

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Engineering data exchange format for use in industrial automation systems
engineering – Automation markup language –
Part 3: Geometry and kinematics**

**Format d'échange de données techniques pour une utilisation dans l'ingénierie
des systèmes d'automatisation industrielle – Automation markup language –
Partie 3: Géométrie et cinématique**



THIS PUBLICATION IS COPYRIGHT PROTECTED

Copyright © 2017 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 l'IEC ou du Comité national de l'IEC du pays du demandeur. Si vous avez des questions sur le copyright de l'IEC ou si vous désirez obtenir des droits supplémentaires sur cette publication, utilisez les coordonnées ci-après ou contactez le Comité national de l'IEC de votre pays de résidence.

IEC Central Office
3, rue de Varembe
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.

IEC Catalogue - webstore.iec.ch/catalogue

The stand-alone application for consulting the entire bibliographical information on IEC International Standards, Technical Specifications, Technical Reports and other documents. Available for PC, Mac OS, Android Tablets and iPad.

IEC publications search - www.iec.ch/searchpub

The advanced search enables 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 online and also once a month by email.

Electropedia - www.electropedia.org

The world's leading online dictionary of electronic and electrical terms containing 20 000 terms and definitions in English and French, with equivalent terms in 16 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

IEC Glossary - std.iec.ch/glossary

65 000 electrotechnical terminology entries in English and French extracted from the Terms and Definitions clause of IEC publications issued since 2002. Some entries have been collected from earlier publications of IEC TC 37, 77, 86 and CISPR.

IEC 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 l'IEC

La Commission Electrotechnique Internationale (IEC) 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 IEC

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

Catalogue IEC - webstore.iec.ch/catalogue

Application autonome pour consulter tous les renseignements bibliographiques sur les Normes internationales, Spécifications techniques, Rapports techniques et autres documents de l'IEC. Disponible pour PC, Mac OS, tablettes Android et iPad.

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

La recherche avancée permet de trouver des publications IEC 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.

IEC Just Published - webstore.iec.ch/justpublished

Restez informé sur les nouvelles publications IEC. 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 de termes électroniques et électriques. Il contient 20 000 termes et définitions en anglais et en français, ainsi que les termes équivalents dans 16 langues additionnelles. Egalement appelé Vocabulaire Electrotechnique International (IEV) en ligne.

Glossaire IEC - std.iec.ch/glossary

65 000 entrées terminologiques électrotechniques, en anglais et en français, extraites des articles Termes et Définitions des publications IEC parues depuis 2002. Plus certaines entrées antérieures extraites des publications des CE 37, 77, 86 et CISPR de l'IEC.

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.

INTERNATIONAL STANDARD

NORME INTERNATIONALE



**Engineering data exchange format for use in industrial automation systems
engineering – Automation markup language –
Part 3: Geometry and kinematics**

**Format d'échange de données techniques pour une utilisation dans l'ingénierie
des systèmes d'automatisation industrielle – Automation markup language –
Partie 3: Géométrie et cinématique**

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

COMMISSION
ELECTROTECHNIQUE
INTERNATIONALE

ICS 01.040.01; 25.040.01; 35.240.30

ISBN 978-2-8322-3794-6

**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.**

CONTENTS

FOREWORD.....	6
INTRODUCTION.....	8
1 Scope.....	10
2 Normative references	10
3 Terms, definitions and abbreviations	10
3.1 Terms and definitions.....	10
3.2 Abbreviations	11
4 Conformity.....	11
5 Extensions of AML libraries for geometry and kinematics.....	11
5.1 General.....	11
5.2 AutomationMLBaseRoleClassLib – RoleClass Frame	11
5.3 AutomationMLInterfaceClassLib	11
5.3.1 InterfaceClass COLLADAInterface	11
5.3.2 InterfaceClass AttachmentInterface	12
6 Frame attribute	12
7 Integration of COLLADA documents	13
8 Attachment of two AML objects.....	14
9 Meta information about the COLLADA source tool.....	15
Annex A (informative) Referencing methods for geometric/kinematic descriptions.....	17
A.1 Integration of a common COLLADA document with explicit referencing	17
A.1.1 General	17
A.1.2 Definition of the Frame attribute.....	18
A.1.3 Structure of the COLLADA documents	20
A.1.4 Referencing using URI and fragments without a target and ID	23
A.1.5 Referencing using URI and fragments including a target without an ID	23
A.1.6 Referencing using URI without a fragment, including a target and an ID	24
A.1.7 Referencing using URI and fragments including a target and an ID.....	25
A.1.8 Referencing using URI without a fragment, target and ID.....	26
A.2 Implicit referencing of COLLADA elements.....	27
A.2.1 General	27
A.2.2 Implicit referencing	27
A.2.3 Implicit referencing to COLLADA subdocuments	29
A.2.4 Publishing elements of a COLLADA document in CAEX.....	33
A.3 Attachment between objects in CAEX	35
Annex B (informative) Modelling of kinematic systems and their combination in AML	41
B.1 General.....	41
B.2 Modelling an AML document of a linear unit in CAEX and COLLADA	41
B.2.1 General	41
B.2.2 Definition of the visual scene	41
B.2.3 Definition of the joint.....	43
B.2.4 Definition of the kinematic model	43
B.2.5 Definition of the articulated system	43
B.2.6 Definition of the kinematic scene	45
B.2.7 Assembling of the scene.....	45
B.2.8 Combination of CAEX and COLLADA into AML.....	46

B.3	Modelling an AML document of a robot in CAEX and COLLADA.....	47
B.3.1	General	47
B.3.2	Definition of the visual scene	48
B.3.3	Definition of joints.....	50
B.3.4	Definition of the kinematic model	51
B.3.5	Definition of the articulated system	51
B.3.6	Definition of the kinematic scene	54
B.3.7	Assembling of the scene	55
B.3.8	Combination of CAEX and COLLADA into AML.....	56
B.4	Modelling an AML document of a combined system including a robot and a linear axis in CAEX and COLLADA	58
B.5	Modelling an AML document of a gripper connected to robot in CAEX and COLLADA	61
B.5.1	General	61
B.5.2	Definition of the visual scene	62
B.5.3	Definition of the kinematic system.....	63
B.5.4	Assembling of the scene	71
B.5.5	Combination of CAEX and COLLADA into AML.....	72
B.6	Modelling an AML document of a work piece connected to a gripper in CAEX and COLLADA	75
B.6.1	General	75
B.6.2	Implicit upper boundary	75
B.6.3	Definition of the work piece.....	77
B.6.4	Combination of CAEX and COLLADA into AML.....	78
Annex C (informative)	XML representation of AML libraries	82
C.1	AutomationMLBaseRoleClassLib	82
C.2	AutomationMLInterfaceClassLib.....	82
Figure 1	– Overview of the engineering data exchange format AML	8
Figure 2	– Required XML text in case of ISO/PAS 17506	16
Figure 3	– Required XML text in case of COLLADA 1.4.1.....	16
Figure A.1	– Decision tree for different referencing methods	17
Figure A.2	– Two frames represented in the InstanceHierarchy of an AML document.....	18
Figure A.3	– XML representation of the AML document.....	18
Figure A.4	– Translation and spatially fixed rotation	19
Figure A.5	– COLLADA scene used in this example	20
Figure A.6	– Structure and References	20
Figure A.7	– Content of the COLLADA document cube.dae.....	21
Figure A.8	– Content of the COLLADA document red_blue_cubes.dae.....	22
Figure A.9	– “RedCube” – Hierarchy of the AML document	23
Figure A.10	– XML representation of the AML document.....	23
Figure A.11	– Referencing the red cube by ID.....	23
Figure A.12	– “BlueCube” – Hierarchy of the AML document.....	24
Figure A.13	– XML representation of the AML document.....	24
Figure A.14	– Referencing the blue cube	24
Figure A.15	– Hierarchy of the AML document	24
Figure A.16	– XML representation of the AML document.....	25

Figure A.17 – Referencing the blue cube starting from the element “subpart”	25
Figure A.18 – Hierarchy of the AML document	25
Figure A.19 – XML representation of the AML document	25
Figure A.20 – Referencing the blue cube	26
Figure A.21 – Hierarchy of the AML document	26
Figure A.22 – XML representation of the AML document	26
Figure A.23 – Referencing the complete COLLADA scene	27
Figure A.24 – Implicit Referencing: Hierarchy of the AML document	28
Figure A.25 – XML representation of the AML document	28
Figure A.26 – Structure and relations of referenced COLLADA subdocuments	29
Figure A.27 – Content of the modified COLLADA document red_blue_cubes.dae	30
Figure A.28 – Content of the COLLADA document red_cube.dae	30
Figure A.29 – Content of the COLLADA document blue_cube.dae	31
Figure A.31 – XML representation of the AML document	32
Figure A.33 – Additional frame element in COLLADA document	33
Figure A.34 – Publishing frames: Hierarchy of the AML document	34
Figure A.35 – XML representation of the AML document	35
Figure A.36 – Structure for attachments between objects in CAEX	36
Figure A.37 – Visualization of yellow cube with additional frame	36
Figure A.38 – COLLADA document of yellow cube with additional frame	37
Figure A.39 – Hierarchy of the AML document	38
Figure A.40 – XML representation of the AML document	39
Figure A.41 – Attachment between geometric AML objects	40
Figure A.42 – XML representation of the AML document	40
Figure B.1 – Visualization of the linear unit	41
Figure B.2 – Definition of the visual scene	42
Figure B.3 – Definition of the joint	43
Figure B.4 – Definition of kinematic model	43
Figure B.5 – Definition of the articulated system library	44
Figure B.6 – Definition of the kinematic articulated system	44
Figure B.7 – Definition of the motion articulated system	45
Figure B.8 – Definition of the kinematic scene	45
Figure B.9 – Instantiation of the kinematic scene	46
Figure B.10 – Hierarchy of the AML document	46
Figure B.11 – XML representation of the AML document	47
Figure B.13 – Definition of the visual scene	50
Figure B.14 – Definition of joints	50
Figure B.15 – Definition of kinematic model	51
Figure B.16 – Definition of the articulated system library	51
Figure B.17 – Definition of the kinematic articulated system	53
Figure B.18 – Definition of the motion articulated system	54
Figure B.19 – Definition of the kinematic scene	55
Figure B.20 – Instantiation of the kinematic scene	56

Figure B.21 – Hierarchy of the AML document	57
Figure B.22 – XML representation of the AML document.....	57
Figure B.24 – Hierarchy of the AML document	59
Figure B.25 – XML representation of the AML document.....	60
Figure B.26 – XML representation of the AML document.....	60
Figure B.27 – Visualization of the robot attached to the linear unit	61
Figure B.30 – Definition of the visual scene	63
Figure B.31 – Definition of the kinematics	64
Figure B.32 – Definition of joints	64
Figure B.33 – Definition of kinematic model	65
Figure B.34 – Definition of the articulated system	66
Figure B.35 – Definition of the articulated system	67
Figure B.36 – Definition of the kinematic scene.....	68
Figure B.37 – Definition of the joint dependency using MathML	68
Figure B.38 – XML representation of the COLLADA document gripper_kinematics.dae	71
Figure B.39 – XML representation of the COLLADA document gripper.dae	72
Figure B.40 – Hierarchy of the AML document	73
Figure B.41 – XML representation of the AML document.....	74
Figure B.42 – XML representation of the AML document.....	75
Figure B.43 – Visualization of the robot on a linear unit and attached gripper	75
Figure B.44 – Example for implicit upper boundary	76
Figure B.45 – Structure for attachments between objects in CAEX.....	76
Figure B.46 – Visualization of the work piece with additional frame.....	77
Figure B.48 – Hierarchy of the AML document	79
Figure B.49 – XML representation of the AML document.....	81
Figure B.50 – Attachment between geometric AML objects	81
Figure B.51 – XML representation of the AML document.....	81
Figure C.1 – XML representation of AML libraries AutomationMLBaseRoleClassLib.....	82
Figure C.2 – XML representation of AML libraries AutomationMLInterfaceClassLib	83
Table 1 – Abbreviations	11
Table 2 – RoleClass Frame.....	11
Table 3 – InterfaceClass COLLADAInterface	12
Table 4 – InterfaceClass AttachmentInterface	12
Table 5 – Attribute “Frame”	13
Table 6 – Sub-attributes of the attribute “Frame”	13
Table 7 – Rules for resolving document and entry point	14
Table 8 – Meta information about the COLLADA source tool	16

INTERNATIONAL ELECTROTECHNICAL COMMISSION

ENGINEERING DATA EXCHANGE FORMAT FOR USE IN INDUSTRIAL AUTOMATION SYSTEMS ENGINEERING – AUTOMATION MARKUP LANGUAGE –

Part 3: Geometry and kinematics

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 62714-3 has been prepared by subcommittee 65E: Devices and integration in enterprise systems, of IEC technical committee 65: Industrial-process measurement, control and automation.

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

CDV	Report on voting
65E/497/CDV	65E/508/RVC

Full information on the voting for the approval of this International 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 in the IEC 62714 series, published under the general title *Engineering data exchange format for use in industrial automation systems engineering – Automation markup language*, can be found on the IEC website.

The committee has decided that the contents of this document will remain unchanged until the stability date indicated on the IEC website under "<http://webstore.iec.ch>" in the data related to the specific document. At this date, the document 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 62714-3:2017

INTRODUCTION

The data exchange format defined in IEC 62714 (Automation Markup Language, AML) is an XML schema based data format and has been developed in order to support the data exchange between engineering tools in a heterogeneous engineering tool landscape. IEC 62714-1 gives an overview about the format.

The goal of AML is to interconnect engineering tools from the existing heterogeneous tool landscape in their different disciplines, e.g. mechanical plant engineering, electrical design, process engineering, process control engineering, HMI development, PLC programming, robot programming etc.

AML stores engineering information following the object oriented paradigm and allows modelling of physical and logical plant components as data objects encapsulating different aspects. An object may consist of other sub-objects and may itself be part of a larger composition or aggregation. Typical objects in plant automation comprise information on topology, geometry, kinematics and logic, whereas logic comprises sequencing, behaviour and control.

AML combines existing industry data formats that are designed for the storage and exchange of different aspects of engineering information. These data formats are used on “as-is” basis within their own specifications and are not branched for AML needs.

The core of AML is the top-level data format CAEX that connects the different data formats. Therefore, AML has an inherent distributed document architecture.

Figure 1 illustrates the basic AML architecture and the distribution of topology, geometry, kinematic and logic information.

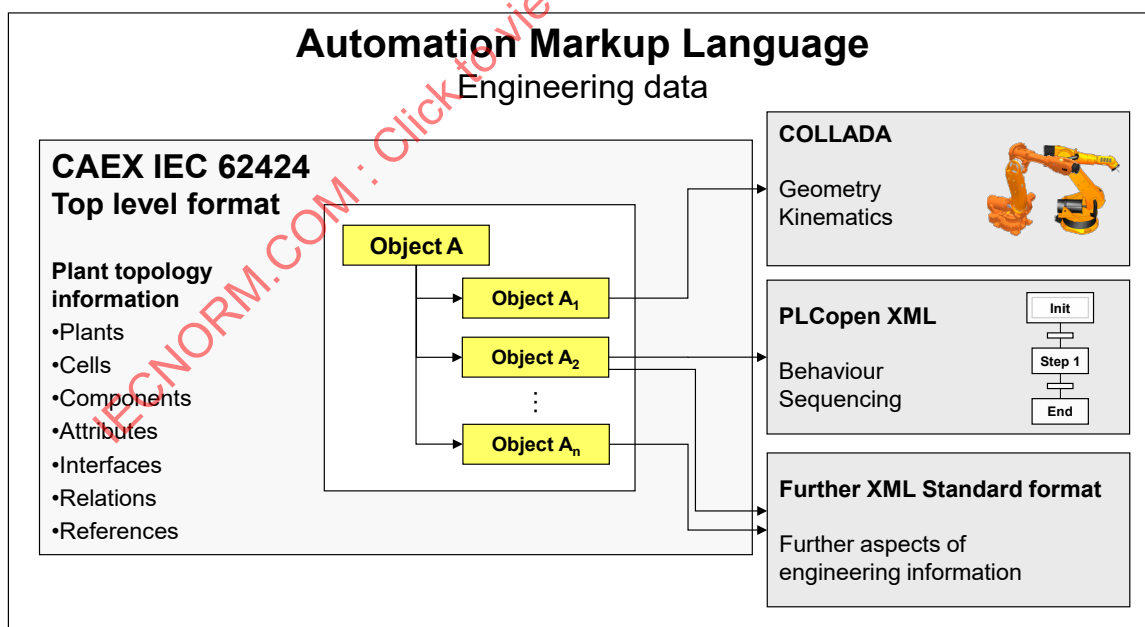


Figure 1 – Overview of the engineering data exchange format AML

Due to the different aspects of AML, IEC 62714 consists of different parts focussing on different aspects.

