sensing the presence and condition of an orange on a conveyor could be expressed textually as follows.

```
FUNCTION_BLOCK ORANGE_SENSOR (* Sense Presence & Color of Orange *)
EVENT_INPUT
    INIT WITH THRESH; (* Set Threshold *)
END_EVENT
EVENT_OUTPUT
    INITO; (* Threshold Set *)
    IND WITH PRESENT, GREEN; (* Change in Presence or Color *)
END_EVENT
VAR_INPUT
    THRESH: REAL; (* Adjustable Color Threshold *)
END_VAR
VAR_OUTPUT
    PRESENT: BOOL; (* Orange is Present *)
    GREEN: BOOL; (* Green is Above Threshold *)
END_VAR
SERVICE ORANGE_SENSOR/resource
SEQUENCE setup
    ORANGE_SENSOR.INIT(THRESH) -> resource.init() ->
ORANGE_SENSOR.INITO();
END_SEQUENCE
SEQUENCE sensing
    resource.change() -> ORANGE_SENSOR.IND(PRESENT,GREEN);
END_SEQUENCE
END_SERVICE
END_FUNCTION_BLOCK
```

The interface and service sequences would appear graphically as follows:



IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

A corresponding XML document would be:

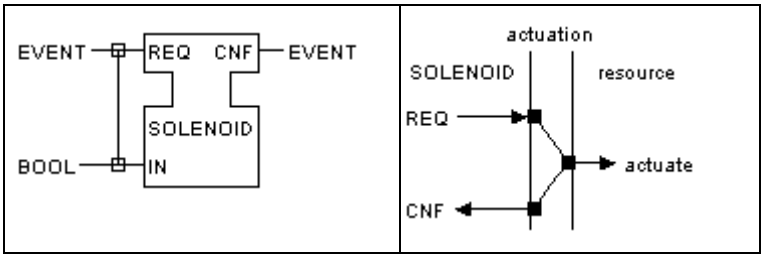
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "../LibraryElement.dtd" >
<FBType Name="ORANGE_SENSOR" Comment="Sense Presence &#38; Color of Orange" >
  <Identification Classification="C0202" ApplicationDomain="Food Processing"
Function="Detection" Type="Photoelectric Sensors" Description="Orange Presence and
Quality" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC" Date="2000-05-
14" Remarks="Modified to use LibraryElement.dtd" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC" Date="2000-01-
26" />
  <CompilerInfo header="package fb.rt.part2;" >
  </CompilerInfo>
  <InterfaceList>
    <EventInputs>
      <Event Name="INIT" Comment="Set Threshold" >
        <With Var="THRESH" />
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Name="INITO" Comment="Threshold Set" >
      </Event>
      <Event Name="IND" Comment="Change in Presence or Color" >
        <With Var="PRESENT" />
        <With Var="GREEN" />
      </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Name="THRESH" Type="REAL" Comment="Adjustable Color Threshold"
/>
    </InputVars>
    <OutputVars>
      <VarDeclaration Name="PRESENT" Type="BOOL" Comment="Orange is Present" />
      <VarDeclaration Name="GREEN" Type="BOOL" Comment="Green is Above Threshold" />
    </OutputVars>
  </InterfaceList>
  <Service RightInterface="resource" LeftInterface="ORANGE_SENSOR" >
    <ServiceSequence Name="setup" >
      <ServiceTransaction >
        <InputPrimitive Interface="ORANGE_SENSOR" Event="INIT" Parameters="THRESH" />
        <OutputPrimitive Interface="resource" Event="init" />
        <OutputPrimitive Interface="ORANGE_SENSOR" Event="INITO" />
      </ServiceTransaction>
    </ServiceSequence>
    <ServiceSequence Name="sensing" >
      <ServiceTransaction >
        <InputPrimitive Interface="resource" Event="change" />
        <OutputPrimitive Interface="ORANGE_SENSOR" Event="IND"
Parameters="PRESENT,GREEN" />
      </ServiceTransaction>
    </ServiceSequence>
  </Service>
</FBType>
```

EXAMPLE 2 A service interface function block type for the actuator of a simple solenoid valve could be expressed textually as follows.

```
FUNCTION_BLOCK SOLENOID (* Solenoid Valve *)
EVENT_INPUT
    REQ WITH IN; (* Set Actuator Status *)
END_EVENT
EVENT_OUTPUT
    CNF; (* Actuator Status Change Confirmed *)
END_EVENT
VAR_INPUT
    IN: BOOL; (* Actuator Value,1=OPEN,0=CLOSED *)
END_VAR
SERVICE SOLENOID/resource
SEQUENCE actuation
    SOLENOID.REQ(IN) -> resource.actuate(IN) -> SOLENOID.CNF();
END_SEQUENCE
END_SERVICE
END_FUNCTION_BLOCK
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

The interface and service sequence would appear graphically as follows.

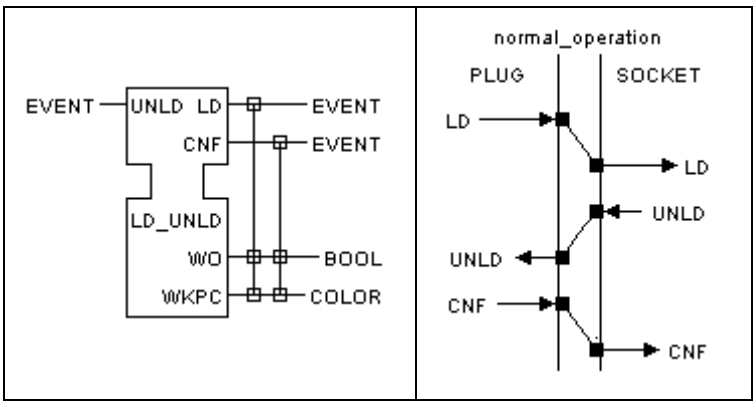


A corresponding XML document would be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE FBType SYSTEM "../LibraryElement.dtd" >
<FBType Name="SOLENOID" Comment="Solenoid Valve" >
  <Identification Classification="C0403" ApplicationDomain="Any" Function="Logic / I/O
Modules &#38; Controllers" Type="Actuators" Description="Solenoid Valve" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC" Date="2000-02-
03" Remarks="Corrected service sequence" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC" Date="2000-01-
26" />
  <CompilerInfo header="package fb.rt.part2;" >
  </CompilerInfo>
  <InterfaceList>
    <EventInputs>
      <Event Name="REQ" Comment="Set Actuator Status" >
        <With Var="IN" />
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Name="CNF" Comment="Actuator Status Change Confirmed" >
      </Event>
    </EventOutputs>
    <InputVars>
      <VarDeclaration Name="IN" Type="BOOL" Comment="Actuator Value,1=OPEN,0=CLOSED"
/>
    </InputVars>
  </InterfaceList>
  <Service RightInterface="resource" LeftInterface="SOLENOID" >
    <ServiceSequence Name="actuation" >
      <ServiceTransaction >
        <InputPrimitive Interface="SOLENOID" Event="REQ" Parameters="IN" />
        <OutputPrimitive Interface="resource" Event="actuate" Parameters="IN" />
        <OutputPrimitive Interface="SOLENOID" Event="CNF" />
      </ServiceTransaction>
    </ServiceSequence>
  </Service>
</FBType>
```

C.3 An adapter interface type

EXAMPLE An adapter interface for use in parts transfer simulations, its typical sequence of operation, and its corresponding XML document, may be as shown below.



IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE AdapterType SYSTEM "../LibraryElement.dtd" >
<AdapterType Name="LD_UNLD" Comment="LOAD/UNLOAD Adapter Interface" >
  <Identification Standard="IEC 61499-2" />
  <VersionInfo Organization="Rockwell Automation" Version="0.0" Author="JHC"
Date="1999-11-17" Remarks="Generated by FBEditor application" />
  <CompilerInfo header="package fb.rt.omac;" >
    <Compiler Language="Java" Vendor="IBM" Product="VisualAge" Version="2.0" />
  </CompilerInfo>
  <InterfaceList>
    <EventInputs>
      <Event Name="UNLD" Comment="UNLOAD Request" >
      </Event>
    </EventInputs>
    <EventOutputs>
      <Event Name="LD" Comment="LOAD Request" >
        <With Var="WO" />
        <With Var="WKPC" />
      </Event>
      <Event Name="CNF" Comment="UNLD Confirm" >
        <With Var="WO" />
        <With Var="WKPC" />
      </Event>
    </EventOutputs>
    <OutputVars>
      <VarDeclaration Name="WO" Type="BOOL" Comment="Workpiece present" />
      <VarDeclaration Name="WKPC" Type="COLOR" Comment="Workpiece Color" />
    </OutputVars>
  </InterfaceList>
  <Service RightInterface="SOCKET" LeftInterface="PLUG" >
    <ServiceSequence Name="normal_operation" >
      <ServiceTransaction >
        <InputPrimitive Interface="PLUG" Event="LD" Parameters="WO,WKPC" />
        <OutputPrimitive Interface="SOCKET" Event="LD" Parameters="WO,WKPC" />
      </ServiceTransaction>
      <ServiceTransaction >
        <InputPrimitive Interface="SOCKET" Event="UNLD" />
        <OutputPrimitive Interface="PLUG" Event="UNLD" />
      </ServiceTransaction>
      <ServiceTransaction >
        <InputPrimitive Interface="PLUG" Event="CNF" />
        <OutputPrimitive Interface="SOCKET" Event="CNF" />
      </ServiceTransaction>
    </ServiceSequence>
  </Service>
</AdapterType>
```

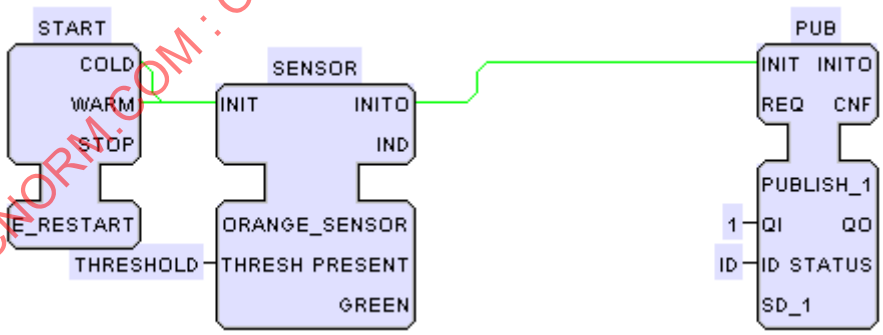
IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

C.4 Resource types

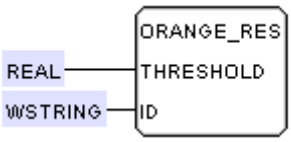
EXAMPLE 1 A resource type containing an instance of the ORANGE_SENSOR function block type defined in Clause C.2, plus an instance of the PUBLISH_1 type to transmit a change in condition, could be declared textually as shown below. This resource type also contains an instance of the E_RESTART type defined in Annex A of IEC 61499-1, interconnected to provide initialization of the other function block instances. The data outputs of the ORANGE_SENSOR block and the SD_1 input of the PUBLISH_1 block, and their corresponding event inputs and outputs, are left unconnected in order to allow application-specific logic to determine the value to be transmitted and the event to trigger the transmission.

```
  RESOURCE_TYPE ORANGE_RES (* A Configurable Orange Presence&Quality
  Sensor *)
  VAR_INPUT
      THRESHOLD: REAL; (* Adjustable Color Threshold *)
      ID: WSTRING; (* UDP Channel ID *)
  END_VAR
  FB_TYPES
    E_RESTART;
    ORANGE_SENSOR;
    PUBLISH_1;
  END_FB_TYPES
  FBS
      START: E_RESTART;
      SENSOR: ORANGE_SENSOR;
      PUB: PUBLISH_1(
          QI:= 1);
  END_FBS
  EVENT_CONNECTIONS
      START.COLD TO SENSOR.INIT;
      START.WARM TO SENSOR.INIT;
      SENSOR.INITO TO PUB.INIT;
  END_CONNECTIONS
  DATA_CONNECTIONS
      THRESHOLD TO SENSOR.THRESH;
      ID TO PUB.ID;
  END_CONNECTIONS
END_RESOURCE_TYPE
```

A graphical representation of this resource type's function block network is:



The resource type's external interface could be represented as:



IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

An equivalent XML document (with additional information for software tools) could be:

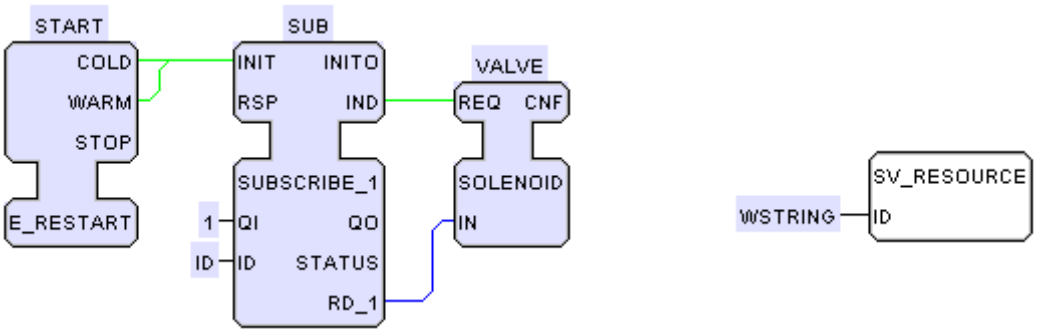
```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ResourceType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd" >
<ResourceType Name="ORANGE_RES" Comment="A Configurable Orange Presence&#38;Quality
Sensor" >
  <Identification Standard="IEC 61499-1" Classification="C0202"
ApplicationDomain="Food Processing" Function="Detection" Type="Photoelectric Sensors"
Description="Orange Presence and Quality" />
  <VersionInfo Organization="IEC SC65B/WG15" Version="0.2" Author="JHC" Date="2011-02-
23" Remarks="Added THRESHOLD, ID parameters." />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC" Date="2000-06-
20" Remarks="Corrected &#34;FBType&#34; to &#34;FBTypeName&#34;." />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC" Date="2000-02-
02" />
  <CompilerInfo header="package fb.rt.part2;" >
  </CompilerInfo>
  <FBTypeName Name="E_RESTART" />
  <FBTypeName Name="ORANGE_SENSOR" />
  <FBTypeName Name="PUBLISH_1" />
  <VarDeclaration Name="THRESHOLD" Type="REAL" Comment="Adjustable Color Threshold" />
  <VarDeclaration Name="ID" Type="WSTRING" Comment="UDP Channel ID" />
  <FBNetwork >
    <FB Name="START" Type="E_RESTART" x="94.44444" y="11.111111" >
    </FB>
    <FB Name="SENSOR" Type="ORANGE_SENSOR" x="672.22217" y="122.22221" >
    </FB>
    <FB Name="PUB" Type="PUBLISH_1" x="2172.2222" y="11.111111" >
      <Parameter Name="QI" Value="1" />
    </FB>
    <EventConnections>
      <Connection Source="START.COLD" Destination="SENSOR.INIT" dx1="33.333332"
dx2="47.0588" dy="-70.5882" />
      <Connection Source="START.WARM" Destination="SENSOR.INIT" dx1="61.111107"
dx2="76.4706" dy="-188.2353" />
      <Connection Source="SENSOR.INITO" Destination="PUB.INIT" dx1="172.22221" />
    </EventConnections>
    <DataConnections>
      <Connection Source="THRESHOLD" Destination="SENSOR.THRESH" />
      <Connection Source="ID" Destination="PUB.ID" />
    </DataConnections>
  </FBNetwork>
</ResourceType>
```

EXAMPLE 2 A resource type containing an instance of the SOLENOID function block type defined in Clause C.2, plus an instance of the SUBSCRIBE_1 type to receive a command to change solenoid status, could be declared textually as shown below. This resource type also contains an instance of the E_RESTART type defined in Annex A of IEC 61499-1, interconnected to provide initialization of the other function block instances.

```
  RESOURCE_TYPE SV_RESOURCE (* A Remotely Activated
  Solenoid Valve Resource *)
  VAR_INPUT
      ID: WSTRING; (* UDP Channel ID *)
  END_VAR
  FBS
      START: E_RESTART;
      SUB: SUBSCRIBE_1(
          QI:= 1);
      VALVE: SOLENOID;
  END_FBS
  EVENT_CONNECTIONS
      START.COLD TO SUB.INIT;
      START.WARM TO SUB.INIT;
      SUB.IND TO VALVE.REQ;
  END_CONNECTIONS
  DATA_CONNECTIONS
      SUB.RD_1 TO VALVE.IN;
      ID TO SUB.ID;
  END_CONNECTIONS
END_RESOURCE_TYPE
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Graphical representations of this resource's function block network and external interface can be as follows:



An equivalent XML document (with additional information for software tools) could be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ResourceType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd" >
<ResourceType Name="SV_RESOURCE" Comment="A Remotely Activated Solenoid Valve
Resource" >
  <Identification Standard="IEC 61499-1" Classification="C0403"
ApplicationDomain="Any" Function="Logic / I/O Modules &#38; Controllers"
Type="Actuators" Description="Solenoid Valve" />
  <VersionInfo Organization="IEC SC65B/WG15" Version="0.2" Author="JHC" Date="2011-02-
23" Remarks="Added ID parameter." />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC" Date="2000-06-
20" Remarks="Now uses LibraryElement.dtd" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC" Date="2000-02-
02" />
  <CompilerInfo header="package fb.rt.part2;" >
  </CompilerInfo>
  <VarDeclaration Name="ID" Type="WSTRING" Comment="UDP Channel ID" />
  <FBNetwork >
    <FB Name="START" Type="E_RESTART" x="72.22222" y="11.111111" >
    </FB>
    <FB Name="SUB" Type="SUBSCRIBE_1" x="705.55554" y="11.111111" >
      <Parameter Name="QI" Value="1" />
    </FB>
    <FB Name="VALVE" Type="SOLENOID" x="1316.6666" y="122.22221" >
    </FB>
    <EventConnections>
      <Connection Source="START.COLD" Destination="SUB.INIT" dx1="33.333332"
dx2="41.1765" dy="-64.7059" />
      <Connection Source="START.WARM" Destination="SUB.INIT" dx1="61.111107"
dx2="64.7059" dy="-182.3529" />
      <Connection Source="SUB.IND" Destination="VALVE.REQ" dx1="38.888885" />
    </EventConnections>
    <DataConnections>
      <Connection Source="SUB.RD_1" Destination="VALVE.IN" dx1="133.33333" />
      <Connection Source="ID" Destination="SUB.ID" />
    </DataConnections>
  </FBNetwork>
</ResourceType>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

C.5 Device types

EXAMPLE 1 A device type containing an instance of the ORANGE_RES resource type defined in Clause C.4 could be declared textually as shown below.

```
DEVICE_TYPE ORANGE_EYE
    (* Programmable Orange Presence+Quality Sensor *)
RESOURCE_TYPES
  ORANGE_RES;
END_RESOURCE_TYPES
RESOURCE R1: ORANGE_RES
END_RESOURCE
END_DEVICE_TYPE
```

An equivalent XML document (with additional information for software tools) could be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DeviceType SYSTEM "../LibraryElement.dtd" >
<DeviceType Name="ORANGE_EYE" Comment="Programmable Orange Presence+Quality Sensor" >
  <Identification Classification="C0202" ApplicationDomain="Food Processing"
Function="Detection" Type="Photoelectric Sensors" Description="Orange Presence and
Quality" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC" Date="2000-06-
20" Remarks="Corrected &#34;ResourceTypeName&#34;" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC" Date="2000-02-
02" />
  <CompilerInfo header="package fb.rt.part2;" >
  </CompilerInfo>
  <ResourceTypeName Name="ORANGE_RES" />
  <Resource Name="R1" Type="ORANGE_RES" >
  </Resource>
</DeviceType>
```

EXAMPLE 2 A device type containing an instance of the SV_RESOURCE type defined in Clause C.4 could be declared textually as shown below.

```
DEVICE_TYPE SOLENOID_VALVE
   (* A Remotely Activated Solenoid Valve *)
RESOURCE R1: SV_RESOURCE
END_RESOURCE
END_DEVICE_TYPE
```

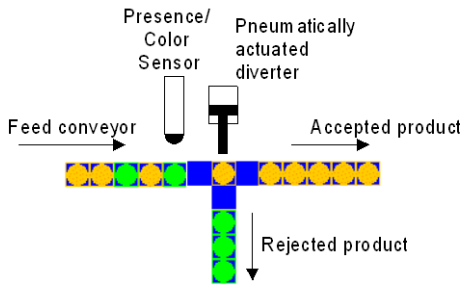
An equivalent XML document (with additional information for software tools) could be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DeviceType SYSTEM "../LibraryElement.dtd" >
<DeviceType Name="SOLENOID_VALVE" Comment="A Remotely
Activated Solenoid Valve" >
  <Identification Classification="C0403"
ApplicationDomain="Any" Function="Logic / I/O Modules
&#38; Controllers" Type="Actuators"
Description="Solenoid Valve" />
  <VersionInfo Organization="IEC TC65/WG6"
Version="0.1" Author="JHC" Date="2000-06-20"
Remarks="Now uses LibraryElement.dtd" />
  <VersionInfo Organization="IEC TC65/WG6"
Version="0.0" Author="JHC" Date="2000-02-02" />
  <CompilerInfo header="package fb.rt.part2;" >
  </CompilerInfo>
  <Resource Name="R1" Type="SV_RESOURCE" >
  </Resource>
</DeviceType>
```

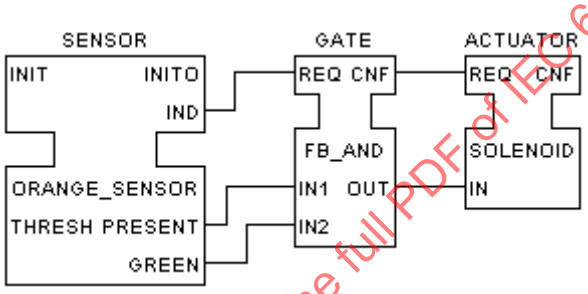
IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

C.6 A system configuration

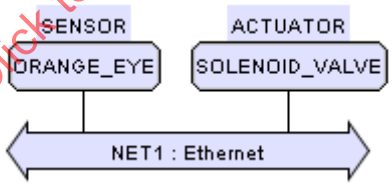
EXAMPLE Oranges are to be sorted by sensing their position and color and diverting any which are too green, as illustrated below.



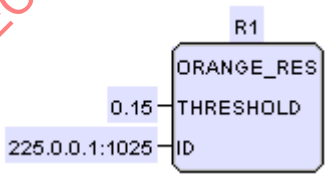
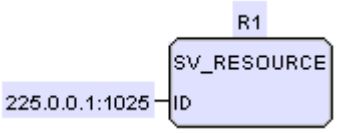
An *application* which can perform this task, using the ORANGE_SENSOR and SOLENOID function block types described in Clause C.2, could be as follows.



A *system* to implement this application, using the previously defined ORANGE_EYE and SOLENOID_VALVE device types, in conjunction with the Ethernet segment type defined in Clause C.7, could have the following top-level *configuration*:

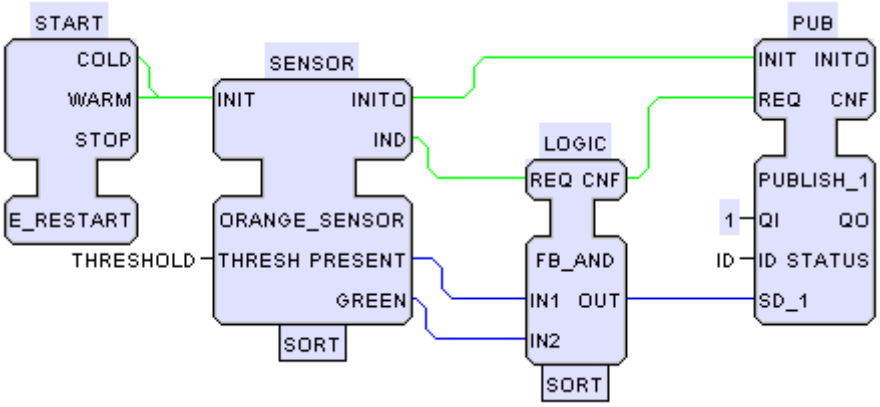


The SENSOR:ORANGE_EYE and ACTUATOR:SOLENOID_VALVE devices could in turn be configured as follows:

SENSOR:ORANGE_EYE	ACTUATOR:SOLENOID_VALVE
	

The resource ACTUATOR.R1:SV_RESOURCE can be used unmodified as defined in Clause C.4, while the resource SENSOR.R1:ORANGE_RES can be configured as follows to implement the application logic. Note the additional tags at the bottom of the SENSOR and LOGIC blocks, indicating that they are associated with the SORT application; this is a "navigation" feature as described in EXAMPLE 2 of 4.4.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012



IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

A *system configuration* implementing the features described above could appear as follows in textual form:

```
  SYSTEM ORANGE_SORTER (* Orange Sorter System Configuration *)
  APPLICATION SORT
  FBS
      SENSOR: ORANGE_SENSOR;
      GATE: FB_AND;
      ACTUATOR: SOLENOID;
  END_FBS
  EVENT_CONNECTIONS
      SENSOR.IND TO GATE.REQ;
      GATE.CNF TO ACTUATOR.REQ;
  END_CONNECTIONS
  DATA_CONNECTIONS
      SENSOR.PRESENT TO GATE.IN1;
      SENSOR.GREEN TO GATE.IN2;
      GATE.OUT TO ACTUATOR.IN;
  END_CONNECTIONS
  END_APPLICATION
  DEVICE SENSOR: ORANGE_EYE
  RESOURCE R1: ORANGE_RES(
          THRESHOLD:= 0.15,
          ID:= 225.0.0.1:1025)
  FBS
      LOGIC: FB_AND;
  END_FBS
  EVENT_CONNECTIONS
      SENSOR.IND TO LOGIC.REQ;
      LOGIC.CNF TO PUB.REQ;
  END_CONNECTIONS
  DATA_CONNECTIONS
      SENSOR.PRESENT TO LOGIC.IN1;
      SENSOR.GREEN TO LOGIC.IN2;
      LOGIC.OUT TO PUB.SD_1;
  END_CONNECTIONS
  END_RESOURCE
  END_DEVICE
  DEVICE ACTUATOR: SOLENOID_VALVE
  RESOURCE R1: SV_RESOURCE(
          ID:= 225.0.0.1:1025)
  END_RESOURCE
  END_DEVICE
  MAPPING
   SORT.SENSOR ON SENSOR.R1.SENSOR;
   SORT.GATE ON SENSOR.R1.LOGIC;
   SORT.ACTUATOR ON ACTUATOR.R1.VALVE;
  END_MAPPING
  SEGMENTS
      NET1: Ethernet;
  END_SEGMENTS
  LINKS
      SENSOR => NET1;
      ACTUATOR => NET1;
  END_LINKS
END_SYSTEM
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

A corresponding XML document would then be:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE System SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd" >
<System Name="ORANGE_SORTER" Comment="Orange Sorter System Configuration" >
  <Identification Standard="61499-2" />
  <VersionInfo Organization="IEC SC65B/WG15" Version="0.2" Author="JHC" Date="2011-02-
23" Remarks="Moved parameters to Resource level." />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.1" Author="JHC" Date="2000-06-
20" Remarks="Now uses LibraryElement.dtd" />
  <VersionInfo Organization="IEC TC65/WG6" Version="0.0" Author="JHC" Date="2000-02-
03" />
  <Application Name="SORT" >
    <SubAppNetwork >
      <FB Name="SENSOR" Type="ORANGE_SENSOR" x="94.1176" y="11.7647" >
      </FB>
      <FB Name="GATE" Type="FB_AND" x="941.1765" y="11.7647" >
      </FB>
      <FB Name="ACTUATOR" Type="SOLENOID" x="1441.1764" y="11.7647" >
      </FB>
      <EventConnections>
        <Connection Source="SENSOR.IND" Destination="GATE.REQ" dx1="88.2353"
dx2="47.0588" dy="-188.2353" />
        <Connection Source="GATE.CNF" Destination="ACTUATOR.REQ" dx1="41.1765" />
      </EventConnections>
      <DataConnections>
        <Connection Source="SENSOR.PRESENT" Destination="GATE.IN1" dx1="64.7059"
dx2="176.4706" dy="982.3529" />
        <Connection Source="SENSOR.GREEN" Destination="GATE.IN2" dx1="123.5294"
dx2="158.8235" dy="870.5882" />
        <Connection Source="GATE.OUT" Destination="ACTUATOR.IN" dx1="152.9412" />
      </DataConnections>
    </SubAppNetwork>
  </Application>
  <Device Name="SENSOR" Type="ORANGE_EYE" x="122.22221" y="11.111111" >
    <Resource Name="R1" Type="ORANGE_RES" x="577.7778" y="11.111111" >
      <Parameter Name="THRESHOLD" Value="0.15" />
      <Parameter Name="ID" Value="225.0.0.1:1025" />
      <FBNetwork >
        <FB Name="LOGIC" Type="FB_AND" x="1538.8888" y="344.44443" >
        </FB>
        <EventConnections>
          <Connection Source="SENSOR.IND" Destination="LOGIC.REQ" dx1="44.444443"
dx2="58.8235" dy="-588.2353" />
          <Connection Source="LOGIC.CNF" Destination="PUB.REQ" dx1="61.111107" />
        </EventConnections>
        <DataConnections>
          <Connection Source="SENSOR.PRESENT" Destination="LOGIC.IN1" dx1="83.33333"
dx2="64.7059" dy="576.4706" />
          <Connection Source="SENSOR.GREEN" Destination="LOGIC.IN2" dx1="38.888885"
dx2="58.8235" dy="458.8235" />
          <Connection Source="LOGIC.OUT" Destination="PUB.SD_1" dx1="100.0" />
        </DataConnections>
      </FBNetwork>
    </Resource>
  </Device>
  <Device Name="ACTUATOR" Type="SOLENOID_VALVE" x="627.7778" y="11.111111" >
    <Resource Name="R1" Type="SV_RESOURCE" x="705.55554" y="22.222221" >
      <Parameter Name="ID" Value="225.0.0.1:1025" />
    </Resource>
  </Device>
  <Mapping From="SORT.SENSOR" To="SENSOR.R1.SENSOR" />
  <Mapping From="SORT.GATE" To="SENSOR.R1.LOGIC" />
  <Mapping From="SORT.ACTUATOR" To="ACTUATOR.R1.VALVE" />
  <Segment Name="NET1" Type="Ethernet" x="616.6666" y="411.11108" dx1="999.99994" >
  </Segment>
  <Link CommResource="SENSOR" SegmentName="NET1" >
  </Link>
  <Link CommResource="ACTUATOR" SegmentName="NET1" >
  </Link>
</System>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