- IEC 62714-1: Architecture and general requirements

This part specifies the general AML architecture, the modelling of engineering data, classes, instances, relations, references, hierarchies, basic AML libraries and extended AML concepts.

- IEC 62714-2: Role class libraries

This part specifies additional AML libraries.

- IEC 62714-3: Geometry and kinematics

This part specifies the modelling of geometry and kinematics information.

Further parts may be added in the future in order to interconnect further data standards to AML.

Clause 5 describes the geometry related extensions of the role class libraries.

Clause 6 describes the frame attribute which can be used to represent the geometric position of an InternalElement, InstanceHierarchy, SystemUnitClass, or SystemUnitClassLibrary with respect to another CAEX Object.

Clause 7 gives a normative description regarding referencing COLLADA documents.

Clause 8 specifies the normative provisions for the attachment of two geometric AML objects.

Clause 9 defines how to store meta informations about the source tool directly into the COLLADA document.

Annex A describes the referencing methods for geometric and kinematic models.

Annex B provides an example for modelling of kinematic systems and their combination in AML.

Annex C gives an informative XML representation of the libraries defined in this part of IEC 62714.

ENGINEERING DATA EXCHANGE FORMAT FOR USE IN INDUSTRIAL AUTOMATION SYSTEMS ENGINEERING – AUTOMATION MARKUP LANGUAGE –

Part 3: Geometry and kinematics

1 Scope

This part of IEC 62714 specifies the integration of geometry and kinematics information for the exchange between engineering tools in the plant automation area by means of AML.

It does not define details of the data exchange procedure or implementation requirements for the import/export tools.

2 Normative references

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

IEC 62714-1:2014, *Engineering data exchange format for use in industrial automation systems engineering – Automation markup language – Part 1: Architecture and general requirements*

IEC 62714-2:2015, *Engineering data exchange format for use in industrial automation systems engineering – Automation markup language – Part 2: Role class libraries*

ISO/PAS 17506, *Industrial automation systems and integration – COLLADA digital asset schema specification for 3D visualization of industrial data*

COLLADA 1.4.1: March 2008 COLLADA – Digital Asset Schema Release 1.4.1
(available at <http://www.khronos.org/files/collada_spec_1_4.pdf>)

Extensible Markup Language (XML) 1.0:2004, W3C Recommendation
(available at <<http://www.w3.org/TR/2004/REC-xml-20040204/>>)

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in IEC 62714-1:2014 and of IEC 62714-2:2015 and the following apply.

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

- IEC Electropedia: available at <http://www.electropedia.org/> [access on 26th September 2016]
- ISO Online browsing platform: available at <http://www.iso.org/obp> [access on 26th September 2016]

3.1.1

SID path

reference of type sidref_type to an arbitrary element within a COLLADA document according to ISO/PAS 17506 or COLLADA 1.4.1

3.2 Abbreviations

For the purposes of this document, the abbreviations of IEC 62714-1:2014 and of IEC 62714-2:2015 apply as well as the abbreviations listed in Table 1.

Table 1 – Abbreviations

SCARA	Selective Compliance Assembly Robot Arm
SID	Scoped Identifier

4 Conformity

To claim conformity to the present document with respect to the support of AML, the requirements of Clauses 5, 6, 7, 8 and 9 shall be fulfilled. In the scope of AML, a COLLADA document shall conform to the specification of ISO/PAS 17506 or COLLADA 1.4.1.

5 Extensions of AML libraries for geometry and kinematics

5.1 General

Clause 5 defines extensions of the standard AML RoleClasses and standard AML InterfaceClasses. These classes are part of the AML standard libraries and a specific extension of IEC 62714-1 for this part of IEC 62714. All described attributes are part of the standard libraries and may be removed in the InstanceHierarchy if not needed.

5.2 AutomationMLBaseRoleClassLib – RoleClass Frame

The RoleClass “Frame” shall be used as specified in Table 2.

Table 2 – RoleClass Frame

Class name	Frame
Description	This role denotes a Cartesian right handed coordinate system.
Parent class	AutomationMLBaseRoleClassLib/AutomationMLBaseRole
Path for element reference	AutomationMLBaseRoleClassLib/AutomationMLBaseRole/Frame

NOTE 1 An AML object referencing the RoleClass “Frame” is useable to define reference coordinate systems like attachment points.

NOTE 2 To define reference coordinate systems like attachment points an AML object referencing the RoleClass “Frame” is useable.

5.3 AutomationMLInterfaceClassLib

5.3.1 InterfaceClass COLLADAInterface

The InterfaceClass “COLLADAInterface” shall be used as specified in Table 3.

Table 3 – InterfaceClass COLLADAInterface

Class name	COLLADAInterface	
Description	The InterfaceClass "COLLADAInterface" shall be used in order to reference external COLLADA documents and to publish interfaces that are defined inside an external COLLADA document.	
Parent class	AutomationMLInterfaceClassLib/AutomationMLBaseInterface/ExternalDataConnector	
Path for element reference	AutomationMLInterfaceClassLib/AutomationMLBaseInterface/ExternalDataConnector/COLLADAInterface	
Attributes	refType (AttributeDataType="xs:string")	The attribute "refType" specifies whether the referenced COLLADA document has an explicit or implicit character. The allowed values are "explicit" or "implicit". The attribute is mandatory.
	target (AttributeDataType="xs:token")	The attribute "target" specifies the SID path of a COLLADA element within the referenced document. The attribute is optional.

NOTE The InterfaceClass "COLLADAInterface" additionally inherits the attribute "refURI" from the parent InterfaceClass "ExternalDataConnector".

5.3.2 InterfaceClass AttachmentInterface

The InterfaceClass "AttachmentInterface" shall be used as specified in Table 4.

Table 4 – InterfaceClass AttachmentInterface

Class name	AttachmentInterface
Description	The InterfaceClass "AttachmentInterface" specifies an interface for geometric or kinematic links between AML objects with RoleClass Frame.
Parent class	AutomationMLInterfaceClassLib/AutomationMLBaseInterface
Path for element reference	AutomationMLInterfaceClassLib/AutomationMLBaseInterface/AttachmentInterface

6 Frame attribute

An InternalElement or a SystemUnitClass may require a frame attribute that represents its geometric position in relation to other objects. For this, the following provisions apply:

- Each frame shall be based on a three dimensional, orthogonal, right-handed coordinate system.
- The elements InstanceHierarchy and SystemUnitClassLib specify a three dimensional, orthogonal, right-handed coordinate system with standard basis. The positive z axis is considered upward, the positive x direction defines the right axis and the negative y direction defines the forward axis.
- The relative translations x , y and z as well as the rotations rx , ry and rz shall be specified as sub-attributes in an AML attribute „Frame“ defined in Table 5 and Table 6. The relative translations x , y and z shall be given in relation to the parent coordinate System specified in the previous point. The rotations rx , ry and rz shall be executed in the order rx , ry and rz with respect to the fixed axes of the parent coordinate System. The origin of the translated Frame shall retain its position (x , y , z) during the spatially-fixed rotation. The attribute "Frame" shall affect its containing AML object and all its children.

NOTE This means, that the point of rotation rotates whereas the frame position remains unchanged. This avoids double changes.

- If the attribute "Frame" is not specified, the default values of x , y , z , rx , ry and rz shall be 0.

- If the attribute “Frame” is specified, all sub-attributes x , y , z , rx , ry and rz shall be listed. Any unused sub-attribute shall have the default value 0.

The attribute “Frame” shall be used as complex attribute specified in Table 5 and Table 6. An example is given in A.1.2.

Table 5 – Attribute “Frame”

Attribute	AttributeDataType	Description
Frame	This attribute has no AttributeDataType since attribute has no value.	The attribute “Frame” shall be used as structure element for the storage of the sub-attributes specified in Table 6.

Table 6 – Sub-attributes of the attribute “Frame”

Attribute	AttributeDataType	Description
x	xs:double	The attribute “x” shall be used to specify the relative position in meters along the x axis of the coordinate system specified by the parent element. The value of attribute “Unit” of the CAEX element “Attribute” shall be “m”.
y	xs:double	The attribute “y” shall be used to specify the relative position in meters along the y axis of the coordinate system specified by the parent element. The value of attribute “Unit” of the CAEX element “Attribute” shall be “m”.
z	xs:double	The attribute “z” shall be used to specify the relative position in meters along the z axis of the coordinate system specified by the parent element. The value of attribute “Unit” of the CAEX element “Attribute” shall be “m”.
rx	xs:double	The attribute “rx” shall be used to specify the relative rotation in degrees around the x axis of the coordinate system specified by the parent element. The value of attribute “Unit” of the CAEX element “Attribute” shall be “deg”.
ry	xs:double	The attribute “ry” shall be used to specify the relative rotation in degrees around the y axis of the coordinate system specified by the parent element. The value of attribute “Unit” of the CAEX element “Attribute” shall be “deg”.
rz	xs:double	The attribute “rz” shall be used to specify the relative rotation in degrees around the z axis of the coordinate system specified by the parent element. The value of attribute “Unit” of the CAEX element “Attribute” shall be “deg”.

7 Integration of COLLADA documents

Regarding referencing COLLADA documents, the following provisions apply:

- A reference from an AML object to a COLLADA document shall be modelled by means of a CAEX ExternalInterface derived from the AML InterfaceClass “COLLADAInterface” as defined in 5.3.1.
- The COLLADA document shall be referenced by its URI within the attribute “refURI” of this CAEX ExternalInterface.
- The value of the attribute “target” shall point at an element within the COLLADA document and shall follow the syntax of a SID path.
- If the attribute “target” is not present or its content is empty and the URI has no fragment, the element “scene” of the COLLADA document shall be considered.
- The COLLADA document and its entry point shall be resolved as specified in Table 7. A decision tree to resolve the different entries is shown in Figure A.1.

- The value “explicit” of the attribute “refType” shall be used for AML objects (henceforth named explicit AML objects) that are part of a data representation.

NOTE 1 An example of explicit AML objects is the geometry of a robot and a conveyor in a common geometric scene.

- The value “implicit” of the attribute “refType” shall be used for AML objects (henceforth named implicit AML objects) that are used for special purposes for referencing objects that are already represented, e.g. for publishing frames or referencing of objects that are already represented.

NOTE 2 An example of implicit AML objects is the reference of a robot's geometry that has been already explicitly referenced. The implicit referencing avoids double representation of geometries.

- Objects in the COLLADA document only may be referenced using value “implicit” if they are part of a COLLADA object or identical with a COLLADA object, which already was referenced with value “explicit”. Implicit AML objects shall be modelled as direct or indirect children of explicit AML objects.
- In cases of a COLLADA document (henceforth named main document) referencing other COLLADA documents (henceforth named subdocuments) according to the rules of instantiation and external referencing as specified in ISO/PAS 17506 all references to elements within the referenced COLLADA subdocuments shall only be done via the COLLADA main document. An example is given in A.2.3.
- An AML object shall have zero or one “COLLADAInterface” as direct child.

Table 7 – Rules for resolving document and entry point

		Attribute “refURI”	
		URI with fragment	URI without fragment
Attribute “target”	SID path starts with an ID	URI specifies the document. SID path specifies the entry point.	URI specifies the document. SID path specifies the entry point.
	SID path starts without an ID	URI specifies the document and a corresponding element for the entry point. SID path specifies the entry point.	Undefined status. This combination shall not be used.
	Missing or its value is empty	URI specifies the document and the entry point.	URI specifies the document. The element “scene” specifies the entry point.

8 Attachment of two AML objects

Clause 8 specifies the normative provisions for the attachment of two geometric AML objects. An example is given in Clause A.3.

Regarding attaching two AML objects, the following provisions apply:

- An AML object shall have zero or one “AttachmentInterface” as direct child.
- A link between two ExternalInterfaces assigned to a InterfaceClass “AttachmentInterface” shall be a unidirectional connection with fixed geometric coupling using a CAEX InternalLink element.
- The relative transformation from the AML object referenced by the attribute RefPartnerSideA to the object referenced by the attribute RefPartnerSideB shall remain constant even if the object of RefPartnerSideA is transformed, resulting into a corresponding transformation of RefPartnerSideB in case of a transformation of RefPartnerSideA. A transformation of RefPartnerSideB does not modify the transformation of RefPartnerSideA.

NOTE 1 RefPartnerSideA and RefPartnerSideB are attributes of the CAEX InternalLink element.

- As an attachment point is part of an AML object, the “AttachmentInterface” shall be part of the corresponding AML object.
- If an “AttachmentInterface” is child of an AML object with the RoleClass “Frame” or an inherited RoleClass, the object to be moved shall be the parent AML object of the AML object with the RoleClass “Frame” or an inherited RoleClass (even if the parent AML object has the RoleClass “Frame” or an inherited RoleClass).
- If an AML object owning an “AttachmentInterface” has the RoleClass “Frame” or an inherited RoleClass and it has an interface of type “COLLADAInterface”, the parent AML object shall have an interface of type “COLLADAInterface” as well. The “COLLADAInterface” of the parent AML object represents the boundary of the geometry of the attached geometry.
- If an attachment refers to a geometric element, a “COLLADAInterface” shall reference a COLLADA node of the visual scene. The attachment shall move the complete geometry (recursive up to the node attribute of a bind_kinematics_model element) of the attached object without modification of the kinematic state of the attached object.
- If an attachment refers to a kinematic element, a “COLLADAInterface” shall reference a COLLADA element inside a COLLADA kinematics_scene element. The attachment shall affect the joint configuration of the attached kinematic object when moving the parent AML object of the attached object. The importing tool is responsible for the calculation of such coupled kinematic elements.

NOTE 2 To establish a bidirectional connection two unidirectional links are necessary, with each link in an opposite direction to the other.

NOTE 3 It is possible to link an “AttachmentInterface” more than once.

9 Meta information about the COLLADA source tool

In order to simplify the data exchange between a source tool and a destination tool, it is useful to store information about the source tool directly into the COLLADA document. Hence, the following provisions apply:

- Each COLLADA document shall provide information about the tool which has written the COLLADA document.
- In a data exchange tool chain, the last tool editing the document shall store this information in the COLLADA document.
- This information shall be stored within the child element “asset” of the element “COLLADA”.
- The meta information shall consist of:
 - vendor of the writer tool;
 - URL of the writer tool;
 - name of the writer tool;
 - product version of the writer tool;
 - product release number of the writer tool;
 - creation timestamp and timestamp of the last modification of the COLLADA document;
 - definition of unit and up axis.
- The meta information shall be stored by means of the elements shown in Table 8.

Table 8 – Meta information about the COLLADA source tool

XML tag name COLLADA 1.4.1	XML tag name ISO/PAS 17506	Type	Level	Example
"author"	"author"	xs:string	Mandatory	ToolX Vendor
"authoring_tool"	"authoring_tool"	xs:string	Mandatory	"ToolX" "0.2" "123 prealpha"
"comments"	"author_website"	xs:anyURI	Mandatory	http://www.ToolX-Vendor.org
"created"	"created"	xs:dateTime	Mandatory	2014-04-14T11:37:18.1875000
"modified"	"modified"	xs:dateTime	Mandatory	2014-04-14T11:37:18.1875000
"unit meter=... name=..."	"unit meter=... name=..."	xs:string	Mandatory	name="meter" meter="1.0"
"up_axis"	"up_axis"	xs:string	Mandatory	Z_UP

The XML elements shall contain the following information:

- "author" provides the vendor of the writer tool;
- "authoring_tool" provides the name, the product version and the product release number of the writer tool separated by a blank character and each of them enclosed in quotation marks;
- in case of ISO/PAS 17506 "author_website" provides the URL of the writer tool, in case of COLLADA 1.4.1 "comments" provides the URL of the writer tool;
- "created" and "modified" provide the creation timestamp and timestamp of the last modification according to the rules as specified in ISO/PAS 17506;
- "unit" and "up_axis" shall be defined as specified in ISO/PAS 17506 or COLLADA 1.4.1

The required XML text in case of ISO/PAS 17506 and COLLADA 1.4.1 by means of an example is shown in Figure 2 and Figure 3, respectively.

```
<asset>
  <contributor>
    <author>ToolX Vendor</author>
    <author_website>http://www.ToolX-Vendor.org</author_website>
    <authoring_tool>"ToolX" "0.2" "123 prealpha"</authoring_tool>
  </contributor>
  <created>2014-04-14T11:37:18.1875000</created>
  <modified>2014-04-14T11:37:18.1875000</modified>
  <!-- ..... -->
  <unit name="meter" meter="1.0" />
  <up_axis>Z_UP</up_axis>
</asset>
```

Figure 2 – Required XML text in case of ISO/PAS 17506

```
<asset>
  <contributor>
    <author>ToolX Vendor</author>
    <comments>http://www.ToolX-Vendor.org</comments>
    <authoring_tool>"ToolX" "0.2" "123 prealpha"</authoring_tool>
  </contributor>
  <created>2014-04-14T11:37:18.1875000</created>
  <modified>2014-04-14T11:37:18.1875000</modified>
  <!-- ..... -->
  <unit name="meter" meter="1.0" />
  <up_axis>Z_UP</up_axis>
</asset>
```

Figure 3 – Required XML text in case of COLLADA 1.4.1

Annex A (informative)

Referencing methods for geometric/kinematic descriptions

A.1 Integration of a common COLLADA document with explicit referencing

A.1.1 General

Clause A.1 describes the explicit referencing methods for geometric and kinematic models, which are located in COLLADA documents. Figure A.1 shows the different methods used in the following subclauses by means of a decision tree.

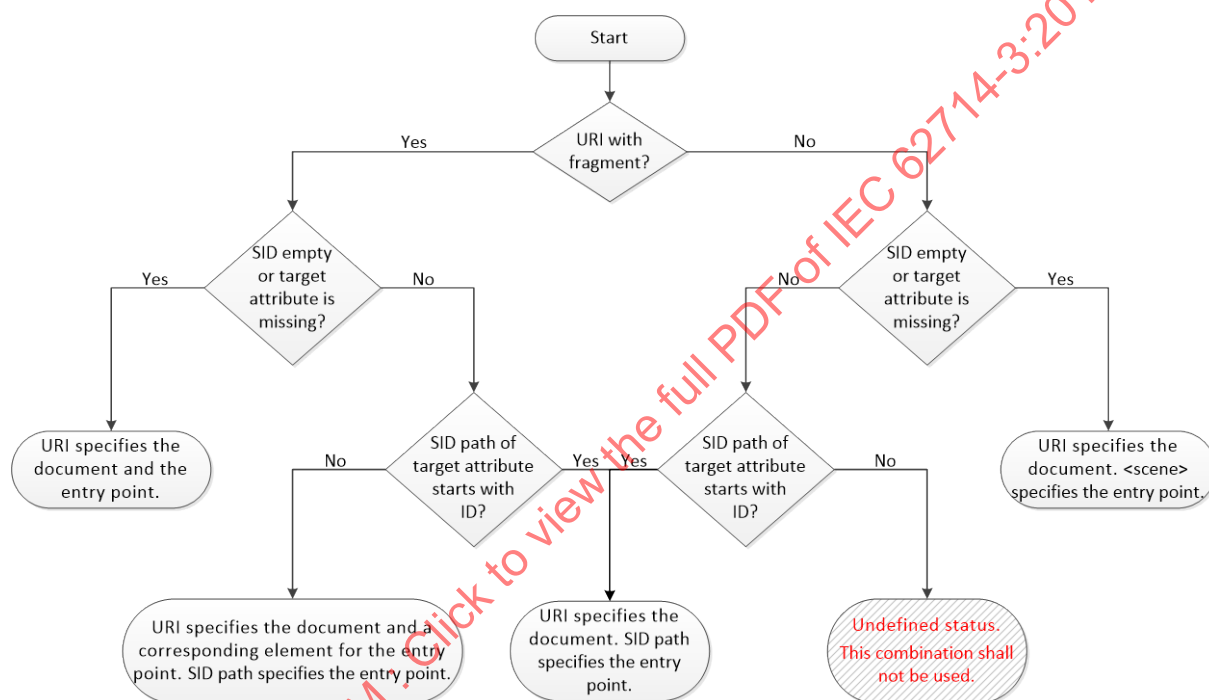


Figure A.1 – Decision tree for different referencing methods

NOTE 1 In this document, path or subpath strings are presented in a short form like “PATH1”, “PATH2”, etc. This serves the readability and is acting for a real path or subpath. Same numbers indicate the same path and thus the semantic equality.

NOTE 2 For better readability, only relevant parts of AML/COLLADA documents are shown. Missing parts are denoted with “[...]”.

NOTE 3 For better readability, any RefBaseClassPath attribute which points to the “AutomationMLInterfaceClassLib/AutomationMLBaseInterface/ExternalDataConnector/COLLADAInterface” class is substituted with “PATH_CI”.

NOTE 4 For better readability, any RefBaseClassPath attribute which points to the “AutomationMLInterfaceClassLib/AutomationMLBaseInterface/AttachmentInterface” class is substituted with “PATH_AI”.

NOTE 5 For better readability, any RefBaseRoleClassPath attribute which points to the “AutomationMLDMIRoleClassLib/DiscManufacturingEquipment/StaticObject” class is substituted with “PATH_SO”.

NOTE 6 For better readability, any RefBaseRoleClassPath attribute which points to the “AutomationMLBaseRoleClassLib/AutomationMLBaseRole/Frame” class is substituted with “PATH_FRAME”.

A.1.2 Definition of the Frame attribute

The following example shows how to define the geometric position and orientation of two CAEX Objects by using the frame attribute mentioned in Clause 6. The corresponding AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with two InternalElements “BaseFrame” and “ChildFrame” (see Figure A.2).

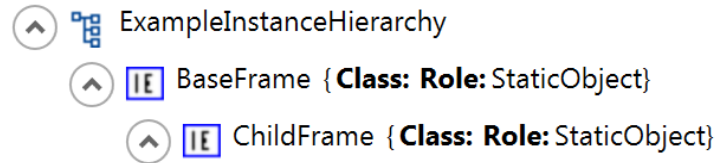


Figure A.2 – Two frames represented in the InstanceHierarchy of an AML document

To describe the complete coordinate transformation of the InternalElement “ChildFrame” with respect to its parent Node “BaseFrame”, the former object owns the frame attribute. Figure A.3 shows the relevant part of the AML document and how to define the frame attribute and its subattributes.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <Version>1.0.0</Version>
  <InternalElement Name="BaseFrame" ID="d0235958-bce3-4b48-bec9-08d1033cc49e">
    <InternalElement Name="ChildFrame" ID="07218e99-f4d8-4dd9-ba92-16c6f0dd4f46">
      <Attribute Name="Frame">
        <Attribute Name="x" AttributeDataType="xs:double">
          <Value>1.00</Value>
        </Attribute>
        <Attribute Name="y" AttributeDataType="xs:double">
          <Value>1.00</Value>
        </Attribute>
        <Attribute Name="z" AttributeDataType="xs:double">
          <Value>1.00</Value>
        </Attribute>
        <Attribute Name="rx" AttributeDataType="xs:double">
          <Value>45.00</Value>
        </Attribute>
        <Attribute Name="ry" AttributeDataType="xs:double">
          <Value>45.00</Value>
        </Attribute>
        <Attribute Name="rz" AttributeDataType="xs:double">
          <Value>45.00</Value>
        </Attribute>
      </Attribute>
      <RoleRequirements RefBaseRoleClassPath="AutomationMLDMIRoleClassLib/./StaticObject"
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="AutomationMLDMIRoleClassLib/./StaticObject" />
  </InternalElement>
</InstanceHierarchy>
  
```

Figure A.3 – XML representation of the AML document

Assuming that the coordinate system $K_1(x_1, y_1, z_1)$ belongs to the InternalElement “ChildFrame”, Figure A.4 depicts its relative translation by x , y and z as well as its spatially fixed rotation by rx , ry and rz with respect to the three axes of the coordinate system $K_0(x_0, y_0, z_0)$ specified by the parent element “BaseFrame”. By the use a homogeneous transformationmatrix H_1^0 and the six subattributes mentioned in Table 6, the relative translation and orientation between K_1 and K_0 can be described in a compact format as follows:

$$H_1^0(x, y, z, rx, ry, rz) = \begin{bmatrix} R_1^0 & d_1^0 \\ 0 & 1 \end{bmatrix} \in R^{4 \times 4}. \quad (A.1)$$

Hereby the Rotationmatrix R_1^0 expresses the orientation of K_1 relative to K_0 . Clause 6 determines the order of rotation by rx , ry and rz . In the context of spatially fixed rotations, it is

recalled that the appropriate rotation matrices have to be multiplied from the left hand side. Thus the rotational part of $\mathbf{H}_1^0$ is given as

$$\mathbf{R}_1^0 = \mathbf{Rot}_z(rz)\mathbf{Rot}_y(ry)\mathbf{Rot}_x(rx) . \quad (\text{A.2})$$

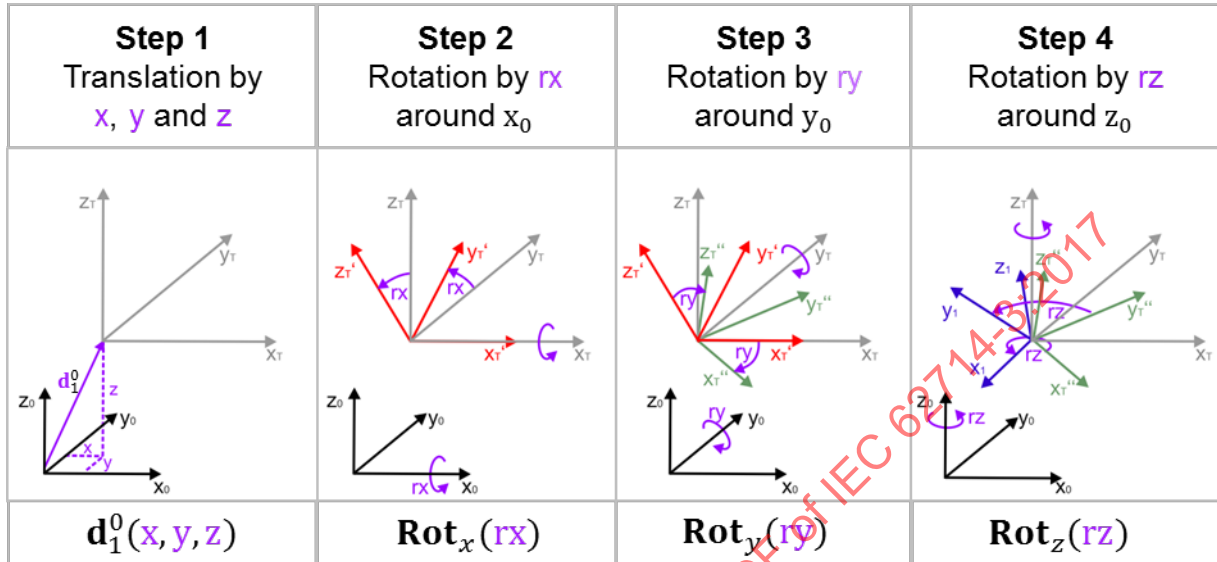


Figure A.4 – Translation and spatially fixed rotation

Each rotation matrix is characterized as

$$\mathbf{Rot}_x(rx) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(rx) & -\sin(rx) \\ 0 & \sin(rx) & \cos(rx) \end{bmatrix}, \quad \mathbf{Rot}_y(ry) = \begin{bmatrix} \cos(ry) & 0 & \sin(ry) \\ 0 & 1 & 0 \\ -\sin(ry) & 0 & \cos(ry) \end{bmatrix} \text{ and}$$

$$\mathbf{Rot}_z(rz) = \begin{bmatrix} \cos(rz) & -\sin(rz) & 0 \\ \sin(rz) & \cos(rz) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The displacement vector is described by

$$\mathbf{d}_1^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}.$$

As shown in Figure A.4, the origin of the translated Frame retains its Cartesian coordinates x , y and z during the spatially-fixed rotation. By substituting $\sin(ri) = sr_i$ and $\cos(ri) = cr_i$, equation (A.2) and the multiplication in equation (A.1) yields the compact format

$$\mathbf{H}_1^0 = \begin{bmatrix} cr_z cr_y & cr_z sr_y sr_x - sr_z cr_x & cr_z sr_y cr_x + sr_z sr_x & x \\ sr_z cr_y & sr_z sr_y sr_x + cr_z cr_x & sr_z sr_y cr_x - cr_z sr_x & y \\ -sr_y & cr_y sr_x & cr_y sr_x & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

which describes the translation and orientation of the InternalElement “ChildFrame” with respect to its parent Node “BaseFrame”.

A.1.3 Structure of the COLLADA documents

The scene used in the following samples consists of two cubes, one on the top of the other. In the visual scene a node for each cube is instantiated. Each node references the same geometry element, but applies a different name (“RedCube”, “BlueCube”), material (red, blue) and transformation. Figure A.5 shows the rendered scene with the corresponding COLLADA scene graph.

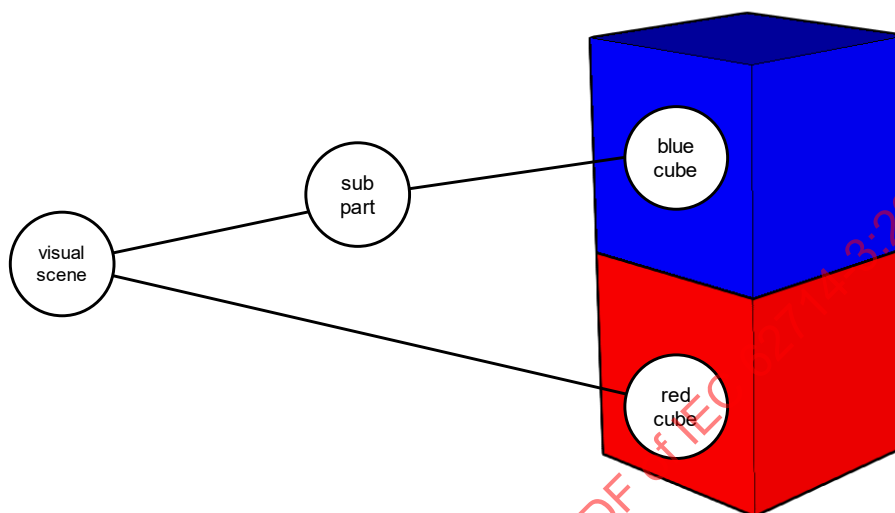


Figure A.5 – COLLADA scene used in this example

In this example, two COLLADA documents are used to define the complete visual scene. The node hierarchy is defined in the COLLADA main document “red_blue_cubes.dae”. The geometry of the cube is defined in a library within the separate COLLADA subdocument “cube.dae”. Each node references the subdocument “cube.dae” and instantiates a cube geometry according to the rules of instantiation and external referencing as specified in ISO/PAS 17506. Figure A.6 depicts the structure and references used in this example.

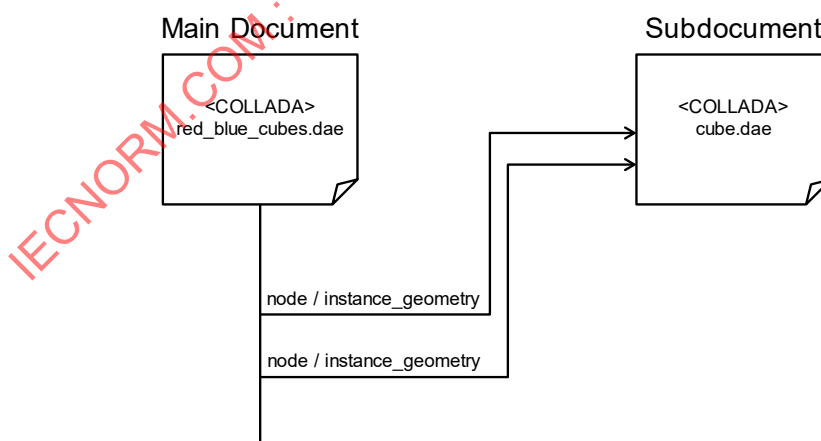


Figure A.6 – Structure and References

The content of the COLLADA subdocument “cube.dae” is shown in Figure A.7.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the geometry of a unit cube stored as reusable library
element</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_geometries>
    <geometry id="cube-geom" name="cube">
      <mesh>
        <source id="cube-geom-positions">
          <float_array id="cube-geom-positions-array" count="144"> 1 1 0 1 0 1 1 1 0 0 1 1 1 1
1 1 1 1 0 1 0 1 1 0 0 0 0 1 1 0 0 0 0 1 0 0 1 0 0 1 1 0 0 0 0 0 1 0 0 1 1 1 1 0 0 1 0 1 1
1 1 1 1 0 1 1 1 1 0 0 1 0 1 0 1 0 1 1 0 0 1 1 1 1 1 1 1 1 0 1 0 1 1 0 0 1 1 0 0 0 1 0 0
0 0 1 0 0 1 1 0 1 0 0 0 1 0 0 0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 0 1 1 0 0 0 0 0 1</float_array>
          <technique_common>
            <accessor count="48" source="#cube-geom-positions-array" stride="3">
              <param name="X" type="float" />
              <param name="Y" type="float" />
              <param name="Z" type="float" />
            </accessor>
          </technique_common>
        </source>
        <source id="cube-geom-normals">
          <float_array id="cube-geom-normals-array" count="144"> 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 -1
0 0 -1 0 0 -1 0 0 -1 0 0 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1
0 0 1 0 0 -1 0 0 -1 0 0 -1 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 -1
0 0 -1 0 0 -1 0 0 -1 0 0 1 0 0 1 0 0 1 0 0 1 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 1 0 0 1 0 0 1 0 0
1 0 0</float_array>
          <technique_common>
            <accessor count="48" source="#cube-geom-normals-array" stride="3">
              <param name="X" type="float" />
              <param name="Y" type="float" />
              <param name="Z" type="float" />
            </accessor>
          </technique_common>
        </source>
        <vertices id="cube-geom-vertices">
          <input semantic="POSITION" source="#cube-geom-positions" />
          <input semantic="NORMAL" source="#cube-geom-normals" />
        </vertices>
        <triangles count="24" material="mat">
          <input offset="0" semantic="VERTEX" source="#cube-geom-vertices" />
          <p>0 1 2 1 0 3 4 5 6 7 6 5 8 9 10 9 8 11 12 13 14 15 14 13 16 17 18 17 16 19 20 21
22 23 22 21 24 25 26 25 24 27 28 29 30 31 30 29 32 33 34 33 32 35 36 37 38 39 38 37 40 41 42
41 40 43 44 45 46 47 46 45</p>
        </triangles>
      </mesh>
    </geometry>
  </library_geometries>
</COLLADA>