C.7 A SegmentType definition

EXAMPLE The `Ethernet` segment type used in Clause C.6 can be defined textually and in XML as shown below.

```
SEGMENT_TYPE Ethernet
VAR
    PHY: WSTRING:= "10BASE-T"; (* Physical medium *)
    Speed: REAL:= 10.0; (* Mbits/sec *)
    Length: REAL:= 25.0; (* Length in metres *)
END_VAR
END_SEGMENT_TYPE
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE SegmentType SYSTEM "http://www.holobloc.com/xml/LibraryElement.dtd" >
<SegmentType Name="Ethernet" >
  <Identification Standard="IEEE 802.3" ApplicationDomain="Networking" />
  <VersionInfo Organization="Holobloc Inc" Version="0.0" Author="JHC"
Date="2011-02-09" />
  <CompilerInfo header="package fb.rt.net;" >
  </CompilerInfo>
  <VarDeclaration Name="PHY" Type="WSTRING" InitialValue="10BASE-T"
Comment="Physical medium" />
  <VarDeclaration Name="Speed" Type="REAL" InitialValue="10.0"
Comment="Mbits/sec" />
  <VarDeclaration Name="Length" Type="REAL" InitialValue="25.0" Comment="Length
in metres" />
</SegmentType>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Bibliography

IEC 61499-4, *Function blocks – Part 4: Rules for compliance profiles*

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

SOMMAIRE

AVANT-PROPOS 49
INTRODUCTION 51
1 Domaine d'application 52
2 Références normatives 52
3 Termes et définitions 52
4 Exigences pour les outils logiciels 53
4.1 Informations devant être fournies par le fournisseur d'outil logiciel 53
4.2 Echange d'éléments bibliothèques 53
4.3 Informations devant être fournies par le fournisseur d'éléments bibliothèques 53
4.4 Affichage des déclarations 53
4.5 Modification de déclarations 54
4.6 Validation de déclarations 54
4.7 Mise en œuvre de déclarations 54
4.8 Exploitation, essais et maintenance de système 55
Annexe A (normative) Définitions de type de document (les DTD) 56
Annexe B (informative) Modèle de graphique 75
Annexe C (informative) Exemples 79
Bibliography 97

Figure B.1 – Modèle de graphique 75
Figure B.2 – Exemple de tracé de graphiques ECC 77

Tableau A.1 – Eléments de définition de type de document (DTD) 56
Tableau A.2 – DTD DataType (1 de 2) 57
Tableau A.3 – DataType des éléments de DTD (1 de 3) 59
Tableau A.4 – DTD des éléments de bibliothèque (1 de 5) 63
Tableau A.5 – LibraryElement éléments de DTD (1 de 6) 68

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

BLOCS FONCTIONNELS –

Partie 2: Exigences pour les outils logiciels

AVANT-PROPOS

1) La Commission Electrotechnique Internationale (CEI) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de la CEI). La CEI a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. A cet effet, la CEI – entre autres activités – publie des Normes internationales, des Spécifications techniques, des Rapports techniques, des Spécifications accessibles au public (PAS) et des Guides (ci-après dénommés "Publication(s) de la CEI"). Leur élaboration est confiée à des comités d'études, aux travaux desquels tout Comité national intéressé par le sujet traité peut participer. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec la CEI, participent également aux travaux. La CEI collabore étroitement avec l'Organisation Internationale de Normalisation (ISO), selon des conditions fixées par accord entre les deux organisations.
2) Les décisions ou accords officiels de la CEI concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de la CEI intéressés sont représentés dans chaque comité d'études.
3) Les Publications de la CEI se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de la CEI. Tous les efforts raisonnables sont entrepris afin que la CEI s'assure de l'exactitude du contenu technique de ses publications; la CEI ne peut pas être tenue responsable de l'éventuelle mauvaise utilisation ou interprétation qui en est faite par un quelconque utilisateur final.
4) Dans le but d'encourager l'uniformité internationale, les Comités nationaux de la CEI s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de la CEI dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de la CEI et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
5) La CEI elle-même ne fournit aucune attestation de conformité. Des organismes de certification indépendants fournissent des services d'évaluation de conformité et, dans certains secteurs, accèdent aux marques de conformité de la CEI. La CEI n'est responsable d'aucun des services effectués par les organismes de certification indépendants.
6) Tous les utilisateurs doivent s'assurer qu'ils sont en possession de la dernière édition de cette publication.
7) Aucune responsabilité ne doit être imputée à la CEI, à ses administrateurs, employés, auxiliaires ou mandataires, y compris ses experts particuliers et les membres de ses comités d'études et des Comités nationaux de la CEI, pour tout préjudice causé en cas de dommages corporels et matériels, ou de tout autre dommage de quelque nature que ce soit, directe ou indirecte, ou pour supporter les coûts (y compris les frais de justice) et les dépenses découlant de la publication ou de l'utilisation de cette Publication de la CEI ou de toute autre Publication de la CEI, ou au crédit qui lui est accordé.
8) L'attention est attirée sur les références normatives citées dans cette publication. L'utilisation de publications référencées est obligatoire pour une application correcte de la présente publication.
9) L'attention est attirée sur le fait que certains des éléments de la présente Publication de la CEI peuvent faire l'objet de droits de brevet. La CEI ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

La Norme internationale CEI 61499-2 a été établie par le sous-comité 65B: Equipements de mesure et de contrôle-commande du comité d'études 65 de la CEI: Mesure, commande et automation dans les processus industriels.

Cette deuxième édition annule et remplace la première édition parue en 2005.

Cette édition inclut les modifications techniques majeures suivantes par rapport à l'édition précédente:

- Le contenu de l'Annexe A a été mis à jour pour être en conformité avec les modifications techniques apportées à la deuxième édition de la CEI 61499-1.
- Les sections CDATA sont désormais autorisées pour les contenus textuels des algorithmes des Tableaux A.4 et A.5.

Le texte de cette norme est issu des documents suivants:

FDIS	Rapport de vote
65B/846/FDIS	65B/856/RVD

Le rapport de vote indiqué dans le tableau ci-dessus donne toute information sur le vote ayant abouti à l'approbation de cette norme.

Cette publication a été rédigée selon les Directives ISO/CEI, Partie 2.

Une liste de toutes les parties de la série CEI 61499, présentées sous le titre général *Blocs fonctionnels*, peut être consultée sur le site web de la CEI.

Les termes utilisés dans la présente Norme internationale, définis à l'Article 3 de la CEI 61499-1:2012 ainsi que dans la présente Norme internationale, sont présentés en *italiques*.

Le comité a décidé que le contenu de cette publication ne sera pas modifié avant la date de stabilité indiquée sur le site web de la CEI sous "http://webstore.iec.ch" dans les données relatives à la publication recherchée. À cette date, la publication sera

- reconduite,
- supprimée,
- remplacée par une édition révisée, ou
- amendée.

IMPORTANT – Le logo "colour inside" qui se trouve sur la page de couverture de cette publication indique qu'elle contient des couleurs qui sont considérées comme utiles à une bonne compréhension de son contenu. Les utilisateurs devraient, par conséquent, imprimer cette publication en utilisant une imprimante couleur.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

INTRODUCTION

La CEI 61499 se compose des parties suivantes, sous le titre général *Blocs fonctionnels*:

- Partie 1: Architecture
- Partie 2: Exigences pour les outils logiciels
- Partie 3: Informations tutorielles (annulée)
- Partie 4: Règles pour les profils de conformité

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

BLOCS FONCTIONNELS –

Partie 2: Exigences pour les outils logiciels

1 Domaine d'application

La présente partie de la CEI 61499 définit les exigences pour les *outils logiciels* pour prendre en charge les tâches d'ingénierie système suivantes qui sont données dans la CEI 61499-1:

- la spécification de *types de bloc fonctionnel*;
- la spécification fonctionnelle de *types de ressource* et de *types d'équipement*;
- la spécification, l'analyse et la validation des systèmes IPMCS distribués;
- la *configuration*, la *mise en œuvre*, l'exploitation et la maintenance des systèmes IPMCS distribués;
- l'échange d'*informations* parmi les *outils logiciels*.

Il est admis que ces outils logiciels peuvent être utilisés dans le contexte d'un système de support d'ingénierie (*Engineering Support System* (ESS)) comme décrit dans la CEI 61499-1.

Il ne relève pas du domaine d'application de la présente norme de spécifier la totalité du cycle de vie des systèmes de mesure et de commande dans les processus industriels (les IPMCS) ou l'ensemble du jeu des tâches et des activités requises pour prendre en charge un système IPCMS pendant tout son cycle de vie. Cependant, d'autres normes spécifiant effectivement de telles tâches et activités peuvent étendre ou modifier les exigences spécifiées dans la présente partie de la CEI 61499.

2 Références normatives

Les documents suivants sont cités en référence de manière normative, en intégralité ou en partie, dans le présent document et sont indispensables pour son application. Pour les références datées, seule l'édition citée s'applique. Pour les références non datées, la dernière édition du document de référence s'applique (y compris les éventuels amendements).

CEI 61131-3:2003, *Programmable controllers – Part 3: Programming languages* (disponible en anglais seulement)

CEI 61499-1:2012, *Blocs fonctionnels – Partie 1: Architecture*

ISO/CEI 8824 (toutes les parties), *Technologies de l'information – Notation de syntaxe abstraite numéro un (ASN.1)*

3 Termes et définitions

Pour les besoins du présent document, les termes et définitions donnés dans la CEI 61499-1 ainsi que les termes et définitions suivants s'appliquent:

3.1
élément bibliothèque
ensemble de *déclarations* s'appliquant à un *type de données, type de bloc fonctionnel, type d'adaptateur, type de sous-application, type de ressource, type d'équipement, type de segment* ou une *configuration système*.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

4 Exigences pour les outils logiciels

4.1 Informations devant être fournies par le fournisseur d'outil logiciel

Le présent Article définit les exigences fonctionnelles relatives aux *outils logiciels* qui supportent l'accomplissement des tâches d'ingénierie système énumérées à l'Article 1.

Le fournisseur d'un *outil logiciel* doit spécifier les informations suivantes, en plus des autres informations requises dans le présent Article:

- le(s) type(s) d'*élément bibliothèque* auquel l'outil logiciel s'applique;
- la/les tâche(s) d'ingénierie soutenue(s) par l'outil logiciel. Les descriptions de tâches peuvent être prises dans l'énumération des tâches d'ingénierie donnée à l'Article 1 ou peuvent être définies par le fournisseur.

4.2 Echange d'éléments bibliothèques

Un *outil logiciel* doit être capable d'échanger ses *éléments bibliothèques* avec d'autres outils logiciels. Cet échange doit prendre la forme de *données* au format défini dans l'Annexe A, inscrites sur des supports physiques ou échangées sur des liaisons ou des réseaux de communication.

4.3 Informations devant être fournies par le fournisseur d'éléments bibliothèques

NOTE Les dispositions du présent paragraphe visent à fournir le moyen par lequel le fournisseur d'un élément bibliothèque peut sauvegarder la protection de la propriété intellectuelle tout en fournissant des informations suffisantes pour permettre l'utilisation efficace de l'élément bibliothèque.

Le fournisseur d'un *élément bibliothèque* peut choisir de fournir une *mise en œuvre* de l'élément bibliothèque.

EXEMPLE 1 Le fournisseur d'un élément bibliothèque de *type de bloc fonctionnel* peut fournir une mise en œuvre du type de bloc fonctionnel sous la forme:

- d'une ou plusieurs *instances* du type de bloc fonctionnel dans une *ressource* contenue dans un *équipement* de Classe 0 ou supérieure telle que décrite dans la CEI 61499-4;
- d'une mise en œuvre instanciable du type de bloc fonctionnel dans une *ressource* contenue dans un *équipement* de Classe 1 ou supérieure telle que décrite dans la CEI 61499-4;
- d'un fichier avec un format **dépendant de la mise en œuvre** adapté à une installation pour une *ressource* contenue dans un *équipement* de Classe 2 telle que décrite dans la CEI 61499-4, par exemple en utilisant la syntaxe XML définie dans l'Annexe D.

Lorsqu'une mise en œuvre d'un élément bibliothèque est fournie, le fournisseur n'est pas tenu de donner les détails complets de la mise en œuvre. Cependant, le fournisseur doit donner suffisamment d'informations pour permettre à l'utilisateur de déterminer complètement la fonctionnalité de l'élément bibliothèque fourni.

EXEMPLE 2 L'exigence dans l'alinéa ci-dessus serait satisfaite par le fournisseur d'une *instance* d'un *type* de bloc fonctionnel d' une *ressource* par au minimum la fourniture des informations suivantes:

- un élément bibliothèque de *type de bloc fonctionnel* spécifiant ses *interfaces d'événements,* ses *interfaces de données* et ses *services* tels que définis dans la CEI 61499-1;
- des éléments bibliothèques de *type ressource* et de *type d'équipement* montrant l'occurrence et les connexions des *instances* du bloc fonctionnel.

4.4 Affichage des déclarations

Un outil logiciel doit être capable d'afficher les *déclarations* de ses *éléments bibliothèques* associés, et ce, dans une forme appropriée à la tâche d'ingénierie. Cet affichage peut utiliser les formats graphiques ou textuels définis dans la CEI 61499-1 ou un format défini par le fournisseur de l'outil logiciel.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

NOTE Les *déclarations* d'un élément bibliothèque peuvent définir ses *interfaces* (entrées et sorties d'événements et de données) et ses *variables* internes ainsi que ses *algorithmes* et le contrôle de leur *exécution*, par exemple par le biais d'un *graphique de contrôle d'exécution* (*Execution Control Chart* (ECC)), etc.

Les outils logiciels peuvent fournir des caractéristiques supplémentaires, au-delà de celles qui sont illustrées dans la CEI 61499-1, dans l'affichage graphique des déclarations.

EXEMPLE 1 Dans l'affichage d'un graphique de contrôle d'exécution (ECC), l'outil peut fournir, en même temps que l'affichage de chaque transition, un nombre cardinal indiquant l'ordre (tel que défini dans la CEI 61499-1) dans lequel la transition est évaluée.

EXEMPLE 2 Un outil logiciel peut fournir un moyen de naviguer dans un *mapping* à partir de l'affichage d'une instance de bloc fonctionnel dans une *application* jusqu'à son affichage correspondant dans une *ressource* et vice versa.

4.5 Modification de déclarations

Un outil logiciel doit permettre à son utilisateur de modifier les déclarations de ses éléments bibliothèques associés, de la manière appropriée pour la tâche d'ingénierie. De telles modifications peuvent comprendre l'ajout, la suppression ou le changement du contenu des déclarations et peuvent être accomplies graphiquement et/ou textuellement.

EXEMPLE L'outil logiciel fournit un moyen commode permettant à l'utilisateur de modifier l'ordre dans lequel les déclarations sont énumérées dans leur représentation textuelle, par exemple dans une liste de transitions dans un graphique de contrôle d'exécution (ECC), sans que l'utilisateur ait à modifier la représentation textuelle par un moyen manuel tel que "couper-coller".

4.6 Validation de déclarations

Si la tâche d'ingénierie associée l'exige, un outil logiciel doit fournir des moyens pour la validation des déclarations de ses éléments bibliothèques associés. De tels moyens peuvent comprendre, sans que cela soit exhaustif:

- la vérification de la correction de la syntaxe des déclarations;
- la vérification de la correction sémantique des déclarations, par exemple, en vérifiant si, oui ou non, toutes les *instances de bloc fonctionnel* dans une *application* et ses *sous-applications* associées sont correctement allouées à des *ressources*, interconnectées au sein de ressources, et en communiquant parmi les ressources dans une *configuration de système*.
- la simulation et des essais du fonctionnement d'une *instance* d'un *type* d'élément bibliothèque, soit toute seule, soit associée à d'autres instances du même type ou de types différents.

4.7 Mise en œuvre de déclarations

Si la tâche d'ingénierie associée l'exige, un outil logiciel doit fournir des moyens pour la *mise en œuvre* des *déclarations* des *éléments bibliothèques* associés. De tels moyens peuvent comprendre, sans que cela soit exhaustif:

- la production de code exécutable (*firmware* (micrologiciel)) pour une intégration dans des instances de types ressources et de types équipementss;
- la création et l'interconnexion (téléchargement) d'instances bloc fonctionnel dans des ressources et des équipements, par exemple en utilisant les moyens de gestion définis en 6.3 et dans les Annexes F et G de la CEI 61499-1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

4.8 Exploitation, essais et maintenance de système

Si la tâche d'ingénierie associée l'exige, un outil logiciel doit fournir des moyens pour l'exploitation, les essais et la maintenance d'un système de mesure et de commande dans les processus industriels (IPMCS) spécifié par ses éléments bibliothèques associés. De tels moyens peuvent comprendre, sans que cela soit exhaustif:

- les moyens décrits dans les paragraphes précédents du présent Article;
- les moyens d'échange d'informations définis dans la CEI 61499-1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Annexe A
(normative)

Définitions de type de document (les DTD)

A.1 Principes généraux

La présente Annexe présente les définitions de type de document (les DTD) pour l'échange d'éléments bibliothèques CEI 61499 entre des *outils logiciels*. Ces DTD sont définies dans la syntaxe définie dans la spécification du langage de balisage extensible (*eXtensible Markup Language* (XML)) à l'adresse www.w3.org/TR/2000/REC-xml-20001006.

Les correspondances entre les éléments de DTD donnés dans cette annexe, les éléments bibliothèques définis dans la CEI 61499-1-C.2.1, et la syntaxe textuelle donnée dans la CEI 61499-1, Annexe B sont présentées dans le Tableau A.1.

Tableau A.1 – Eléments de définition de type de document (DTD)

Elément de DTD	LibraryElement	Syntaxe textuelle
`DataType`	DataTypeDeclaration	`data_type_declaration` (CEI 61131-3, B.1.3)
`FBType`	FBTypeDeclaration	`fb_type_declaration`
`SubapplicationType`	SubapplicationTypeDeclaration	`subapplication_type_declaration`
`AdapterType`	AdapterTypeDeclaration	`adapter_type_declaration`
`ResourceType`	ResourceTypeDeclaration	`resource_type_specification`
`DeviceType`	DeviceTypeDeclaration	`device_type_specification`
`System`	SystemConfiguration	`system_configuration`

Le premier tableau de chaque paragraphe de la présente Annexe contient la DTD pour l'élément bibliothèque correspondant. Le deuxième tableau de chaque paragraphe fournit (le cas échéant) une référence à la syntaxe textuelle plus une explication des éléments et des attributs majeurs contenus dans la DTD. Ensuite, des exemples sont donnés et concernent les fichiers XML obtenus pour des éléments bibliothèques typiques.

NOTE 1 En cas de conflit entre les dispositions de la présente Annexe et celles de l'Annexe B de la CEI 61499-1, les dispositions de cette dernière prévalent.

NOTE 2 Les exemples donnés dans la présente Annexe donnent un échantillon représentatif, mais non exhaustif, des caractéristiques des DTD associées. En particulier, ces exemples ne sont pas censés être utilisés comme une série d'essais pour la conformité aux dispositions de la présente norme.

A.2 DTD DataType

Un document XML conforme à la DTD du Tableau A.2 représente un objet **DataTypeDeclaration** comme décrit en C.2.2 de la CEI 61499-1.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.2 – DTD DataType (1 de 2)