```

Figure A.7 – Content of the COLLADA document cube.dae

The main document ("red_blue_cubes.dae") including the node hierarchy and external references to the cube geometry ("cube.dae") is shown in Figure A.8. Each instantiated geometry overwrites the material symbol with its custom material definition.

```
<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the node hierarchy and instantiates cubes from
subdocuments</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="red-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0 0 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="blue-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0 0 1 1 </color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="red" name="Red">
      <instance_effect url="#red-fx" />
    </material>
    <material id="blue" name="Blue">
      <instance_effect url="#blue-fx" />
    </material>
  </library_materials>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="redcube" name="RedCube">
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#red" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
      <node id="subpart">
        <translate>0 0 1</translate>
        <node name="bluecube" sid="bluecube">
          <instance_geometry url="./cube.dae#cube-geom">
            <bind_material>
              <technique_common>
                <instance_material symbol="mat" target="#blue" />
              </technique_common>
            </bind_material>
          </instance_geometry>
        </node>
      </node>
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#visualscene" />
  </scene>
</COLLADA>
```

Figure A.8 – Content of the COLLADA document red_blue_cubes.dae

A.1.4 Referencing using URI and fragments without a target and ID

The following example AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with one single InternalElement “RedCube”. The latter includes an CAEX ExternalInterface of type “COLLADAInterface” named “FileLinkExplicit”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib” as depicted in Figure A.9.

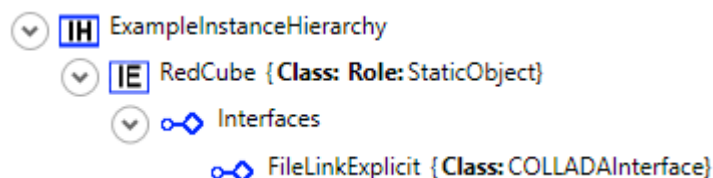


Figure A.9 – “RedCube” – Hierarchy of the AML document

Figure A.10 shows the relevant part of the AML document and how to define attributes for referencing.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#redcube</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>
  
```

Figure A.10 – XML representation of the AML document

The ExternalInterface “FileLinkExplicit” points to the COLLADA document, which is shown in Figure A.11. The red cube is referenced unambiguously by the use of the relative URI “./red_blue_cubes.dae#redcube”.

```

<visual_scene id="visualscene">
  <node id="redcube" name="RedCube">
    <instance_geometry url="./cube.dae#cube-geom">
      <bind_material>
        <technique common>
          <instance_material symbol="mat" target="#red" />
        </technique common>
      </bind_material>
    </instance_geometry>
  </node>
  <node id="subpart"> [...] </node>
</visual_scene>
  
```

Figure A.11 – Referencing the red cube by ID

A.1.5 Referencing using URI and fragments including a target without an ID

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with a single InternalElement “BlueCube”. The latter one includes an ExternalInterface “COLLADAInterface” named “FileLinkExplicit”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib” as depicted in Figure A.12.

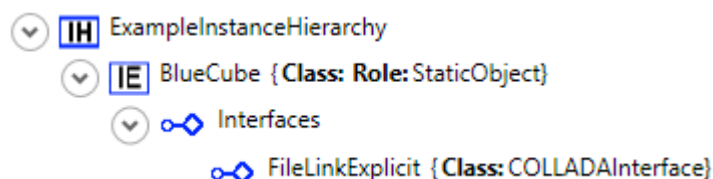


Figure A.12 – “BlueCube” – Hierarchy of the AML document

Figure A.13 shows the relevant part of the AML document and how to define attributes for referencing.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="BlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#subpart</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./bluecube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>
  
```

Figure A.13 – XML representation of the AML document

The ExternalInterface “FileLinkExplicit” points to the COLLADA document, which is shown in Figure A.14. First the SID’s entry point is specified by the relative URI “./red_blue_cubes.dae#subpart”. The target element defined in the COLLADA document named “bluecube” is then resolved by browsing from the referenced element “subpart” according to the rules of a SID.

```

<visual_scene id="visualscene">
  <node id="redcube" name="RedCube"> [...] </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube"> [...] </node>
  </node>
</visual_scene>
  
```

Figure A.14 – Referencing the blue cube

A.1.6 Referencing using URI without a fragment, including a target and an ID

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with a single InternalElement “BlueCube”. The latter one includes an ExternalInterface “COLLADAInterface” named “FileLinkExplicit”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib” as depicted in Figure A.15.

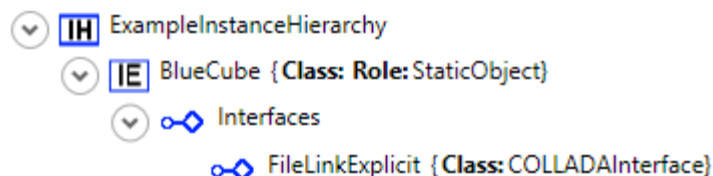


Figure A.15 – Hierarchy of the AML document

Figure A.16 shows the relevant part of the AML document and how to define attributes for referencing.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="BlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>subpart/bluecube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>

```

Figure A.16 – XML representation of the AML document

The ExternalInterface “FileLinkExplicit” points to the COLLADA document, which is shown in Figure A.17. First the SID’s entry point is specified by the relative URI “./red_blue_cubes.dae”. The target element defined in the COLLADA document named “bluecube” is then resolved by browsing from the entry point of the SID “subpart/bluecube”.

```

<visual_scene id="visualscene">
  <node id="redcube" name="RedCube"> [...] </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube"> [...] </node>
  </node>
</visual_scene>

```

Figure A.17 – Referencing the blue cube starting from the element “subpart”

A.1.7 Referencing using URI and fragments including a target and an ID

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with a single InternalElement “BlueCube”. The latter one includes an ExternalInterface “COLLADAInterface” named “FileLinkExplicit”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib” as depicted in Figure A.18.

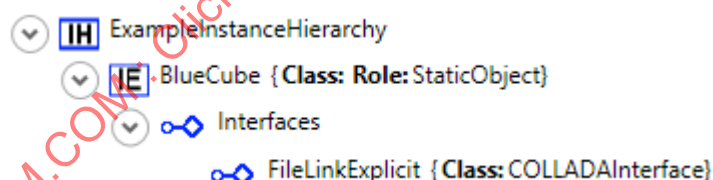


Figure A.18 – Hierarchy of the AML document

Figure A.19 shows the relevant part of the AML document and how to define attributes for referencing.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="BlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#redcube</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>subpart/bluecube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>

```

Figure A.19 – XML representation of the AML document

The ExternalInterface “FileLinkExplicit” points to the COLLADA document, which is shown in Figure A.20. In this example, the blue cube is addressed as already shown in A.1.6. Initially the SID’s entry point is specified by the relative URI “./red_blue_cubes.dae#redcube”. At this point it is demonstrated that the fragment of the specified URI is actually not necessary for referencing the blue cube. In this case, the blue cube is resolved by evaluating the SID beginning with an ID. As the ID of any SID must be unique across the entire document, the proposed referencing is unambiguous as well.

```
<visual_scene id="visualscene">
  <node id="redcube" name="RedCube"> [...] </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube"> [...] </node>
  </node>
</visual_scene>
```

Figure A.20 – Referencing the blue cube

A.1.8 Referencing using URI without a fragment, target and ID

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with a single InternalElement “RedAndBlueCube”. The latter one includes an ExternalInterface “COLLADAInterface” named “FileLinkExplicit”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib” as depicted in Figure A.21.

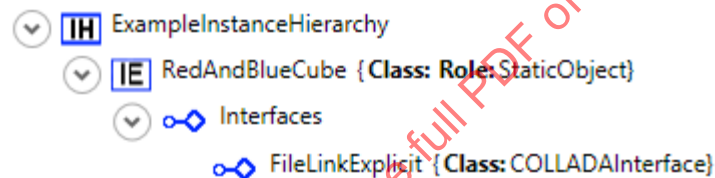


Figure A.21 – Hierarchy of the AML document

Figure A.22 shows the relevant part of the AML document and how to define attributes for referencing.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>
```

Figure A.22 – XML representation of the AML document

The ExternalInterface “FileLinkExplicit” points to the COLLADA document, which is shown in Figure A.23. In this sample, the complete COLLADA element “scene” is addressed. Both the red and the blue cube are referenced unambiguously by the use of the relative URI “./red_blue_cubes.dae” without any additional fragment or target.

```

<COLLADA>
[... ]
<library_visual_scenes>
  <visual_scene id="visualscene">
    <node id="redcube" name="RedCube">
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#red" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
    <node id="subpart">
      <translate>0 0 1</translate>
      <node name="bluecube" sid="bluecube">
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#blue" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
    </node>
  </visual_scene>
</library_visual_scenes>
<scene>
  <instance_visual_scene url="#visualscene" />
</scene>
</COLLADA>

```

Figure A.23 – Referencing the complete COLLADA scene

A.2 Implicit referencing of COLLADA elements

A.2.1 General

Clause A.2 describes the implicit referencing method for geometric and kinematic models, which are located in external COLLADA documents. Implicit referencing is used to refer from a CAEX object to a COLLADA object and to provide additional information about the COLLADA object at CAEX level, e.g. a frame for providing attachment capabilities. In this case the COLLADA object is already known from a different context, e.g. from an explicit reference (see Clause A.1). Subclause A.2.2 describes the implicit referencing to a single COLLADA document, whereas A.2.3 describes the implicit referencing to nested COLLADA documents. Subclause A.2.4 describes how to publish elements respectively frames of a COLLADA document.

A.2.2 Implicit referencing

The following sample uses the visual scene taken from Subclause A.1.3 (COLLADA document "red_blue_cubes.dae"). The AML document consists of an InstanceHierarchy named "ExampleInstanceHierarchy" with three InternalElements as depicted in Figure A.24. The InternalElement "RedAndBlueCubeExplicit" includes an ExternalInterface "COLLADAInterface" named "FileLinkExplicit". Both InternalElements "BlueCubeImplicit" and "RedCubeImplicit" include an ExternalInterface "COLLADAInterface" named "FileLinkImplicit", which are part of the standard InterfaceClassLibrary "AutomationMLInterfaceClassLib".

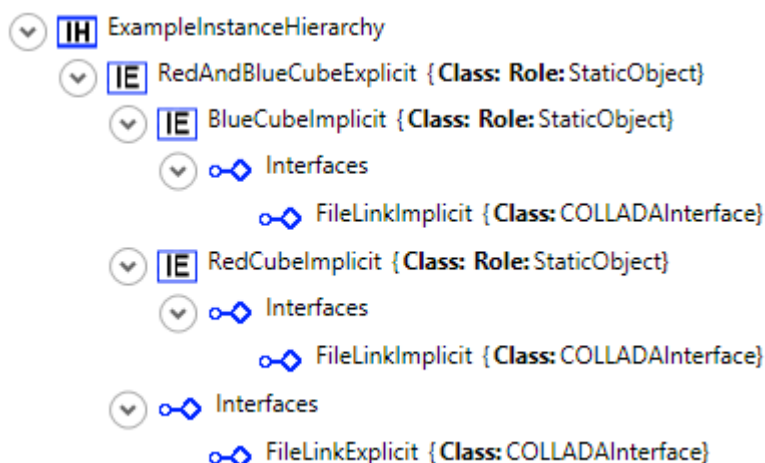


Figure A.24 – Implicit Referencing: Hierarchy of the AML document

The ExternalInterface “FileLinkExplicit” of the InternalElement “RedAndBlueCubeExplicit” points to the COLLADA document and addresses the complete COLLADA scene explicitly (see subclause A.1.8). Both the red and the blue cube are referenced unambiguously by the use of the relative URI “./red_blue_cubes.dae” without any additional fragment or target.

The InternalElements “RedCubeImplicit” and “BlueCubeImplicit” reference the red and blue cube implicitly. Figure A.25 shows the relevant part of the AML document and how to define attributes for implicit referencing.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="BlueCubeImplicit" ID="GUID5">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#bluecube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <InternalElement Name="RedCubeImplicit" ID="GUID3">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#redcube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InternalElement>
</InstanceHierarchy>
```

Figure A.25 – XML representation of the AML document

In the case of visualization, it is not necessary to render both the red and blue cube once more because both were visualised already via the explicit reference of the “RedAndBlueCubeExplicit” element. In this case the explicitly referenced scene (“RedAndBlueCubeExplicit”) is visualised.

A.2.3 Implicit referencing to COLLADA subdocuments

According to external referencing as specified in ISO/PAS 17506 or COLLADA 1.4.1 the geometric description of a scene may be divided into separate COLLADA documents. In such a case a main COLLADA document references elements which are stored in referenced COLLADA subdocuments. Subclause A.2.3 describes the implicit referencing to COLLADA elements, which are located in COLLADA subdocuments and referenced from a main COLLADA document. Figure A.26 depicts the structure and relations used in this example.

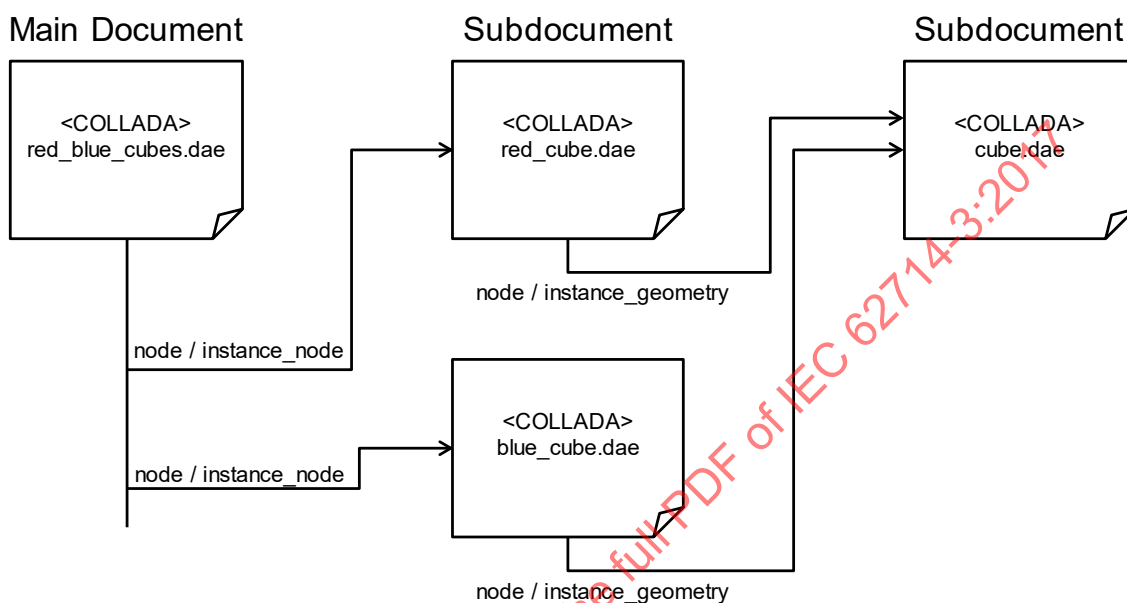


Figure A.26 – Structure and relations of referenced COLLADA subdocuments

The following sample uses the visual scene taken from Subclause A.1.3 with a modified structure of COLLADA documents. The main COLLADA document ("red_blue_cubes.dae") consists of two nodes, which each instantiate a node provided in a COLLADA subdocument ("red_cube.dae" and "blue_cube.dae"). The latter COLLADA subdocuments consist of a library, which defines a single node and a material. Each node of both documents instantiate a cube geometry, which is defined in a COLLADA subdocument ("cube.dae"). While the definition of the geometry remains the same as used in A.1.3, the main COLLADA document ("red_blue_cubes.dae") is modified and two additional COLLADA subdocuments are introduced ("red_cube.dae" and "blue_cube.dae") in this example. Figure A.27 depicts the content of the modified main COLLADA document ("red_blue_cubes.dae").

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the node hierarchy and instantiates nodes from
subdocuments</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="redcube" name="RedCubeInstance">
        <instance_node sid="redcuberef" url="./red_cube.dae#redcube" />
      </node>
      <node id="bluecube" name="blueCubeInstance">
        <translate>0 0 1</translate>
        <instance_node sid="bluecuberef" url="./blue_cube.dae#bluecube" />
      </node>
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#visualscene" />
  </scene>
</COLLADA>

```

Figure A.27 – Content of the modified COLLADA document red_blue_cubes.dae

The visual scene consists of two node elements (redcube and bluecube) which each instantiate a node in a subdocument ("red_cube.dae" and "blue_cube.dae"). The content of both COLLADA subdocuments are shown in Figure A.28 and Figure A.29.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines a node which instantiates a cube geometry from a
COLLADA subdocument and applies a red material</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="red-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0 0 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="red" name="Red">
      <instance_effect url="#red-fx" />
    </material>
  </library_materials>
  <library_nodes>
    <node id="redcube" name="RedCube">
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#red" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
  </library_nodes>
</COLLADA>

```

Figure A.28 – Content of the COLLADA document red_cube.dae

The COLLADA document “red_cube.dae” defines a node named “RedCube” within a library which may be referenced and instantiated. Additionally, a red material is defined to attach to the instantiated geometry in the COLLADA subdocument “cube.dae”. In a similar manner, the node and geometry of the blue cube named “BlueCube” is defined in the COLLADA subdocument “blue_cube.dae” as depicted in Figure A.29.

```
<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines a node which instantiates a cube geometry from a
COLLADA subdocument and applies a blue material</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="blue-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0 0 1 1 </color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="blue" name="Blue">
      <instance_effect url="#blue-fx" />
    </material>
  </library_materials>
  <library_nodes>
    <node id="bluecube" name="BlueCube">
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#blue" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
  </library_nodes>
</COLLADA>
```

Figure A.29 – Content of the COLLADA document blue_cube.dae

To reference the blue or red cube in the COLLADA subdocuments from an AML document the full path starting from the root COLLADA document shall be specified. The following example shows an extract of an AML document to demonstrate the implicit referencing to elements located in referenced COLLADA subdocuments.

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with three InternalElements as depicted in Figure A.30. All InternalElements include an ExternalInterface of type “COLLADAInterface”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib”.

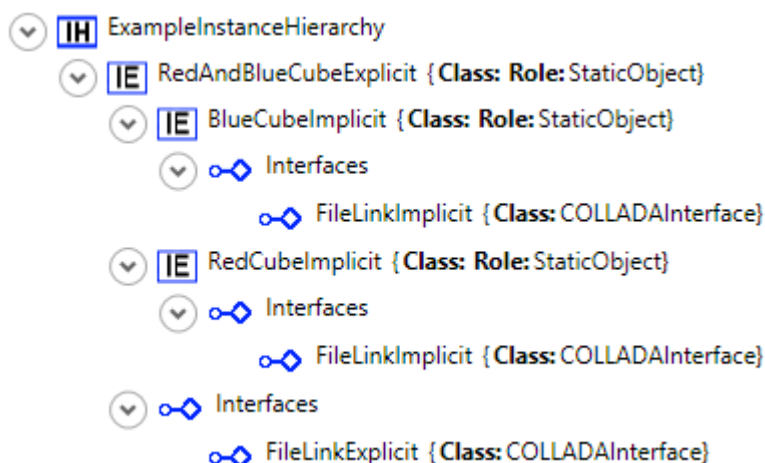


Figure A.30 – Implicit Referencing: Hierarchy of the AML document

The ExternalInterface “FileLinkExplicit” of the InternalElement “RedAndBlueCubeExplicit” points to the COLLADA document and addresses the complete COLLADA scene explicitly (see A.1.8). Both the red and the blue cube are referenced unambiguously by the use of the relative URI “./red_blue_cubes.dae” without any additional fragment or target. The InternalElements “RedCubeImplicit” and “BlueCubeImplicit” reference the red and blue cube implicitly, which both are located in a COLLADA subdocument. Figure A.31 shows the relevant part of the AML document for implicit referencing.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
  </InternalElement>
  <InternalElement Name="RedCubeImplicit" ID="GUID3">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_cube.dae</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>redcube/redcuberef</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <InternalElement Name="BlueCubeImplicit" ID="GUID5">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./blue_cube.dae</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>bluecube/bluecuberef</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InstanceHierarchy>
```

Figure A.31 – XML representation of the AML document

A.2.4 Publishing elements of a COLLADA document in CAEX

Subclause A.2.4 shows a use case where both (explicit and implicit) referencing methods are relevant. This Subclause describes how to publish frames of COLLADA documents and its referencing within an AML document.

Geometric and kinematic COLLADA elements can be published at CAEX level to make them available for usage. This makes it possible to reference them and assign further information. In this example the COLLADA document and its visual scene taken from A.1.3 (COLLADA document “red_blue_cubes.dae”) with an additional COLLADA element is used. The additional node named “bluecubeframe” is attached to the blue cube and represents a frame.

Figure A.32 shows the modified visual scene with the corresponding COLLADA scene graph.

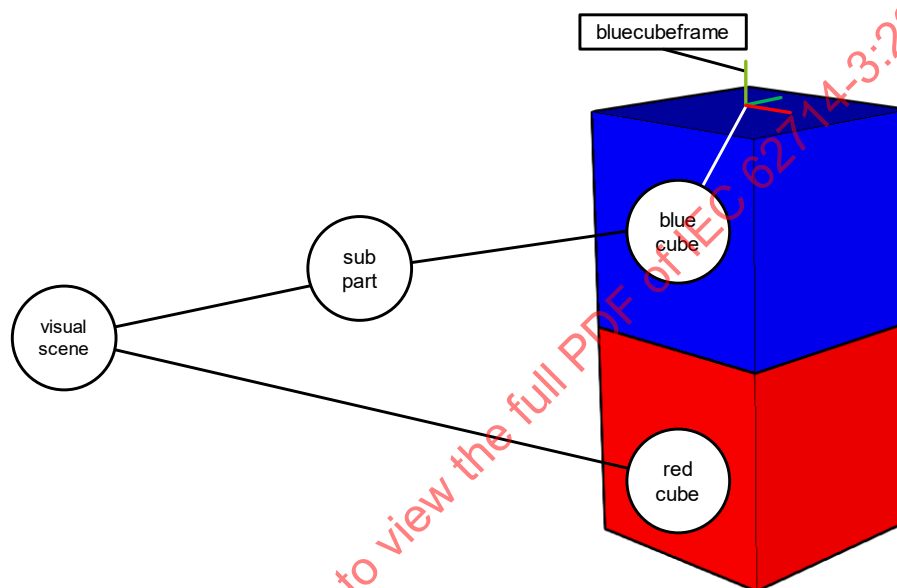


Figure A.32 – Modified COLLADA scene with additional node

The relevant part of the COLLADA document, in particular the additional frame node, is shown in Figure A.33.

```
<visual_scene id="visual_scene">
  <node id="redcube" name="RedCube">
    <instance_geometry url="./cube.dae#cube-geom">
      <bind_material>
        <technique_common>
          <instance_material symbol="mat" target="#red" />
        </technique_common>
      </bind_material>
    </instance_geometry>
  </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube">
      <node sid="bluecubeframe">
        <translate>0.5 0.5 1.0</translate>
      </node>
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#blue" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
  </node>
</visual_scene>
```

Figure A.33 – Additional frame element in COLLADA document

In this example, the AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with an InternalElement “RedAndBlueCubeExplicit”, which includes an ExternalInterface “COLLADAInterface” named “FileLinkExplicit”. This ExternalInterface is used to reference the complete scene explicitly as depicted in Figure A.34.

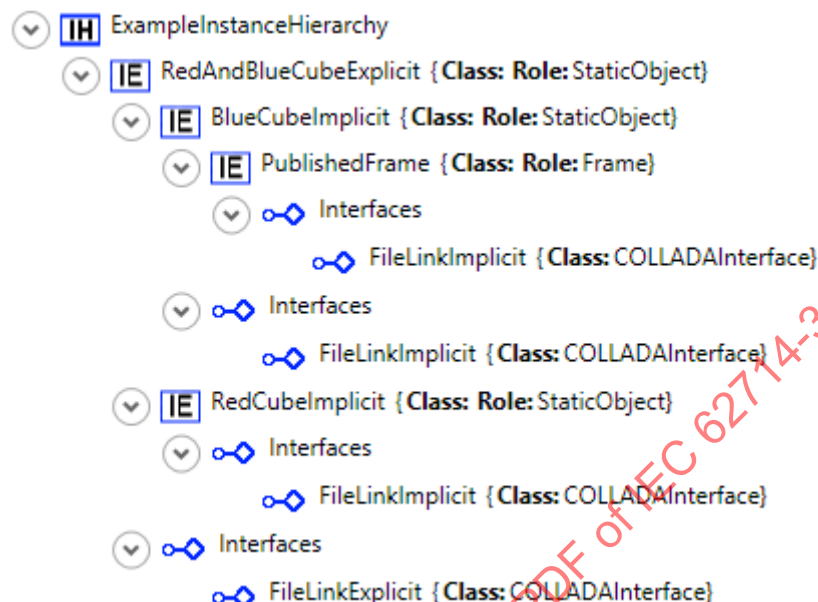


Figure A.34 – Publishing frames: Hierarchy of the AML document

The InternalElement “RedAndBlueCubeExplicit” has two InternalElements named “BlueCubeImplicit” and “RedCubeImplicit”. The InternalElement “BlueCubeImplicit” has an InternalElement “PublishedFrame” with assigned RoleClass “Frame”. The ExternalElement “FileLinkImplicit” of the InternalElement “PublishedFrame” references the COLLADA node “bluecubeframe” in the COLLADA document node implicitly.

Figure A.35 shows the relevant part of the AML document and how to define attributes for implicit referencing.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="BlueCubeImplicit" ID="GUID3">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#subpart</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./bluecube</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="PublishedFrame" ID="GUID5">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#subpart</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./bluecube/bluecubeframe</Value>
      </Attribute>
      <Attribute Name="AdditionalInformation1" AttributeDataType="xs:token">
        <Value>AdditionalAttributeValue</Value>
      </Attribute>
      <Attribute Name="AdditionalInformation2" AttributeDataType="xs:token">
        <Value>AdditionalAttributeValue</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_FRAME" />
  </InternalElement>
  <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InternalElement>
  <InternalElement Name="RedCubeImplicit" ID="GUID7">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID8">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#redcube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InternalElement>
</InstanceHierarchy>

```

Figure A.35 – XML representation of the AML document

NOTE A published CAEX object with assigned RoleClass "Frame" does not have to point to a COLLADA geometry object (it may point to a COLLADA object without geometry). In former case the transformation information is taken from the COLLADA object and applied to the published CAEX object with assigned RoleClass "Frame", even if the CAEX frame object provides a CAEX frame attribute. If the COLLADA object does not provide any transformation, the one from the frame attribute of the CAEX object with assigned RoleClass "Frame" is used.

A.3 Attachment between objects in CAEX

Clause A.3 describes how to attach a geometric AML object to another geometric AML object (both of type InternalElement). Both AML objects have a reference to any COLLADA geometry objects by using explicit references. Furthermore both AML objects have a child InternalElement Frame with assigned RoleClass "Frame". Each InternalElement Frame has an implicit ExternalInterface derived from the InterfaceClass "COLLADAInterface" which references a node defined in the COLLADA document. Additionally each InternalElement

Frame has ExternalInterface derived from the InterfaceClass “AttachmentInterface”, which may be connected or linked. The structure is shown in Figure A.36.

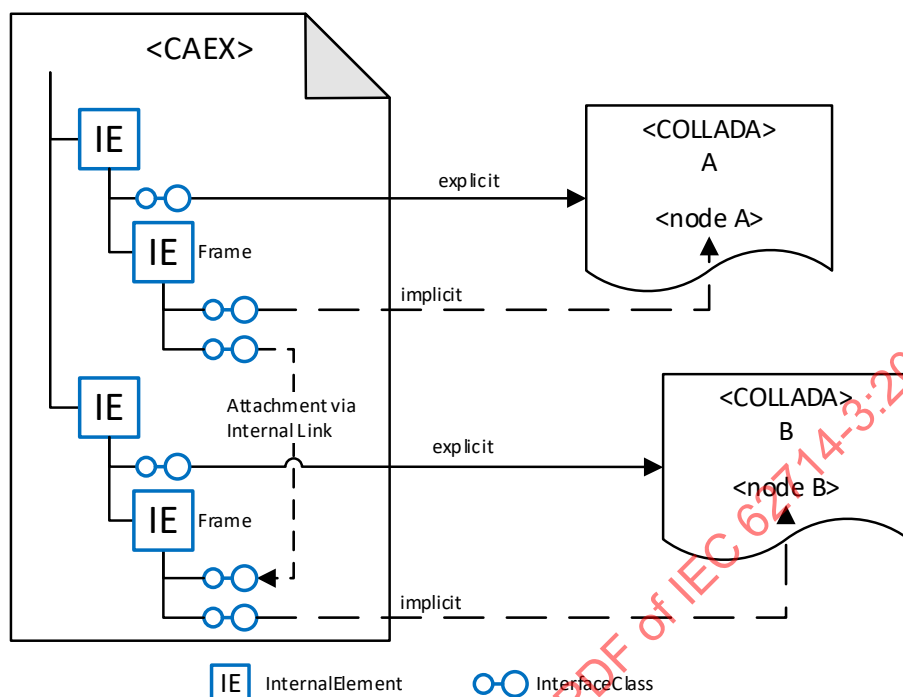


Figure A.36 – Structure for attachments between objects in CAEX

In this example, the COLLADA document and its visual scene taken from Subclause A.2.3 with an additional node is used. The additional node, which is attached to the blue cube, represents the published frame. An additional yellow cube is defined in a separate COLLADA document (“yellow_cube.dae”) with an additional node, which is published. The connection between both additional nodes (attachment frames) is established within the AML document. Figure A.37 shows the visualization of the yellow cube (“YellowCube”) and its additional frame.