```
<?xml version="1.0" encoding="UTF-8"?>

<!ELEMENT DataType (Identification?, VersionInfo+, CompilerInfo?,
ASN1Tag?, (DirectlyDerivedType | EnumeratedType | SubrangeType |
ArrayType | StructuredType))>

<!ATTLIST DataType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT Identification EMPTY>
<!ATTLIST Identification
 Standard CDATA #IMPLIED
 Classification CDATA #IMPLIED
 ApplicationDomain CDATA #IMPLIED
 Function CDATA #IMPLIED
 Type CDATA #IMPLIED
 Description CDATA #IMPLIED>
```

```
<!ELEMENT VersionInfo EMPTY>
<!ATTLIST VersionInfo
 Organization CDATA #REQUIRED
 Version CDATA #REQUIRED
 Author CDATA #REQUIRED
 Date CDATA #REQUIRED
 Remarks CDATA #IMPLIED>
```

```
<!ELEMENT ASN1Tag EMPTY>
<!ATTLIST ASN1Tag
 Class (UNIVERSAL | APPLICATION | CONTEXT | PRIVATE) #IMPLIED
 Number CDATA #REQUIRED>
```

```
<!ELEMENT CompilerInfo (Compiler*)>
<!ATTLIST CompilerInfo
 header CDATA #IMPLIED
 classdef CDATA #IMPLIED>
```

```
<!ELEMENT Compiler EMPTY>
<!ATTLIST Compiler
 Language (Java | Cpp | C | Other) #REQUIRED
 Vendor CDATA #REQUIRED
 Product CDATA #REQUIRED
 Version CDATA #REQUIRED>
```

```
<!ELEMENT DirectlyDerivedType EMPTY>
<!ATTLIST DirectlyDerivedType
 BaseType (BOOL | SINT | INT | DINT | LINT | USINT | UINT | UDINT
| ULINT | REAL | LREAL | TIME | DATE | TIME_OF_DAY | TOD
| DATE_AND_TIME | DT | STRING | BYTE |WORD | DWORD |LWORD |
WSTRING) #REQUIRED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT EnumeratedType (EnumeratedValue+)>
<!ATTLIST EnumeratedType
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT EnumeratedValue EMPTY>
<!ATTLIST EnumeratedValue
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.2 (2 de 2)

```
<!ELEMENT SubrangeType (Subrange)>
<!ATTLIST SubrangeType
 BaseType (SINT|INT|DINT|LINT|USINT|UINT|UDINT|ULINT) #REQUIRED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT Subrange EMPTY>
<!ATTLIST Subrange
 LowerLimit CDATA #REQUIRED
 UpperLimit CDATA #REQUIRED>
```

```
<!ELEMENT ArrayType (Subrange+)>
<!ATTLIST ArrayType
 BaseType CDATA #REQUIRED
 InitialValues CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT StructuredType (VarDeclaration|SubrangeVarDeclaration)+>
<!ATTLIST StructuredType
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT VarDeclaration EMPTY >
<!ATTLIST VarDeclaration
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 ArraySize CDATA #IMPLIED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT SubrangeVarDeclaration (Subrange+) >
<!ATTLIST SubrangeVarDeclaration
 Name CDATA #REQUIRED
 Type (SINT|INT|DINT|LINT|USINT|UINT|UDINT|ULINT) #REQUIRED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

Le Tableau A.3 donne les explications des éléments de la DTD ci-dessus et (le cas échéant) les références à la syntaxe formelle pour leurs attributs.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.3 – DataType des éléments de DTD (1 de 3)

Elément attributs	Syntaxe textuelle (CEI 61131-3, Annexe B)	Explication
DataType		Voir CEI 61131-3
Name	data_type_name	
Comment	--	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)
Identification		Informations pour la récupération de base de données
Standard		Norme de référence principale au format numéro-partie-paragraphe
Classification		Code de classification tel que défini dans la norme de référence
ApplicationDomain		Domaine d'application tel que défini dans la norme de référence
Function		Fonction de cet élément telle que définie dans la norme de référence
Type		Type d'élément (par exemple, type d'équipement) tel que défini dans la norme de référence
Description		Expression descriptive telle que définie dans la norme de référence
VersionInfo	--	Eventuellement l'une parmi plusieurs entrées: Première entrée – Version la plus récente Deuxième entrée – Version publiée juste précédemment Dernière entrée – Première version publiée
Organization	--	L'organisation fournissant cet élément bibliothèque
Version	chiffre [chiffre] '.' chiffre [chiffre] [lettre]	L'identification de la version pour cet élément bibliothèque
Author	--	L'auteur de cet élément bibliothèque
Date	date_literal ['-' daytime]	La date de publication de cette version
Remarks	--	Commentaires relatifs à cette version
ASN1Tag		Étiquette ASN.1 selon l'ISO/CEI 8824
Class		Classe d'étiquette ASN.1 selon l'ISO/CEI 8824
Number		Numéro d'étiquette ASN.1 selon l'ISO/CEI 8824
CompilerInfo	--	Informations destinées et relatives aux compilateurs utilisés avec cette classe
header	--	Informations d'en-tête telles que paquet, importations, etc.
classdef	--	Les informations de définition de classe telles que la superclasse et les interfaces mises en œuvre. S'il n'en est donné aucune, une superclasse abstraite par défaut est utilisée.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.3 (2 de 3)

Elément attributs	Syntaxe textuelle (CEI 61131-3, Annexe B)	Explication
Compiler	--	Eventuellement l'un des différents compilateurs utilisés avec cette version
Language	--	Le langage source du compilateur
Vendor	--	Le vendeur du compilateur
Product	--	Le nom de produit du compilateur
Version	--	La version du compilateur
DirectlyDerivedType	Voir la CEI 61131-3, Tableaux 12 et 14, point 1	
BaseType	elementary_type_name	
InitialValue	constant (constante)	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
EnumeratedType	Voir la CEI 61131-3, Tableaux 12 et 14, point 2	
InitialValue	enumerated_value	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
EnumeratedValue	Voir la CEI 61131-3, Tableau 14, point 2	
Name	enumerated_value	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
SubrangeType	--	Voir la CEI 61131-3, Tableaux 12 et 14, point 3
BaseType	integer_type_name	
InitialValue	signed_integer	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
Subrange	Voir la CEI 61131-3, Tableaux 12 et 14, point 3	
LowerLimit	signed_integer	
UpperLimit	signed_integer	
ArrayType	Voir la CEI 61131-3, Tableaux 12 et 14, point 4	
BaseType	non_generic_type_name	
InitialValues	array_initialization	
StructuredType	Voir la CEI 61131-3, Tableaux 12, point 5 et 14, point 5 et point 6	
VarDeclaration		
Name	structure_element_name	
Type	non_generic_type_name	
ArraySize	a	
InitialValue	b	

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.3 (3 de 3)

Élément Attributs	Syntaxe textuelle (CEI 61131-3, Annexe B)	Explication
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
SubrangeVarDeclaration	Voir la CEI 61131-3, 2.3.3.	
Name	structure_element_name	
Type	integer_type_name	
InitialValue	signed_integer	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	

[a] La syntaxe de cet attribut lorsqu'il est présent doit être conforme à l'expression syntaxique (subrange {',' subrange}) | integer {',' integer} où les non-terminaux subrange et integer sont tels que définis dans l'Annexe B de la CEI 61131-3. Chaque terme de la seconde forme équivaut à la sous-plage 0..n-1, où n est la valeur de l'élément syntaxique integer correspondant. Si l'attribut est absent, le composant de structure n'est pas une matrice définie de façon anonyme.

[b] La syntaxe de cet attribut est la syntaxe pour l'initialisation du type de variable correspondant tel que défini dans en B.1.4.3 de la CEI 61131-3.

EXEMPLE L'exemple de type de donnée structuré ANALOG_CHANNEL_CONFIGURATIONI est exprimé dans la CEI 61131-3, Tableau 14 comme suit:

```
TYPE ANALOG_CHANNEL_CONFIGURATIONI:
    STRUCT
      RANGE: ANALOG_SIGNAL_RANGE;
      MIN_SCALE: ANALOG_DATA:= -4095;
      MAX_SCALE: ANALOG_DATA:=  4095;
    END_STRUCT;
END_TYPE
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Un document XML correspondant pourrait être:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE DataType SYSTEM
"DataType.dtd" >
<DataType
   Name="ANALOG_CHANNEL_CONFIGURATIONI"
   Comment="IEC 61131-3, Table 14#5">
<Identification
   Function="Configuration Data"
   Standard="61131-3-2.3.3.2"
   ApplicationDomain="Any"
   Classification="Data type"
   Type="Analog"
   Description="Table 14, #5"/>
<VersionInfo
   Organization="IEC SC65B/WG7/TF3"
   Version="2.0"
   Author="JHC"
   Date="2000-01-31"/>
<StructuredType>
  <VarDeclaration Name="SIGNAL_RANGE"
     Type="ANALOG_SIGNAL_RANGE"/>
  <VarDeclaration Name="MIN_SCALE"
     Type="ANALOG_DATA"
     InitialValue="-4095"/>
  <VarDeclaration Name="MAX_SCALE"
     Type="ANALOG_DATA"
     InitialValue="4095"/>
</StructuredType>
</DataType>
```

A.3 DTD LibraryElement

Un document XML conforme à la DTD du Tableau A.4 représente un objet **LibraryElement** comme décrit dans l'Annexe C de la CEI 61499-1. Les éléments racines possibles d'un tel document sont `FBType`, `AdapterType`, `ResourceType`, `DeviceType`, `System`, et `SubappType`, représentant respectivement les sous-classes concrètes **FBTypeDeclaration**, **AdapterTypeDeclaration**, **ResourceTypeDeclaration**, **DeviceTypeDeclaration**, **SystemConfiguration** et **SubapplicationTypeDeclaration** de la superclasse abstraite **LibraryElement**. La superclasse **DataTypeDeclaration** est représentée séparément par la DTD donnée à l'Article A.2.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.4 – DTD des éléments de bibliothèque (1 de 5)

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Common elements -->
<!ELEMENT Identification EMPTY>
<!ATTLIST Identification
 Standard CDATA #IMPLIED
 Classification CDATA #IMPLIED
 ApplicationDomain CDATA #IMPLIED
 Function CDATA #IMPLIED
 Type CDATA #IMPLIED
 Description CDATA #IMPLIED>
```

```
<!ELEMENT VersionInfo EMPTY>
<!ATTLIST VersionInfo
 Organization CDATA #REQUIRED
 Version CDATA #REQUIRED
 Author CDATA #REQUIRED
 Date CDATA #REQUIRED
 Remarks CDATA #IMPLIED>
```

```
<!ELEMENT CompilerInfo (Compiler*)>
<!ATTLIST CompilerInfo
 header CDATA #IMPLIED
 classdef CDATA #IMPLIED>

<!ELEMENT Compiler EMPTY>
<!ATTLIST Compiler
 Language (Java | Cpp | C | Other) #REQUIRED
 Vendor CDATA #REQUIRED
 Product CDATA #REQUIRED
 Version CDATA #REQUIRED>
```

```
<!ELEMENT FBNetwork
(FB*,EventConnections?,DataConnections?,AdapterConnections?)>

<!ELEMENT FB (Parameter*)>
<!ATTLIST FB
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
  x CDATA #IMPLIED
  y CDATA #IMPLIED>
```

```
<!ELEMENT EventConnections (Connection+)>

<!ELEMENT DataConnections (Connection+)>
<!ELEMENT AdapterConnections (Connection+)>
<!ELEMENT Connection EMPTY>
<!ATTLIST Connection
 Source CDATA #REQUIRED
 Destination CDATA #REQUIRED
 Comment CDATA #IMPLIED
 dx1 CDATA #IMPLIED
 dx2 CDATA #IMPLIED
 dy CDATA #IMPLIED
>
```

```
<!-- FBType elements -->
<!ELEMENT FBType
(Identification?,VersionInfo+,CompilerInfo?,InterfaceList,
 (BasicFB | FBNetwork)?, Service?) >
<!ATTLIST FBType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED
>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.4 (2 de 5)

```
<!ELEMENT InterfaceList
(EventInputs?,EventOutputs?,InputVars?,OutputVars?,
 Sockets?, Plugs?)>
<!ELEMENT EventInputs (Event+)>
<!ELEMENT EventOutputs (Event+)>
<!ELEMENT InputVars (VarDeclaration+)>
<!ELEMENT OutputVars (VarDeclaration+)>
<!ELEMENT Sockets (AdapterDeclaration+)>
<!ELEMENT Plugs (AdapterDeclaration+)>
```

```
<!ELEMENT Event (With*)>
<!ATTLIST Event
 Name CDATA #REQUIRED
 Type CDATA #IMPLIED
 Comment CDATA #IMPLIED>

<!ELEMENT With EMPTY>
<!ATTLIST With
 Var CDATA #REQUIRED>
```

```
<!ELEMENT VarDeclaration EMPTY>
<!ATTLIST VarDeclaration
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 ArraySize CDATA #IMPLIED
 InitialValue CDATA #IMPLIED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT AdapterDeclaration (Parameter*)>
<!ATTLIST AdapterDeclaration
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

```
<!ELEMENT BasicFB (InternalVars?,ECC?,Algorithm*)>

<!ELEMENT InternalVars (VarDeclaration+)>

<!ELEMENT ECC (ECState+,ECTransition+)>
```

```
<!ELEMENT ECState (ECAction*)>
<!ATTLIST ECState
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

```
<!ELEMENT ECTransition EMPTY>
<!ATTLIST ECTransition
 Source CDATA #REQUIRED
 Destination CDATA #REQUIRED
 Condition CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

```
<!ELEMENT ECAction EMPTY>
<!ATTLIST ECAction
 Algorithm CDATA #IMPLIED
 Output CDATA #IMPLIED>
```

```
<!ELEMENT Algorithm (VarDeclaration*, (FBD | ST | LD | Other))>
<!ATTLIST Algorithm
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT FBD (FB+,DataConnections)>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.4 (3 de 5)

```
<!ELEMENT ST (#PCDATA)>
<!ATTLIST ST
 Text CDATA #IMPLIED >
```

```
<!ELEMENT LD (Rung+)>
<!ELEMENT Rung (#PCDATA)>
<!ATTLIST Rung
 Output CDATA #REQUIRED
 Expression CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT Other (#PCDATA)>
<!ATTLIST Other
 Language CDATA #REQUIRED >
```

```
<!ELEMENT Service (ServiceSequence+)>
<!ATTLIST Service
 RightInterface CDATA #REQUIRED
 LeftInterface CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT ServiceSequence (ServiceTransaction*)>
<!ATTLIST ServiceSequence
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!ELEMENT ServiceTransaction (InputPrimitive?, OutputPrimitive*)>

<!ELEMENT InputPrimitive EMPTY>
<!ATTLIST InputPrimitive
 Interface CDATA #REQUIRED
 Event CDATA #REQUIRED
 Parameters CDATA #IMPLIED>

<!ELEMENT OutputPrimitive EMPTY>
<!ATTLIST OutputPrimitive
 Interface CDATA #REQUIRED
 Event CDATA #REQUIRED
 Parameters CDATA #IMPLIED>
```

```
<!-- AdapterType elements -->
<!ELEMENT AdapterType
(Identification?,VersionInfo+,CompilerInfo?,InterfaceList,Service?)>
<!ATTLIST AdapterType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>
```

```
<!-- ResourceType elements -->
<!ELEMENT ResourceType (Identification?,VersionInfo+, CompilerInfo?,
FBTypeName*, VarDeclaration*, FBNetwork?)>
<!ATTLIST ResourceType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT FBTypeName EMPTY>
<!ATTLIST FBTypeName
 Name CDATA #REQUIRED>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.4 (4 de 5)

```
<!-- DeviceType elements -->
<!ELEMENT DeviceType (Identification?,VersionInfo+, CompilerInfo?,
VarDeclaration*, ResourceTypeName*, Resource*, FBNetwork?)>
<!ATTLIST DeviceType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT ResourceTypeName EMPTY>
<!ATTLIST ResourceTypeName
 Name CDATA #REQUIRED >

<!ELEMENT Resource (Parameter*,FBNetwork?)>
<!ATTLIST Resource
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

```
<!-- System elements -->
<!ELEMENT System (Identification?, VersionInfo+, CompilerInfo?,
Application*, Device+, Mapping*, Segment*, Link*)>
<!ATTLIST System
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT Application (SubAppNetwork)>
<!ATTLIST Application
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT Mapping EMPTY>
<!ATTLIST Mapping
 From CDATA #REQUIRED
 To CDATA #REQUIRED>