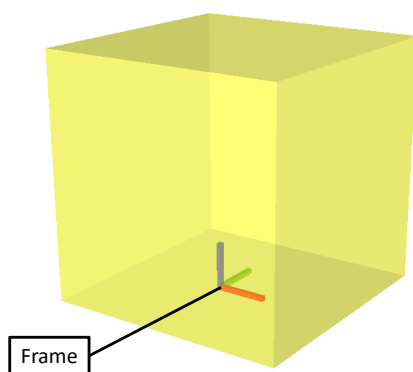


Figure A.37 – Visualization of yellow cube with additional frame

The complete COLLADA document of the yellow cube (“yellow_cube.dae”) is shown in Figure A.38. It uses the geometry definition from the external COLLADA document “cube.dae” taken from Subclause A.1.3.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines a node which instantiates a cube geometry from a
COLLADA subdocument and applies a yellow material. An additional node is
provided.</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="yellow-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 1 0 0.5 </color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="yellow" name="Yellow">
      <instance_effect url="#yellow-fx" />
    </material>
  </library_materials>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="yellowcube" name="YellowCube">
        <translate>0 0 2</translate>
        <node sid="yellowcubeframe">
          <translate>0.5 0.5 0</translate>
        </node>
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#yellow" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#visualscene" />
  </scene>
</COLLADA>

```

Figure A.38 – COLLADA document of yellow cube with additional frame

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with two InternalElements (“RedAndBlueCubeExplicit” and “YellowCubeExplicit”). In this example, the yellow cube (“YellowCube”) from the separate COLLADA document is attached to the blue one (“BlueCube”). Both InternalElements include an ExternalInterface “COLLADAInterface” named “FileLinkExplicit”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib”.

Each ExternalInterface named “FileLinkExplicit” of both InternalElements points to the separate COLLADA documents and addresses the complete COLLADA scene explicitly (see also A.1.8). In addition both InternalElements have a child InternalElement (“PublishedFrame”) with applied RoleClass “Frame”, which is part of the standard RoleClassLibrary as well. Each InternalElement “PublishedFrame” has two ExternalInterfaces: an ExternalInterface derived from the InterfaceClass “COLLADAInterface” to reference a COLLADA node implicitly (“FileLinkImplicit”) and an ExternalInterface derived from the InterfaceClass “AttachmentInterface” to define an attachment (“Attachment”). The structure is depicted in Figure A.39.

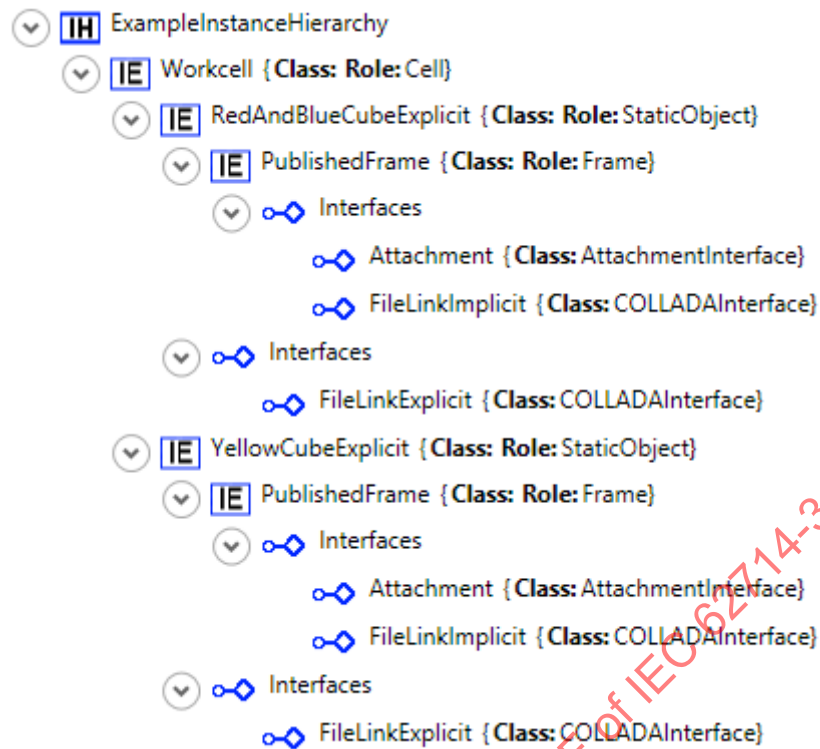


Figure A.39 – Hierarchy of the AML document

Figure A.40 shows the relevant part of the AML document and how to publish frames for attaching objects, which are defined in AML.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="Workcell" ID="GUID0">
    <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
      <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./red_blue_cubes.dae</Value>
        </Attribute>
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>explicit</Value>
        </Attribute>
      </ExternalInterface>
    <InternalElement Name="PublishedFrame" ID="GUID3">
      <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID4" />
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID5">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./red_blue_cubes.dae#subpart</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./bluecube/bluecubeframe</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_FRAME" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <InternalElement Name="YellowCubeExplicit" ID="GUID6">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID7">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./yellow_cube.dae</Value>
      </Attribute>
    </ExternalInterface>
    <InternalElement Name="PublishedFrame" ID="GUID8">
      <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID9" />
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID10">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./yellow_cube.dae#yellowcube</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./yellowcube#frame</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_FRAME" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <InternalLink Name="AttachmentBlueMovesYellow" RefPartnerSideA="GUID3:Attachment" RefPartnerSideB="GUID7:Attachment" />
  <RoleRequirements RefBaseRoleClassPath="PATH5/Cell" />
</InternalElement>
</InstanceHierarchy>

```

Figure A.40 – XML representation of the AML document

As a result the yellow cube is attached to the blue one as depicted in Figure A.41. In opposite to mounting the attaching does not include any additional transformation of the attached object. The connection is unidirectional. If the blue cube is translated or rotated, the transformation of the blue cube affects the yellow cube as well.

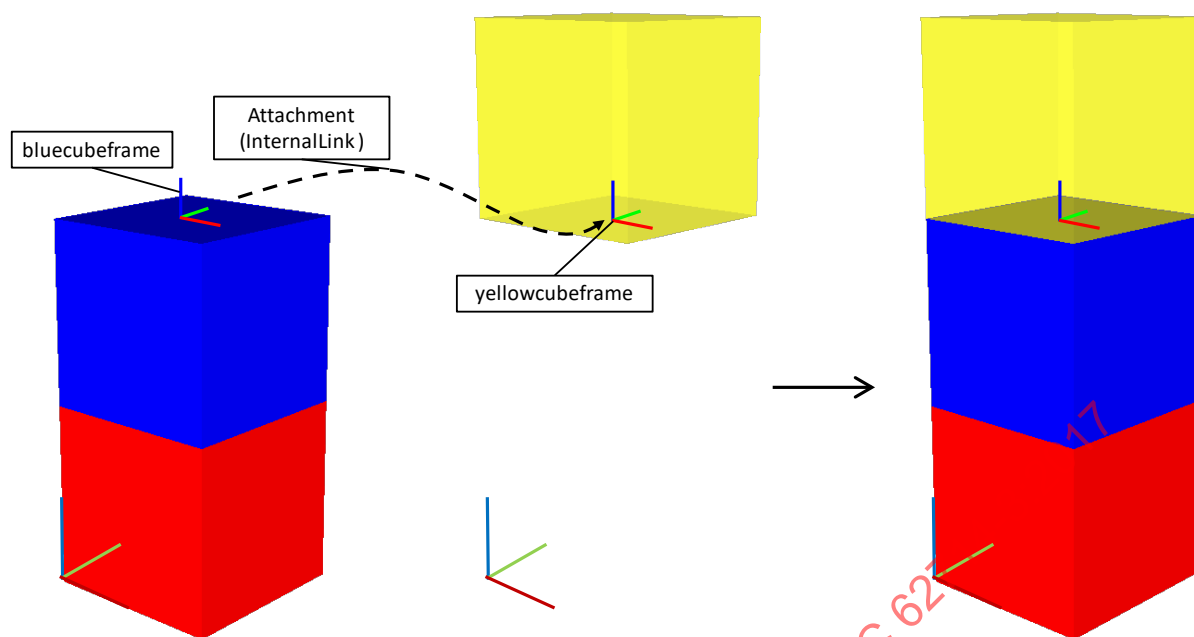


Figure A.41 – Attachment between geometric AML objects

Finally, the connection is established by the use of an InternalLink, which connects both published frames. Figure A.42 shows the relevant part of the AML document and how to define an InternalLink. The link is stored at the CAEX InternalElement “Workcell”, which is the lowest common parent of the connected CAEX objects. This is according to IEC 62714-1.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="Workcell" ID="GUID0">
    [...]
    <InternalLink Name="AttachmentBlueRedMovesYellowCube"
      RefPartnerSideA="GUID3:Attachment"
      RefPartnerSideB="GUID8:Attachment" />
  </InternalElement>
</InstanceHierarchy>
```

Figure A.42 – XML representation of the AML document

Annex B (informative)

Modelling of kinematic systems and their combination in AML

B.1 General

The following clauses describe the modelling of kinematic systems and the combination in AML. In the clauses a scene will be composed stepwise containing a linear unit (Clause B.2) and a robot (Clause B.3) fixed to the linear unit (Clause B.4). In addition a gripper will be defined and mounted to the robot (Clause B.5). Finally a workpiece will be attached to the gripper (Clause B.6).

NOTE 1 For better readability all substitutions for RefBaseRoleClassPath attributes from A.1 apply.

NOTE 2 For better readability any RefBaseRoleClassPath attribute which points to the "AutomationMLMIRoleClassLib/ManufacturingEquipment/Robot" class is substituted with "PATH_ROB".

NOTE 3 For better readability any RefBaseRoleClassPath attribute which points to the "AutomationMLMIRoleClassLib/ManufacturingEquipment/Carrier" class is substituted with "PATH_CAR".

B.2 Modelling an AML document of a linear unit in CAEX and COLLADA

B.2.1 General

Clause B.2 describes the modelling of a linear unit. In B.2.2 the visual scene is described. In the subsequent Subclauses B.2.3 – B.2.7 the scene is complemented stepwise with kinematic elements. Finally, the AML document will be defined and combined with the kinematic system in B.2.8.

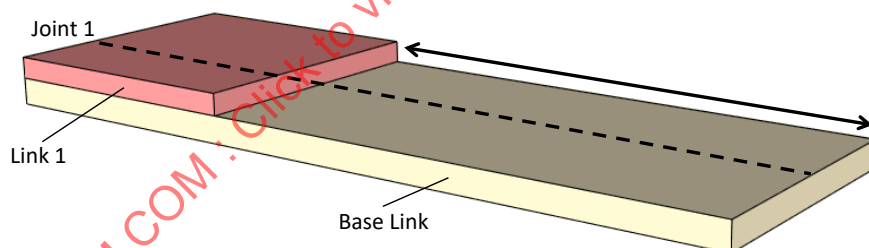


Figure B.1 – Visualization of the linear unit

B.2.2 Definition of the visual scene

The visual scene of the linear unit in this example consists of two geometric elements, one on the top of the other as depicted in Figure B.1. To keep the COLLADA content to a minimum, the geometries of the linear unit are represented by simple rectangular cuboids, which are defined within the COLLADA document "cube.dae" as introduced in Figure A.7. Each cube has a different scaling and provides a custom material. The lower cube represents the base link of the linear unit while the upper cube represents the slider.

Figure B.2 depicts the content of the COLLADA document "linear_unit.dae" which defines the geometric representation of the unit.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the geometry and kinematic of a linear unit</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="base-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0.9 0.8 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="link-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0.6 0.6 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="base-mat" name="BaseMat">
      <instance_effect url="#base-fx" />
    </material>
    <material id="link-mat" name="LinkMat">
      <instance_effect url="#link-fx" />
    </material>
  </library_materials>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="linearunit" name="LinearUnit">
        <node id="geombase" name="GeomBase">
          <scale>3.0 1.0 0.1</scale>
          <instance_geometry url="./cube.dae#cube-geom">
            <bind_material>
              <technique_common>
                <instance_material symbol="mat" target="#base-mat" />
              </technique_common>
            </bind_material>
          </instance_geometry>
          <node id="geomlink1" name="GeomLink1">
            <translate>0 0 1</translate>
            <scale>0.33 1 0.8</scale>
            <rotate sid="transform">0 0 0 0</rotate>
            <instance_geometry url="./cube.dae#cube-geom">
              <bind_material>
                <technique_common>
                  <instance_material symbol="mat" target="#link-mat" />
                </technique_common>
              </bind_material>
            </instance_geometry>
          </node>
        </node>
      </node>
    </visual_scene>
  </library_visual_scenes>
  <!-- Step 1: Definition of joint elements -->
  <!-- Step 2: Definition of kinematic model element -->
  <!-- Step 3: Definition of articulated system (kinematics) element -->
  <!-- Step 4: Definition of kinematics scene element -->
  <scene>
    <instance_visual_scene url="#visualscene" />
    <!-- Step 5: Instantiation of kinematics scene -->
  </scene>
</COLLADA>

```

Figure B.2 – Definition of the visual scene

Each cuboid is assigned to a node (geombase and geomlink1). In the subclauses B.2.3 – B.2.7, the required kinematic elements will be defined and applied to this scene to complete the kinematic system.

B.2.3 Definition of the joint

In this case the kinematic system has a single prismatic joint named “Joint1”, which is defined in a library and which will be instantiated in a subsequent step. Prismatic joints are defined by the element “prismatic”. Additionally, the direction of the joint’s axis and the optional minimum and maximum limit values are specified according to the rules as specified in ISO/PAS 17506 or COLLADA 1.4.1. The extract of the COLLADA document is shown in Figure B.3.

NOTE Physical units are definable according to ISO/PAS 17506 or COLLADA 1.4.1. By default degrees (angles) and meters (distances) are useable. The latter is redefineable within the “unit” element.

```
<library_joints>
  <joint id="joint1" name="Joint1">
    <prismatic sid="axis0">
      <axis>0 0 1</axis>
      <limits>
        <min>0</min>
        <max>3</max>
      </limits>
    </prismatic>
  </joint>
</library_joints>
```

Figure B.3 – Definition of the joint

B.2.4 Definition of the kinematic model

In Subclause B.2.4, the kinematic model and its hierarchy is defined. First, for the kinematic model all joints are defined or instantiated from a joints library. The following kinematic chain consists of links and a joint and its hierarchical relationship. Any link or joint may provide a transformation. The transformation of a link or a joint is always relative to its predecessor and is described through the elements “translate” and “rotate”. The kinematic model is defined within a library and will be instantiated in a subsequent step. Figure B.4 shows the relevant part of the COLLADA document.

```
<library_kinematics_models>
  <kinematics_model id="linearunit_kinmodel">
    <technique_common>
      <instance_joint sid="inst_joint1" url="#joint1" />
      <link name="BaseLink" sid="baselink">
        <attachment_full joint="linearunit_kinmodel/inst_joint1">
          <rotate>0 1 0 90</rotate>
          <link name="Link1" sid="link1" />
        </attachment_full>
      </link>
    </technique_common>
  </kinematics_model>
</library_kinematics_models>
```

Figure B.4 – Definition of kinematic model

In this example, two links named “BaseLink” and “Link1” are defined, which are connected through a single joint, which references a joint defined in the library as described in the previous subclause.

B.2.5 Definition of the articulated system

Kinematic and dynamic properties are assembled within “articulated_system” elements. Since each articulated system definition only contains either kinematic or dynamic information, two separate “articulated_system” systems are defined within a library.

```

<library articulated_systems>
  <articulated_system id="linearunit_kinematics">
    <kinematics>
      <!-- ... ARTICULATED SYSTEM KINEMATICS ... -->
    </kinematics>
  </articulated_system>
  <articulated_system id="linearunit_motion">
    <motion>
      <!-- ... ARTICULATED SYSTEM MOTION ... -->
    </motion>
  </articulated_system>
</library articulated_systems>

```

Figure B.5 – Definition of the articulated system library

Figure B.5 depicts the structure of the library definition. The comments in the two articulated system elements denote the missing definitions, which will be explained in subclauses B.2.6 and B.2.7.

The “kinematics” section of an articulated system contains information about the kinematic behaviour of an articulated model, for instance activity of joints, limits, or dependencies between joints. Figure B.6 shows the relevant part of the COLLADA document and its additional kinematic parameters used in this example.

```

<articulated_system id="linearunit_kinematics">
  <kinematics>
    <instance_kinematics_model sid="inst_linearunit_kinmodel" url="#linearunit_kinmodel">
      <newparam sid="linearunit_kinematics_inst_linearunit_kinmodel">
        <SIDREF>linearunit_kinematics/inst_linearunit_kinmodel</SIDREF>
      </newparam>
      <newparam sid="linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_axis0">
        <SIDREF>linearunit_kinematics/inst_linearunit_kinmodel/inst_joint1/axis0</SIDREF>
      </newparam>
      <newparam sid="linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_value">
        <float>0.0</float>
      </newparam>
    </instance_kinematics_model>
    <technique_common>
      <axis_info axis="linearunit_kinematics/inst_linearunit_kinmodel/inst_joint1/axis0" sid=
"inst_joint1_info">
        <active>
          <bool>true</bool>
        </active>
        <locked>
          <bool>false</bool>
        </locked>
        <limits>
          <min>
            <float>0.0</float>
          </min>
          <max>
            <float>3.0</float>
          </max>
        </limits>
      </axis_info>
      <frame_origin link="linearunit_kinematics/inst_linearunit_kinmodel/baselink" sid="base
frame" />
      <frame_tip link="linearunit_kinematics/inst_linearunit_kinmodel/link1" sid="flangefram
e">
        <translate>1 2 3</translate>
      </frame_tip>
    </technique_common>
  </kinematics>
</articulated_system>

```

Figure B.6 – Definition of the kinematic articulated system

To provide dynamic information an additional articulated system shall be defined. The motion section of an articulated system contains information about the dynamic behaviour of an articulated model, for instance velocities of joints, acceleration, or deceleration of joints. Figure B.7 depicts the relevant part of the COLLADA document and its additional dynamics parameters used in this example.

```

<articulated_system id="linearunit_motion">
  <motion>
    <instance_articulated_system sid="inst_linearunit_kinematics" url="#linearunit_kinematic
s">
      <newparam sid="linearunit_motion_linearunit_kinematics_inst_LinearUnit_kinematics_mode
l">
        <SIDREF>linearunit_kinematics/linearunit_kinematics_inst_linearunit_kinmodel</SIDREF
>
        </newparam>
        <newparam sid="linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_j
oint1_axis0">
          <SIDREF>linearunit_kinematics/linearunit_kinematics_inst_linearunit_kinmodel_inst_jo
int1_axis0</SIDREF>
          </newparam>
          <newparam sid="linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_j
oint1_value">
            <SIDREF>linearunit_kinematics/linearunit_kinematics_inst_linearunit_kinmodel_inst_jo
int1_value</SIDREF>
            </newparam>
          </instance_articulated_system>
          <technique_common>
            <axis_info axis="linearunit_kinematics/inst_joint1_info">
              <speed>
                <float>0.0</float>
              </speed>
              <acceleration>
                <float>0.0</float>
              </acceleration>
              <deceleration>
                <float>0.0</float>
              </deceleration>
              <jerk>
                <float>0.0</float>
              </jerk>
            </axis_info>
          </technique_common>
        </motion>
      </articulated_system>