<!ELEMENT Device (Parameter*,Resource*,FBNetwork?)>
<!ATTLIST Device
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
 x CDATA #IMPLIED
 y CDATA #IMPLIED>
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.4 (5 de 5)

```
<!-- SubAppType elements -->
<!ELEMENT SubAppType
(Identification?,VersionInfo+,CompilerInfo?,SubAppInterfaceList,
SubAppNetwork?)>
<!ATTLIST SubAppType
 Name CDATA #REQUIRED
 Comment CDATA #IMPLIED>

<!ELEMENT SubAppInterfaceList
(SubAppEventInputs?,SubAppEventOutputs?,InputVars?,OutputVars?)>
<!ELEMENT SubAppEventInputs (SubAppEvent+)>
<!ELEMENT SubAppEventOutputs (SubAppEvent+)>

<!ELEMENT SubAppEvent EMPTY>
<!ATTLIST SubAppEvent
 Name CDATA #REQUIRED
 Type CDATA #IMPLIED
 Comment CDATA #IMPLIED>

<!ELEMENT SubAppNetwork
(SubApp*,FB*,EventConnections?,DataConnections?, AdapterConnections?)>

<!ELEMENT SubApp EMPTY>
<!ATTLIST SubApp
 Name CDATA #REQUIRED
 Type CDATA #REQUIRED
 Comment CDATA #IMPLIED
  x CDATA #IMPLIED
  y CDATA #IMPLIED>
```

```
<!-- Network elements -->
<!ELEMENT SegmentType (Identification?,VersionInfo+, CompilerInfo?,
VarDeclaration*)>
<!ATTLIST SegmentType
  Name CDATA #REQUIRED
  Comment CDATA #IMPLIED >

<!ELEMENT Segment (Parameter*)>
<!ATTLIST Segment
  Name CDATA #REQUIRED
  Type CDATA #REQUIRED
  Comment CDATA #IMPLIED
  x CDATA #IMPLIED
  y CDATA #IMPLIED
  dx1 CDATA #IMPLIED>

<!ELEMENT Parameter EMPTY>
<!ATTLIST Parameter
  Name CDATA #REQUIRED
  Value CDATA #REQUIRED
  Comment CDATA #IMPLIED>

<!ELEMENT Link (Parameter*)>
<!ATTLIST Link
  SegmentName CDATA #REQUIRED
  CommResource CDATA #REQUIRED
  Comment CDATA #IMPLIED >
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Le Tableau A.5 donne des explications de certains des éléments de la DTD ci-dessus et (le cas échéant) les références à la syntaxe formelle pour leurs attributs.

Tableau A.5 – LibraryElement éléments de DTD (1 de 6)

Elément attributs	Syntaxe (CEI 61499-1, Annexe B)	Explication
Identification	Voir Tableau A.3	
VersionInfo		
CompilerInfo		
Compiler		
FBNetwork	Un *réseau de bloc fonctionnel* tel que défini dans la CEI 61499-1.	
FB	Une *instance de bloc fonctionnel* telle que définie dans la CEI 61499-1.	
Name	fb_instance_name	
Type	fb_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
x, y	Voir Annexe B.	
Connection	Une *connexion d'événement, connexion de données* ou *connexion d'adaptateur* telle que définie dans la CEI 61499-1.	
Source	c) [1]	
Destination	c)	
dx1, dx2, dy	Voir Annexe B.	
FBType	Un *type de bloc fonctionnel* tel que décrit dans la CEI 61499-1.	
Name	fb_type_name	
Comment	Un commentaire conforme à la CEI 61131 sans les délimiteurs (* et *)	
Event	Une *déclaration* d'une *interface d'événement*.	
Name	event_input_name | event_output_name	d)
Type	event_type	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
With	Une *déclaration* d'une association entre un *événement* et une *variable*.	
Var	input_variable_name | output_ variable _name	d)

1 Voir la note en bas de tableau à la fin de ce tableau.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.5 (2 de 6)

Elément attributs	Syntaxe (CEI 61499-1, Annexe B)	Explication
VarDeclaration	Une déclaration de variable	
Name	input_variable_name | output_variable_name | internal_variable_name	e) [1]
Type	identifier	
ArraySize	f)	
InitialValue	g)	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
AdapterDeclaration	Une déclaration d'une interface de prise mâle ou de prise femelle d'un type de bloc fonctionnel.	
Name	plug_name | socket_name	h)
Type	adapter_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
x, y	Emplacement (voir Annexe B) de prise mâle ou de prise femelle dans le réseau de bloc fonctionnel interne d'un *type de bloc fonctionnel composite*.	
ECState	Un *état EC* comme défini dans la CEI 61499-1.	
Name	ec_state_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
x, y	Voir Annexe B.	
ECTransition	Une *transition EC* comme définie dans la CEI 61499-1.	
Source	ec_state_name	
Destination	ec_state_name	
Condition	ec_transition_condition	
x, y	Voir Annexe B.	
ECAction	Une *action EC* comme défini dans la CEI 61499-1.	
Algorithm	algorithm_name	
Output	event_output_name	
Algorithm	Un *algorithme* dans un langage spécifié a)	
Name	algorithm_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
ST	Un *algorithme* dans le langage ST de la CEI 61131-3.	
Text	Contenu d'algorithme dans la syntaxe d'un statement_list selon la CEI 61131-3, Annexe B.	l)

[1] Voir la note en bas de tableau à la fin de ce tableau.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.5 (3 de 6)

Elément attributs	Syntaxe (CEI 61499-1, Annexe B)	Explication
Rung	Un échelon d'un algorithme en langage LD.	
Output	b) [1]	
Expression	Voir b) et l)	
Other	Un algorithme dans un langage autre que FBD, ST ou LD, .	
Language	Le nom du langage de programmation	a)
Text	Contenu de l'algorithme dans la syntaxe définie pour le langage spécifique	Voir a) et l)
Service	Une *déclaration* d'un *service* selon la CEI 61499-1	
RightInterface	service_interface_name	
LeftInterface	service_interface_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
ServiceSequence	Une déclaration d'une séquence de service selon la CEI 61499-1	
Name	sequence_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
InputPrimitive	Une *primitive de service* input selon la CEI 61499-1	
Interface	service_interface_name	
Event	(([plug_name '.'] event_input_name) | (socket_name '.' event_output_name)) ['+' | '-']	
Parameters	input_variable_name {',' input_variable_name}	
OutputPrimitive	Une *primitive de service* output selon la CEI 61499-1	
Interface	service_interface_name	
Event	('NULL' | ([plug_name '.'] event_output_name) | (socket_name '.' event_input_name)) ['+' | '-']	
Parameters	output_variable_name {',' output_variable_name}	
AdapterType	Une déclaration d'un type d'interface d'adaptateur selon la CEI 61499-1	
Name	adapter_type_name	
Comment	Un commentaire conforme à la CEI 61131 -3 sans les délimiteurs (* et *)	
ResourceType	Une *déclaration* d'un *type de ressource* selon la CEI 61499-1	
Name	resource_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	

1 Voir la note en bas de tableau à la fin de ce tableau.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.5 (4 de 6)

Elément attributs	Syntaxe (CEI 61499-1, Annexe B)	Explication
FBTypeName	Le nom d'un type de bloc fonctionnel pris en charge par toutes les instances d'un type de ressource	
Name	fb_type_name	
DeviceType	Une *déclaration* d'un *type d'équipement* selon la CEI 61499-1	
Name	device_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
ResourceTypeName	Le nom d'un *type de ressource* pris en charge par toutes les *instances* d'un *type d'équipement*	
Name	resource_type_name	
Resource	Une instance de ressource présente dans toutes les instances d'un type d'équipement	
Name	resource_instance_name	
Type	resource_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
Système	Une déclaration d'une configuration de système selon la CEI 61499-1	
Name	system_name	
Comment	Un commentaire conforme à la CEI 61131-3, 2.1.5 sans les délimiteurs (* et *)	
Application	Une *déclaration* d'une *application* selon la CEI 61499-1	
Name	application_name	
Comment	Un commentaire conforme à la CEI 61131 -3 sans les délimiteurs (* et *)	
Mapping	Mapping d'une *instance de bloc fonctionnel* issue d'une *application* avec une instance de bloc fonctionnel dans une *ressource*.	
From	fb_instance_reference	*Nom d'instance de bloc fonctionnel* hiérarchique dans son *application*. Par exemple: APP1.SUBAPP2.FB2.
To	fb_resource_reference	*Nom d'instance de bloc fonctionnel* hiérarchique dans le système physique (voir j)) [1].
1 Voir la note en bas de tableau à la fin de ce tableau.		

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.5 (5 de 6)

Elément attributs	Syntaxe (CEI 61499-1, Annexe B)	Explication
Device	Une déclaration d'une configuration d'équipement selon la CEI 61499-1	
Name	device_instance_name	
Type	device_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
SubAppType	Une déclaration d'un type de sous-application selon la CEI 61499-1	
Name	subapp_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
SubAppEvent	Une déclaration d'une interface d'événement d'un type sous-application.	
Name	event_input_name | event_output_name	i) [1]
Type	event_type	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
SubApp	Une *instance de sous-application* comme défini dans la CEI 61499-1.	
Name	subapp_instance_name	
Type	subapp_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
x, y	Voir Annexe B.	
SegmentType	Une *déclaration* d'un *type segment* selon la CEI 61499-1	
Name	segment_type_name	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
Segment	Un segment d'un *réseau* de communication	
Name	identifier	
Type	identifier	
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
x, y, dx1	Voir Annexe B.	

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Tableau A.5 (6 de 6)

Elément attributs	Syntaxe (CEI 61499-1, Annexe B)	Explication
Link	Une liaison entre un élément Segment et un élément Device	
CommResource	resource_hierarchy	k)
SegmentName	identifier	Le segment devant être lié
Comment	Un commentaire conforme à la CEI 61131-3 sans les délimiteurs (* et *)	
Parameter	Un *paramètre* d'un élément. Par exemple: un élément Segment ou Link	
Name	identifier	
Value	Une chaîne de caractères en un format approprié pour exprimer la valeur du paramètre associé.	
Comment	Un commentaire conforme à la CEI 61131 -3 sans les délimiteurs (* et *)	

a) La spécification d'algorithmes en des langages autres que FBD, ST et LD ne relève pas du domaine d'application de la présente Partie de la CEI 61499.

b) Sachant que les langages FBD et ST sont disponibles pour la spécification d'algorithmes complexes, il est recommandé que l'usage du langage LD dans le contexte de la présente norme soit limité aux échelons accomplissant l'évaluation des énoncés d'affectation de la forme <output>:= <expression>. Pour la portabilité entre outils logiciels, il est en outre recommandé que l'élément XML Expression ait la simple syntaxe textuelle d'opérateur postfixé suivante avec des termes séparés par des espaces (whitespace_separated):