```

Figure B.7 – Definition of the motion articulated system

B.2.6 Definition of the kinematic scene

A kinematic scene assembles different kinematic systems. All parameters that are needed for the whole scene are published within a kinematic scene. In this example the kinematic scene instantiates the articulated system, which is defined in B.2.5. Additionally the kinematic model, the joint axis, and its values are published. Figure B.8 shows the relevant part of the COLLADA document.

```

<library_kinematics_scenes>
  <kinematics_scene id="linearunit_kinscene">
    <instance_articulated_system sid="inst_linearunit_motion" url="#linearunit_motion">
      <bind symbol="linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_LinearU
nit_kinematics_model">
        <param ref="linearunit_motion/linearunit_motion_linearunit_kinematics_inst_LinearUni
t_kinematics_model" />
      </bind>
      <bind symbol="linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_axis0">
        <param ref="linearunit_motion/linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_axis0" />
      </bind>
      <bind symbol="linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_value">
        <param ref="linearunit_motion/linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_value" />
      </bind>
    </instance_articulated_system>
  </kinematics_scene>
</library_kinematics_scenes>

```

Figure B.8 – Definition of the kinematic scene

B.2.7 Assembling of the scene

In the element “scene”, all kinematics scenes are instantiated that are used. In addition the linkage to the visual scene is done here, as depicted in Figure B.9. This linkage defines the flow of the joint value from a kinematic calculation to the transformation in the geometry.

```

<scene>
  <instance_visual_scene url="#visualscene" />
  <instance_kinematics_scene sid="inst_linearunit_kinscene" url="#linearunit_kinscene">
    <bind_kinematics_model node="linearunit">
      <param>linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_LinearUnit_kinematics_model</param>
    </bind_kinematics_model>
    <bind_joint_axis target="geomlink1/transform">
      <axis>
        <param>linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_axis0</param>
      </axis>
      <value>
        <param>linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_value</param>
      </value>
    </bind_joint_axis>
  </instance_kinematics_scene>
</scene>

```

Figure B.9 – Instantiation of the kinematic scene

B.2.8 Combination of CAEX and COLLADA into AML

The CAEX document consists of an InstanceHierarchy named "ExampleInstanceHierarchy" with a single InternalElement "LinearUnit". The latter one includes an ExternalInterface named "FileLinkExplicit" derived from the InterfaceClass "COLLADAInterface", which is part of the standard InterfaceClassLibrary "AutomationMLInterfaceClassLib". This interface is used to reference the complete scene explicitly and in particular the linear unit as depicted in Figure B.10.

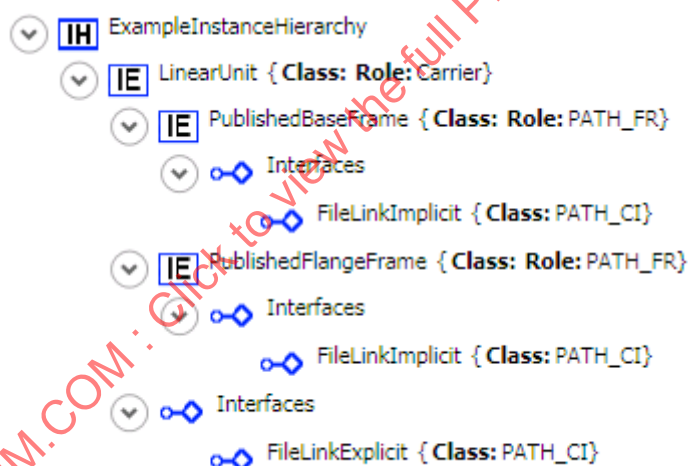


Figure B.10 – Hierarchy of the AML document

The InternalElement "LinearUnit" has two additional InternalElements named "PublishedBaseFrame" and "PublishedFlangeFrame" with assigned RoleClass "Frame". Both InternalElements reference kinematic elements in the COLLADA document implicitly. Figure B.11 shows the relevant part of the AML document.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="LinearUnit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./linear_unit.dae</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="PublishedBaseFrame" ID="GUID3">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./linear_unit.dae#linearunit_kinematics</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./baseframe</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
  </InternalElement>
  <InternalElement Name="PublishedFlangeFrame" ID="GUID5">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./linear_unit.dae#linearunit_kinematics</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./flangeframe</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
  </InternalElement>
  <RoleRequirements RefBaseRoleClassPath="PATH_CAR" />
</InternalElement>
</InstanceHierarchy>

```

Figure B.11 – XML representation of the AML document

Each published frame references unambiguously the “articulated_system” element within the COLLADA document by the use of the relative URI “./linear_unit.dae#linearunit_kinematics”. The target elements “frame_tip” and “frame_origin” (see Figure B.6) are resolved by browsing from the entry point by the use of a relative path “./flangeframe” and “./baseframe” according to the rules of a SID.

B.3 Modelling an AML document of a robot in CAEX and COLLADA

B.3.1 General

Clause B.3 describes the modelling of a Selective Compliance Assembly Robot Arm (SCARA), which is a specialized type of an industrial robot. In subclause B.3.2, the visual scene is described. In the subsequent subclauses, the scene is complemented stepwise with kinematic elements. Finally the AML document will be defined and combined with the kinematic system.

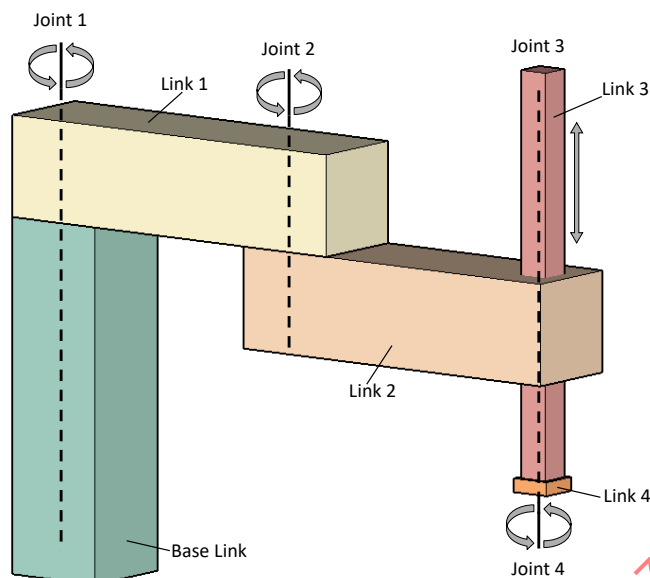


Figure B.12 – Visualization of the robot

B.3.2 Definition of the visual scene

The visual scene of the SCARA robot in this example consists of five geometric elements as depicted in Figure B.12. Each geometric element represents a link of the robot. To keep the COLLADA content to a minimum, each link is represented by a rectangular cuboid, which is defined within the COLLADA document “cube.dae” as introduced Figure A.7. To obtain the needed shape each cube has a different scaling. The first cube represents the base link of the SCARA robot while the remaining cubes represent the subsequent links.

Figure B.13 depicts the content of the COLLADA document “robot.dae” which defines the geometric representation of the robot.

```
<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the geometry and kinematic of a SCARA robot</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="base-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0.60 0.77 0.72 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="link1-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0.94 0.91 0.77 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="link2-fx">
```

```

<profile_COMMON>
  <technique sid="COMMON">
    <phong>
      <diffuse>
        <color>0.93 0.80 0.69 1</color>
      </diffuse>
    </phong>
  </technique>
</profile_COMMON>
</effect>
<effect id="link3-fx">
  <profile_COMMON>
    <technique sid="COMMON">
      <phong>
        <diffuse>
          <color>0.85 0.59 0.57 1</color>
        </diffuse>
      </phong>
    </technique>
  </profile_COMMON>
</effect>
<effect id="link4-fx">
  <profile_COMMON>
    <technique sid="COMMON">
      <phong>
        <diffuse>
          <color>0.95 0.64 0.41 1</color>
        </diffuse>
      </phong>
    </technique>
  </profile_COMMON>
</effect>
</library_effects>
<library_materials>
  <material id="base-mat" name="BaseMat">
    <instance_effect url="#base-fx" />
  </material>
  <material id="link1-mat" name="Link1Mat">
    <instance_effect url="#link1-fx" />
  </material>
  <material id="link2-mat" name="Link2Mat">
    <instance_effect url="#link2-fx" />
  </material>
  <material id="link3-mat" name="Link3Mat">
    <instance_effect url="#link3-fx" />
  </material>
  <material id="link4-mat" name="Link4Mat">
    <instance_effect url="#link4-fx" />
  </material>
</library_materials>
<library_visual_scenes>
  <visual_scene id="visualscene">
    <node id="robot" name="Robot">
      <scale>2 2 2</scale>
      <node id="geombase" name="GeomBase">
        <translate>-0.05 -0.05 0</translate>
        <scale>0.1 0.1 0.35</scale>
        <instance_geometry url="../cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#base-mat" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      <node id="geomlink1" name="GeomLink1">
        <translate>0 0 1</translate>
        <scale>3.8 1 0.2857</scale>
        <instance_geometry url="../cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#link1-mat" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      <node id="geomlink2" name="GeomLink2">
        <translate>0.736848 0 -1</translate>
        <scale>0.947376 1 1</scale>
        <instance_geometry url="../cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#link2-mat" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
    </node>
  </visual_scene>
</library_visual_scenes>

```

```

</instance_geometry>
<node id="geomlink3" name="GeomLink3">
  <translate>0.861 0.35 -1</translate>
  <scale>0.08333 0.3 4</scale>
  <instance_geometry url="./cube.dae#cube-geom">
    <bind_material>
      <technique_common>
        <instance_material symbol="mat" target="#link3-mat" />
      </technique_common>
    </bind_material>
  </instance_geometry>
<node id="geomlink4" name="GeomLink4">
  <translate>-0.1666 -0.166 -0.0333</translate>
  <scale>1.333 1.333 0.0333</scale>
  <instance_geometry url="./cube.dae#cube-geom">
    <bind_material>
      <technique_common>
        <instance_material symbol="mat" target="#link4-mat" />
      </technique_common>
    </bind_material>
  </instance_geometry>
</node>
</node>
</node>
</node>
</node>
</visual_scene>
</library_visual_scenes>
<!-- Step 1: Definition of joint elements -->
<!-- Step 2: Definition of kinematic model element -->
<!-- Step 3: Definition of articulated system (kinematics) element -->
<!-- Step 4: Definition of kinematics scene element -->
<scene>
  <instance_visual_scene url="#visualscene" />
  <!-- Step 5: Instantiation of kinematics scene -->
</scene>
</COLLADA>

```

Figure B.13 – Definition of the visual scene

Each geometry is assigned to a node (geombase, geomlink1, geomlink2, geomlink3, geomlink4). The next subclauses define the required kinematic elements and apply them to this scene to complete the kinematic system.

B.3.3 Definition of joints

The robot used in this example consists of four joints (rotational, rotational, prismatic, rotational). All joints are defined in a library and instantiated in a subsequent step. Prismatic joints are defined by the element “prismatic” whereas rotational joints are defined by the element “revolute”. While in this example the direction of the joint axis is specified, the lower and upper joint limits are defined exemplarily at a different location. The extract of the COLLADA document is shown in Figure B.14.

```

<library_joints>
  <joint id="joint1" name="joint1">
    <revolute sid="axis0">
      <axis>0 0 1</axis>
    </revolute>
  </joint>
  <joint id="joint2" name="joint2">
    <revolute sid="axis0">
      <axis>0 0 1</axis>
    </revolute>
  </joint>
  <joint id="joint3" name="joint3">
    <prismatic sid="axis0">
      <axis>0 0 1</axis>
    </prismatic>
  </joint>
  <joint id="joint4" name="joint4">
    <revolute sid="axis0">
      <axis>0 0 1</axis>
    </revolute>
  </joint>
</library_joints>

```

Figure B.14 – Definition of joints

B.3.4 Definition of the kinematic model

In Subclause B.3.4, the kinematic model and its hierarchy is defined. For the kinematic model first all joints are instantiated from a joints library. The following kinematic chain consists of links and joints and its hierarchical relationship. Any link or joint may provide a transformation. The transformation of a link or a joint is always relative to its predecessor and is described through the elements “translate” and “rotate”. The kinematic model is defined within a library and will be instantiated in a subsequent step. Figure B.15 shows the relevant part of the COLLADA document.

```
<library_kinematics_models>
  <kinematics_model id="robot_kinmodel">
    <technique_common>
      <instance_joint sid="inst_joint1" url="#joint1" />
      <instance_joint sid="inst_joint2" url="#joint2" />
      <instance_joint sid="inst_joint3" url="#joint3" />
      <instance_joint sid="inst_joint4" url="#joint4" />
      <link name="BaseLink" sid="baselink">
        <attachment_full joint="robot_kinmodel/inst_joint1">
          <translate>0.0 0.0 0.3</translate>
          <link name="Link1" sid="link1">
            <attachment_full joint="robot_kinmodel/inst_joint2">
              <translate>0.28 0.0 0.0</translate>
              <link name="Link2" sid="link2">
                <attachment_full joint="robot_kinmodel/inst_joint3">
                  <translate>0.26 0.0 0.0</translate>
                  <link name="Link3" sid="link3">
                    <attachment_full joint="robot_kinmodel/inst_joint4">
                      <translate>0.0 0.0 -0.15</translate>
                      <link name="Link4" sid="link4" />
                    </attachment_full>
                  </link>
                </attachment_full>
              </link>
            </attachment_full>
          </link>
        </attachment_full>
      </link>
    </technique_common>
  </kinematics_model>
</library_kinematics_models>
```

Figure B.15 – Definition of kinematic model

In this example, five links (BaseLink, Link1, Link2, Link3, Link4) are defined which are connected through joints referencing joints defined in the library as described in the previous subclause.

B.3.5 Definition of the articulated system

Kinematic and dynamic properties are assembled within “articulated_system” elements. Since each articulated system definition only contains either kinematic or dynamic information, two separate “articulated_system” systems are defined within a library. Figure B.16 depicts the structure of the articulated system library definition. The comments in the two articulated system elements denote the missing definitions which will be explained subclauses B.3.6 and B.3.7.

```
<library_articulated_systems>
  <articulated_system id="robot_kinematics">
    <kinematics>
      <!-- ... ARTICULATED SYSTEM KINEMATICS ... -->
    </kinematics>
  </articulated_system>
  <articulated_system id="robot_motion">
    <motion>
      <!-- ... ARTICULATED SYSTEM MOTION ... -->
    </motion>
  </articulated_system>
</library_articulated_systems>
```

Figure B.16 – Definition of the articulated system library

The kinematic section of an articulated system contains information about the kinematic behaviour of an articulated model, for instance activity of joints, limits, or dependencies between joints. Figure B.17 shows the relevant part of the COLLADA document and its additional kinematic parameters used in this example.