```
expression::= and_expression
and_expression::= (variable_name ['!']) | or_expression
      | and_expression and_expression '&'
or_expression::= and_expression | or_expression or_expression '|'
```

Voir EXEMPLE 1 à l'Article C.1pour une illustration de cet usage recommandé.

c) Selon le contexte, il convient qu'un élément Source ou Destination corresponde à la syntaxe de l'élément respectif dans l'une des productions event_conn, data_conn, adapter_conn, subapp_event_conn, subapp_data_conn, config_event_conn, config_data_conn, config_adapter_conn, devtype_event_conn, devtype_data_conn, ou devtype_adapter_conn données dans l'Annexe B de la CEI 61499-1.

d) Les productions event_input_name et input_variable_name s'appliquent lorsque l'élément Event est partie intégrante d'un élément EventInputs, alors que event_output_name et output_variable_name s'appliquent lorsqu'il est partie intégrante d'un élément EventOutputs.

e) Les productions input_variable_name, output_variable_name et internal_variable_name s'appliquent lorsque l'élément VarDeclaration associé est respectivement partie intégrante d'un élément InputVars, OutputVars ou InternalVars.

f) La syntaxe de cet élément lorsqu'il est présent doit être conforme à l'expression syntaxique (subrange {',' subrange}) | integer {',' integer} où les non-terminaux subrange et integer sont tels que définis dans l'Annexe B de la CEI 61131-3. Chaque terme de la seconde forme équivaut à la sous-plage 0..n-1, où n est la valeur de l'élément syntaxique integer correspondant. Si l'élément est absent, la variable n'est pas une matrice de type anonyme, bien qu'elle puisse encore être une instance d'un type array (matrice) défini antérieurement.

g) La syntaxe de cet élément est la syntaxe pour l'initialisation du type de variable correspondant tel que défini dans l'Annexe B de la CEI 61131-3.

h) Les productions plug_name et socket_name s'appliquent lorsque l'élément AdapterDeclaration associé est respectivement partie intégrante d'un élément Plugs ou Sockets.

i) Les productions event_input_name et event_output_name s'appliquent lorsque l'élément

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

SubAppEvent est respectivement partie intégrante d'un élément SubAppEventInputs ou SubAppEventOutputs.

j) Cet élément peut montrer une hiérarchie complète nom d'équipement/ressource/FB (par exemple: DEV1.RES2.FB2); une hiérarchie équipement/ressource (par exemple: DEV1.RES2); ou (dans le cas où l'équipement lui-même est une seule ressource) simplement un nom d'équipement (par exemple: DEV1). Dans les deux derniers cas, l'instance FB doit avoir le même nom que dans l'application. Par exemple: si la source est APP1.FB3, le mapping résultant doit être faite respectivement avec DEV1.RES2.FB3 et DEV1.FB3.

k) Cet attribut fait référence à la ressource de communication reliée au segment de réseau. Il peut montrer une séquence complète de nom équipement/ressource (par exemple: DEV1.RES2); ou (dans le cas où l'équipement lui-même fournirait l'interface de communication) simplement un nom d'équipement (par exemple: DEV1).

l) Afin d'augmenter la lisibilité et de réduire la probabilité d'erreurs au cours du codage et de l'analyse syntaxique des données, il est recommandé, pour ces éléments, d'utiliser des sections CDATA à la place des attributs textuels.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Annexe B
(informative)

Modèle de graphique

B.1 Système de coordonnées

La présente Annexe présente un modèle graphique simple qui permet la reconstitution approchée de l'aspect graphique des *réseaux de communication, réseaux de blocs fonctionnels* et des *graphiques de contrôle d'exécution* (les ECC) entre outils logiciels, en utilisant les données définies pour les éléments **FBNetwork, ECC, ResourceType, DeviceType, SubAppType** et **System** dans la DTD **LibraryElement** de l'Annexe A.

NOTE Le modèle graphique de la présente Annexe vise à permettre la transmission d'informations parmi des outils logiciels avec une sémantique commune, mais l'exacte reproduction des graphiques peut ne pas être obtenue avec des outils ayant des algorithmes de présentation et de dessin différents.

Comme le sens principal du flot d'événements et de données dans le réseau de blocs fonctionnels est de gauche à droite et, en second lieu, de haut en bas, le système de coordonnées du modèle graphique a son origine (x=y=0) au coin supérieur gauche, les valeurs de la coordonnée x augmentant de gauche à droite tandis que les valeurs de la coordonnée y augmentent de haut en bas (voir Figure B.1).

Afin d'assurer une 'indépendance vis-à-vis des tailles de police et les résolutions d'affichage, les coordonnées x et y sont toutes deux exprimées en multiples de 1 % de la hauteur de ligne (désignée par **h** à la Figure B.1) utilisée pour la présentation des entrées et sorties de bloc fonctionnel.

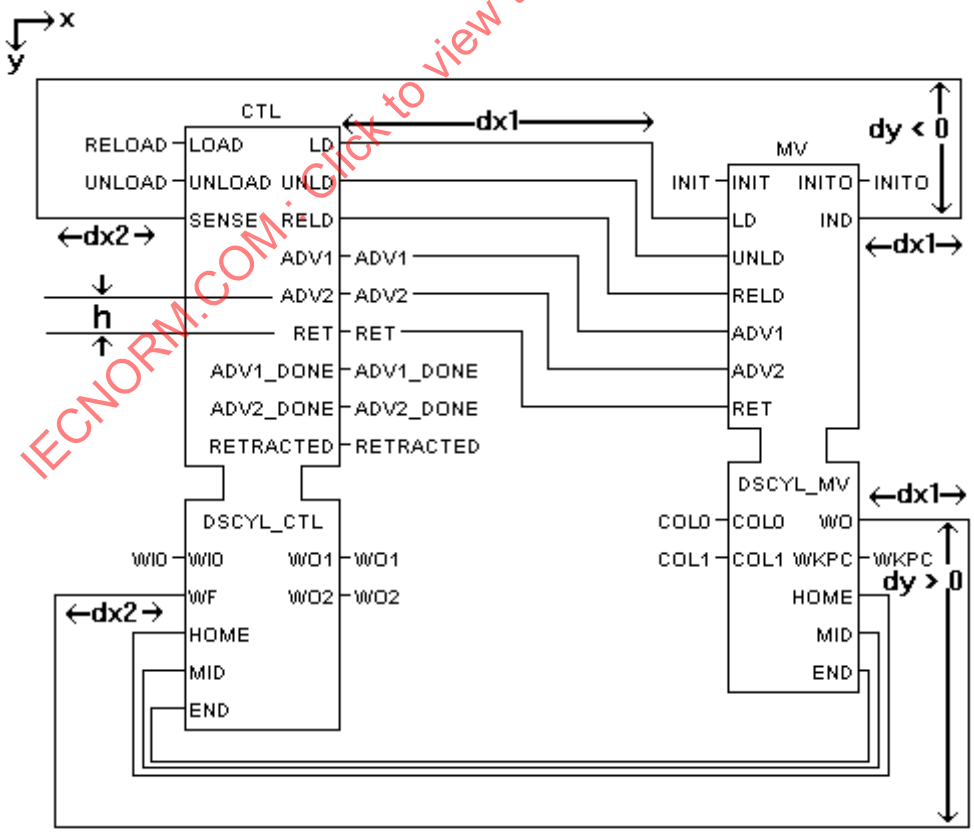


Figure B.1 – Modèle de graphique

EXEMPLE Le coin supérieur gauche de l'instance DSCYL_CTL nommé CTL à la Figure B.1 est situé approximativement à 10h unités du bord gauche du schéma et à 5h unités du haut du schéma; par conséquent, les

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

valeurs des attributs x et y du sous-élément FB correspondant dans un élément FBNetwork d'un document XML défini conformément à l'une des DTD énumérées ci-dessus seraient respectivement 1 000 et 500.

B.2 Emplacement des éléments graphiques

L'emplacement d'une *instance de bloc fonctionnel* est déterminé par l'emplacement du coin supérieur gauche de son contour graphique.

L'emplacement d'un *état EC* est déterminé par le point central du cadre de délimitation contenant le nom d'état.

L'emplacement d'une *condition de transition EC* est déterminé par le point central du cadre de délimitation invisible contenant la condition de transition.

NOTE Les cordonnées `(x,y)` peuvent être utilisées dans la syntaxe de transfert des *instances* équipement et *ressource*. Cependant, les outils logiciels qui utilisent par défaut les notations graphiques ou arborescentes ne sont pas tenus d'utiliser ou de produire ces attributs.

B.3 Acheminement des connexions

Comme illustré à la Figure B.1, les *connexions de données*, *connexions d'événements* et *connexions d'adaptateurs* peuvent être dessinées comme un nombre impair de segments de ligne selon les lignes directrices suivantes:

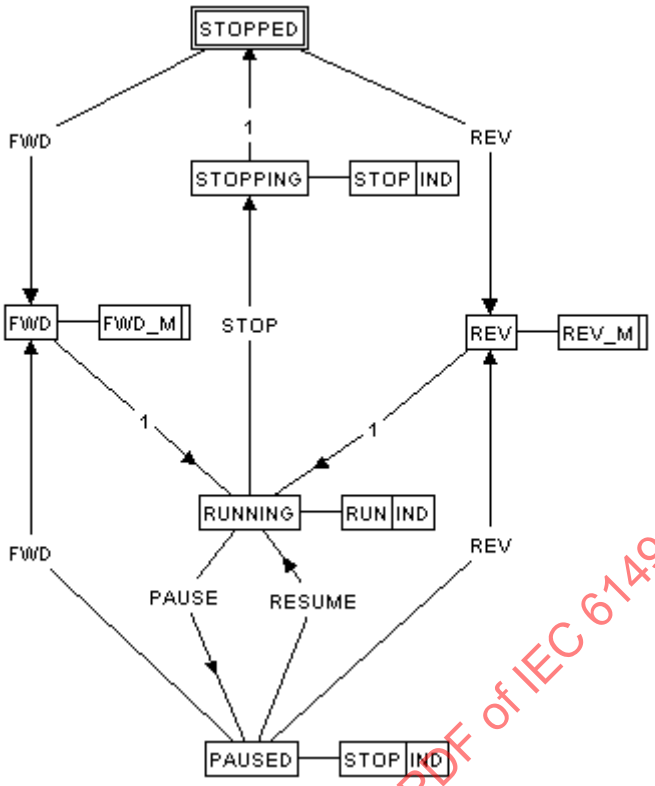
a) Lorsque la source de la connexion se situe à gauche de sa destination, la ligne peut être tracée sous la forme d'une seule ligne droite progressant du bord droit du bloc fonctionnel qui fournit la source de la connexion jusqu'au bord gauche du bloc fonctionnel qui fournit la destination de la connexion.

b) Lorsque la source de la connexion se situe à gauche de sa destination, la connexion peut être tracée sous la forme de trois segments de droite contigus progressant vers la droite sur une distance **dx1** du bord droit du bloc fonctionnel qui fournit la source de la connexion; ensuite verticalement sur une distance appropriée pour progresser horizontalement vers le bord gauche du bloc fonctionnel qui fournit la destination de la connexion.

c) Lorsque la source de la connexion se situe à gauche ou à droite de sa destination, la connexion peut être tracée sous la forme de cinq segments de droite contigus progressant vers la droite sur une distance **dx1** du bord droit du bloc fonctionnel qui fournit la source de la connexion; ensuite verticalement sur une distance **dy**; ensuite horizontalement jusqu'à une coordonnée x à une distance **dx2** à gauche du bord gauche du bloc fonctionnel qui fournit la destination de la connexion; ensuite verticalement sur une distance appropriée pour progresser horizontalement vers le bord gauche du bloc fonctionnel qui fournit la destination de la connexion.

d) Les *transitions EC* sont tracées sous la forme de deux lignes droites allant de l'emplacement de l'*action EC* source jusqu'à l'emplacement de la condition de transition EC et ensuite jusqu'à l'emplacement de l'action EC de destination, les emplacements étant comme défini à l'Article B.2. Les portions de ces lignes dans les limites des cadres de délimitation des noms d'état EC et des conditions de transition sont cachées (voir Figure B.2). Il est permis d'utiliser des pointes de flèches ou autres moyens pour indiquer le sens des transitions.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012



Anglais	Français
Stopped	Arrêté
Fwd	Direct
Rev	Inverse
Stopping	En arrêt
Stop	Arrêter
Running	En marche
Run	Faire marcher
Pause	Suspendre
Resume	Recommencer
Paused	Suspendu

Figure B.2 – Exemple de tracé de graphiques ECC

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

B.4 Présentations par défaut

Les fournisseurs d'outils logiciels doivent spécifier les moyens employés pour obtenir les présentations par défaut des *ECC* et des *réseaux de blocs fonctionnels* lorsque les informations graphiques nécessaires ne sont pas données dans le document de transfert XML de l'*élément bibliothèque* associé ou lorsque les informations fournies dans le document sont incompatibles avec les algorithmes de dessin utilisés par l'outil spécifique.

NOTE Un exemple d'une telle incompatibilité est lorsqu'un outil logiciel utilise un algorithme différent pour déterminer la largeur et la hauteur par rapport à l'algorithme employé par l'outil produisant le document, ce qui peut induire des perturbations parmi les éléments comme le chevauchement de lignes de connexion avec des contours d'éléments.

B.5 Représentation graphique des configurations systèmes

Une *configuration système* comme défini dans la CEI 61499-1, dont la représentation XML est donnée dans la notation de l'élément `System` définie dans les Tableaux A.4 et A.5, peut être présentée dans le système de coordonnées défini à l'Article B.1, selon les règles suivantes:

a) Un *équipement* est représenté sous la forme d'un bloc généralement rectangulaire contenant le nom de type de l'équipement, avec son nom d'instance placé en haut du bloc. Les coordonnées (x,y) du centre du bloc sont données par les attributs `x` et `y` de l'élément XML `Device`.

b) Un *segment de réseau* est représenté sous la forme d'un segment de ligne horizontale, qui peut être suffisamment épaisse verticalement pour contenir des informations textuelles telles que le nom d'instance et le nom de type du segment et peut avoir d'autres caractéristiques dépendant de la mise en œuvre telles que des pointes de flèches. L'origine et la longueur du segment de ligne d'une part et l'emplacement vertical du centre de la ligne d'autre part sont respectivement donnés par les paramètres `x`, `dx1` et `y` de l'élément XML `Segment`.

c) Une *liaison de réseau* est représentée sous la forme d'un segment de ligne vertical allant du centre d'un équipement (ou autre position dépendant de la mise en œuvre) jusqu'à l'axe central horizontal de la liaison de réseau correspondante. Des portions de la liaison peuvent être recouvertes par les représentations graphiques de segments et d'équipements.

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012

Annexe C
(informative)

Exemples

C.1 Types de bloc fonctionnel de base

EXEMPLE 1 Un type de bloc fonctionnel de base contenant un algorithme Ladder Diagram (LD «diagramme à contacts») selon la NOTE 2 du Tableau A.5 pourrait être exprimé sous la forme textuelle suivante.

```
FUNCTION_BLOCK LD_TEST (* Exemple d'algorithme en LD *)
EVENT_INPUT
    REQ WITH A, B, C, D, E, F, G, H;
END_EVENT
EVENT_OUTPUT
    CNF WITH Q1, Q2; (* Confirmation d'exécution *)
END_EVENT
VAR_INPUT
    A: BOOL;
    B: BOOL;
    C: BOOL;
    D: BOOL;
    E: BOOL;
    F: BOOL;
    G: BOOL;
    H: BOOL;
END_VAR
VAR_OUTPUT
    Q1: BOOL;
    Q2: BOOL;
END_VAR
EC_STATES
    START; (* État initial *)
    REQ: REQ -> CNF; (* Exécution normale *)
END_STATES
EC_TRANSITIONS
    START TO REQ:= REQ;
    REQ TO START:= 1;
END_TRANSITIONS
ALGORITHM REQ IN LD:
    Q1:= ((!À|B)&(C|D))|((E|F)&(G|!H));
    Q2:= ((A&B)|(C&D))&((E&F)|(G&H));
END_ALGORITHM
END_FUNCTION_BLOCK
```

IECNORM.COM : Click to view the full PDF of IEC 61499-2:2012