```
<articulated_system id="robot_kinematics">
  <kinematics>
    <instance_kinematics_model sid="inst_robot_kinmodel" url="#robot_kinmodel">
      <newparam sid="robot_kinematics_inst_robot_kinmodel">
        <SIDREF>robot_kinematics/inst_robot_kinmodel</SIDREF>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint1_axis0">
        <SIDREF>robot_kinematics/inst_robot_kinmodel/inst_joint1/axis0</SIDREF>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint1_value">
        <float>0.0</float>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint2_axis0">
        <SIDREF>robot_kinematics/inst_robot_kinmodel/inst_joint2/axis0</SIDREF>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint2_value">
        <float>0.0</float>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint3_axis0">
        <SIDREF>robot_kinematics/inst_robot_kinmodel/inst_joint3/axis0</SIDREF>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint3_value">
        <float>0.0</float>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint4_axis0">
        <SIDREF>robot_kinematics/inst_robot_kinmodel/inst_joint4/axis0</SIDREF>
      </newparam>
      <newparam sid="robot_kinematics_inst_robot_kinmodel_inst_joint4_value">
        <float>0.0</float>
      </newparam>
    </instance_kinematics_model>
  </kinematics>
  <technique_common>
    <axis_info axis="robot_kinematics_model/inst_joint1/axis0" sid="inst_joint1_info">
      <active>
        <bool>true</bool>
      </active>
      <locked>
        <bool>false</bool>
      </locked>
      <limits>
        <min>
          <float>-180</float>
        </min>
        <max>
          <float>180</float>
        </max>
      </limits>
    </axis_info>
    <axis_info axis="robot_kinematics_model/inst_joint2/axis0" sid="inst_joint2_info">
      <active>
        <bool>true</bool>
      </active>
      <locked>
        <bool>false</bool>
      </locked>
      <limits>
        <min>
          <float>-140</float>
        </min>
        <max>
          <float>140</float>
        </max>
      </limits>
    </axis_info>
    <axis_info axis="robot_kinematics_model/inst_joint3/axis0" sid="inst_joint3_info">
      <active>
        <bool>true</bool>
      </active>
      <locked>
        <bool>false</bool>
      </locked>
      <limits>
        <min>
          <float>0.1</float>
        </min>
        <max>
          <float>-0.2</float>
        </max>
      </limits>
    </axis_info>
  </technique_common>
</articulated_system>
```

```

    </max>
  </limits>
</axis_info>
<axis_info axis="robot_kinematics_model/inst_joint4/axis0" sid="inst_joint4_info">
  <active>
    <bool>true</bool>
  </active>
  <locked>
    <bool>>false</bool>
  </locked>
  <limits>
    <min>
      <float>-360</float>
    </min>
    <max>
      <float>360</float>
    </max>
  </limits>
</axis_info>
<frame_origin link="robot_kinematics/inst_robot_kinmodel/baselink" sid="baseframe" />
<frame_tip link="robot_kinematics/inst_robot_kinmodel/link4" sid="flangeframe">
  <translate>1 2 3</translate>
</frame_tip>
</technique_common>
</kinematics>
</articulated_system>

```

Figure B.17 – Definition of the kinematic articulated system

To provide dynamics information, an additional articulated system shall be defined. The motion section of an articulated system contains information about the dynamics behaviour of an articulated model, for instance velocities of joints, acceleration, or deceleration of joints. Figure B.18 depicts the relevant part of the COLLADA document and its additional dynamics parameters used in this example.

```

<articulated_system id="robot_motion">
  <motion>
    <instance_articulated_system sid="inst_robot_kinematics" url="#robot_kinematics">
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint1_axis0">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint1_axis0</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint1_value">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint1_value</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint2_axis0">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint2_axis0</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint2_value">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint2_value</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint3_axis0">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint3_axis0</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint3_value">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint3_value</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint4_axis0">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint4_axis0</SIDREF>
      </newparam>
      <newparam sid="robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint4_value">
        <SIDREF>robot_kinematics/robot_kinematics_inst_robot_kinmodel_inst_joint4_value</SIDREF>
      </newparam>
    </instance_articulated_system>
    <technique_common>
      <axis_info axis="robot_kinematics/inst_joint1_info">
        <speed>
          <float>0.0</float>
        </speed>
        <acceleration>
          <float>0.0</float>
        </acceleration>
      </axis_info>
    </technique_common>
  </motion>
</articulated_system>

```

```

</acceleration>
<deceleration>
  <float>0.0</float>
</deceleration>
<jerk>
  <float>0.0</float>
</jerk>
</axis_info>
<axis_info axis="robot_kinematics/inst_joint2_info">
  <speed>
    <float>0.0</float>
  </speed>
  <acceleration>
    <float>0.0</float>
  </acceleration>
  <deceleration>
    <float>0.0</float>
  </deceleration>
  <jerk>
    <float>0.0</float>
  </jerk>
</axis_info>
<axis_info axis="robot_kinematics/inst_joint3_info">
  <speed>
    <float>0.0</float>
  </speed>
  <acceleration>
    <float>0.0</float>
  </acceleration>
  <deceleration>
    <float>0.0</float>
  </deceleration>
  <jerk>
    <float>0.0</float>
  </jerk>
</axis_info>
<axis_info axis="robot_kinematics/inst_joint4_info">
  <speed>
    <float>0.0</float>
  </speed>
  <acceleration>
    <float>0.0</float>
  </acceleration>
  <deceleration>
    <float>0.0</float>
  </deceleration>
  <jerk>
    <float>0.0</float>
  </jerk>
</axis_info>
</technique_common>
</motion>
</articulated_system>

```

Figure B.18 – Definition of the motion articulated system

B.3.6 Definition of the kinematic scene

A kinematic scene assembles different kinematic systems. All parameters needed for the whole scene are published within a kinematic scene. In this example the kinematic scene instantiates the articulated system which has been defined in B.3.5. Additionally, the kinematic model, the joint axis, and its values are published. Figure B.19 shows the relevant part of the COLLADA document.

```

<library_kinematics_scenes>
  <kinematics_scene id="robot_kinscene">
    <instance_articulated_system sid="inst_robot_motion" url="#robot_motion">
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int1_axis0">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t1_axis0" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int1_value">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t1_value" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int2_axis0">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t2_axis0" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int2_value">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t2_value" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int3_axis0">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t3_axis0" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int3_value">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t3_value" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int4_axis0">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t4_axis0" />
      </bind>
      <bind symbol="robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_jo
int4_value">
        <param ref="robot_motion/robot_motion_robot_kinematics_inst_robot_kinmodel_inst_join
t4_value" />
      </bind>
    </instance_articulated_system>
  </kinematics_scene>
</library_kinematics_scenes>

```

Figure B.19 – Definition of the kinematic scene

B.3.7 Assembling of the scene

In the element “scene” all kinematic scenes are instantiated that are used. In addition the linking to the visual scene is done here, as depicted in Figure B.20. This linking defines the flow of the joint value from a kinematic calculation to the transformation in the geometry.

```

<scene>
  <instance_visual_scene url="#visualscene" />
  <instance_kinematics_scene sid="inst_robot_kinscene" url="#robot_kinscene">
    <bind_kinematics_model node="robot">
      <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel</param>
    </bind_kinematics_model>
    <bind_joint_axis target="joint1/transform">
      <axis>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint1_
axis0</param>
      </axis>
      <value>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint1_
value</param>
      </value>
    </bind_joint_axis>
    <bind_joint_axis target="joint2/transform">
      <axis>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint2_
axis0</param>
      </axis>
      <value>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint2_
value</param>
      </value>
    </bind_joint_axis>
    <bind_joint_axis target="joint3/transform">
      <axis>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint3_
axis0</param>
      </axis>
      <value>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint3_
value</param>
      </value>
    </bind_joint_axis>
    <bind_joint_axis target="joint4/transform">
      <axis>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint4_
axis0</param>
      </axis>
      <value>
        <param>robot_kinscene_robot_motion_robot_kinematics_inst_robot_kinmodel_inst_joint4_
value</param>
      </value>
    </bind_joint_axis>
  </instance_kinematics_scene>
</scene>

```

Figure B.20 – Instantiation of the kinematic scene

B.3.8 Combination of CAEX and COLLADA into AML

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with a single InternalElement “SCARA Robot”. The latter one includes an ExternalInterface named “FileLinkExplicit” derived from the InterfaceClass “COLLADAInterface”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib”. This interface is used to reference the complete scene explicitly and in particular the robot as depicted in Figure B.21.

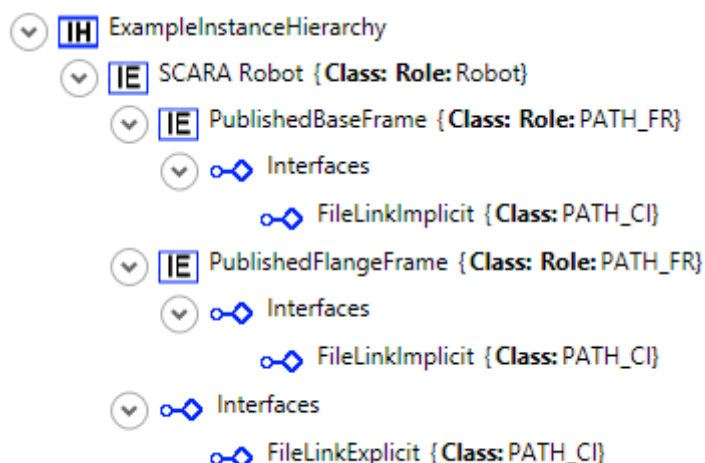


Figure B.21 – Hierarchy of the AML document

The InternalElement “SCARA Robot” has two additional InternalElements named “PublishedBaseFrame” and “PublishedFlangeFrame” with assigned RoleClass “Frame”. Both InternalElements reference kinematic elements in the COLLADA document implicitly. Figure B.22 shows the relevant part of the AML document.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="SCARA Robot" ID="GUID7">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID8">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./robot.dae</Value>
      </Attribute>
    </ExternalInterface>
    <InternalElement Name="PublishedBaseFrame" ID="GUID9">
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID10">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./robot.dae#robot_kinematics</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./baseframe</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
    </InternalElement>
    <InternalElement Name="PublishedFlangeFrame" ID="GUID11">
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID12">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./robot.dae#robot_kinematics</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./flangeframe</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_ROB" />
  </InternalElement>
</InstanceHierarchy>
  
```

Figure B.22 – XML representation of the AML document

Each published frame references unambiguously the “articulated_system” element within the COLLADA document by the use of the relative URI “./robot.dae#robot_kinematics”. The target elements “frame_tip” and “frame_origin” (see Figure B.22) are resolved by browsing from the

entry point by the use of a relative path “./flangeframe” and “./baseframe” according to the rules of a SID.

B.4 Modelling an AML document of a combined system including a robot and a linear axis in CAEX and COLLADA

Clause B.4 demonstrates how to attach a robot to a linear unit. Therefore, the definition of the robot is taken from Clause B.3 and the definition of the linear unit is taken from Clause B.2. In this example, both definitions are extended by additional elements needed for the attachment which is realized in the top level AML document. In general, there are different ways how to specify the combined system as depicted in Figure B.23. Firstly, robot and linear unit may be defined in separate COLLADA documents and the offset transformation for the attachment may be defined in the AML document. Secondly, a combined main COLLADA document is used to reference both the robot and the linear unit. In the latter case, the offset transformation for the attachment is defined in the main COLLADA document. The example of this clause describes the first case.

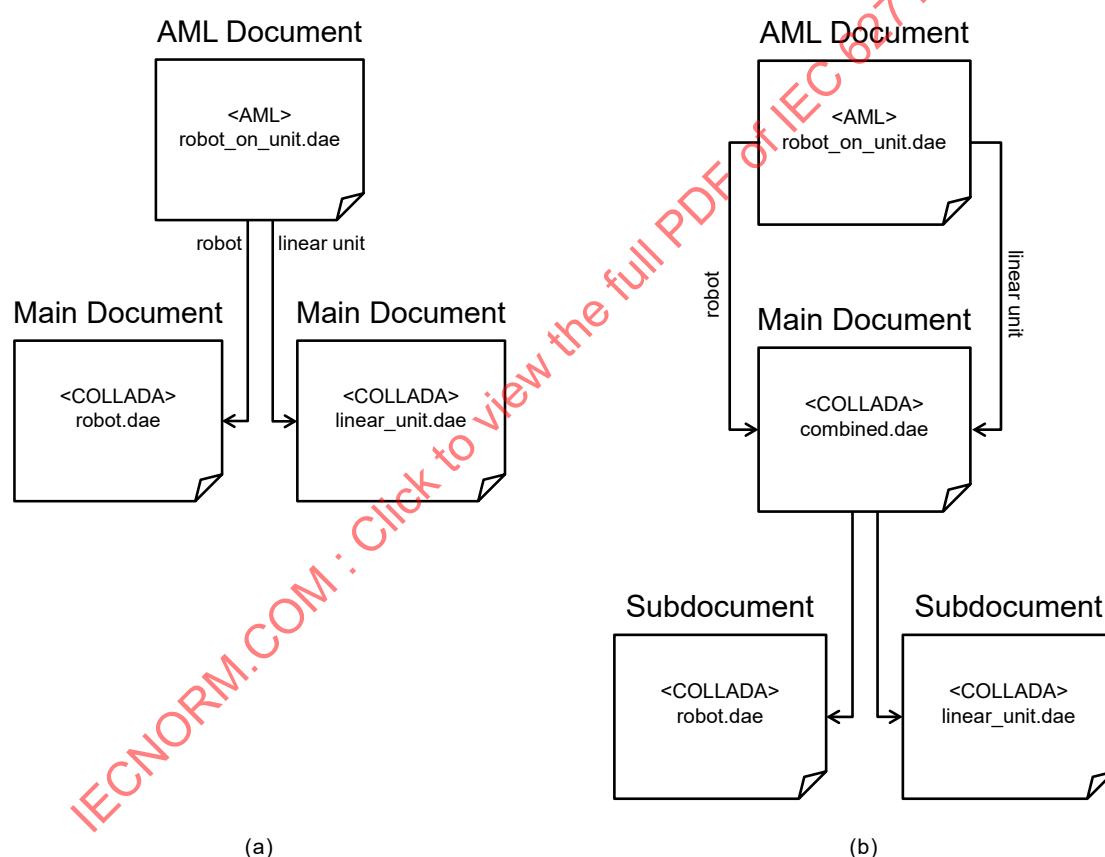


Figure B.23 – Structure of referencing COLLADA subdocuments

The AML document consists of an InstanceHierarchy named “ExampleInstanceHierarchy” with a root InternalElement named “RobotOnLinearUnit”. This InternalElement has two InternalElements (“LinearUnit”, “SCARA Robot”). In this example the robot is attached to the linear unit. Both InternalElements include an ExternalInterface named “FileLinkExplicit” derived from the InterfaceClass “COLLADAInterface”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib”. Each ExternalInterface “FileLinkExplicit” of both InternalElements points to a separate COLLADA document defined in Clause B.2 and Clause B.3 and addresses the complete COLLADA scene explicitly. In addition both InternalElements have a child element (“PublishedBaseFrame”, “PublishedFlangeFrame”) with applied RoleClass “Frame”, which is part of the standard RoleClassLibrary as well. The InternalElements “PublishedBaseFrame” and

“PublishedFlangeFrame” have two ExternalInterfaces: an ExternalInterface derived from the InterfaceClass “COLLADAInterface” to reference a COLLADA node implicitly (“FileLinkImplicit”) and an ExternalInterface derived from the InterfaceClass “AttachmentInterface” to define an attachment (“Attachment”). The structure is depicted in Figure B.24.

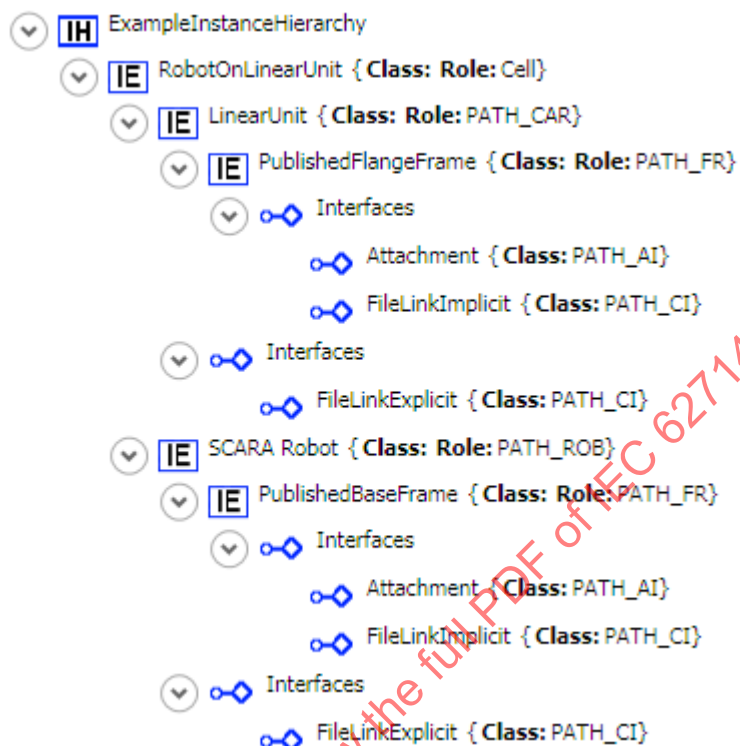


Figure B.24 – Hierarchy of the AML document

Figure B.25 shows the relevant part of the AML document and how to publish frames for attaching the robot to the linear unit, which are defined in AML.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RobotOnLinearUnit" ID="GUID0">
    <InternalElement Name="LinearUnit" ID="GUID1">
      <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>explicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./linear_unit.dae</Value>
        </Attribute>
      </ExternalInterface>
    <InternalElement Name="PublishedFlangeFrame" ID="GUID3">
      <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID4" />
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID5">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./linear_unit.dae#linearunit_kinematics</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./flangeframe</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_CAR" />
  </InternalElement>
  <InternalElement Name="SCARA Robot" ID="GUID6">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID7">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./robot.dae</Value>
      </Attribute>
    </ExternalInterface>
    <InternalElement Name="PublishedBaseFrame" ID="GUID8">
      <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID9" />
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID10">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./robot.dae#robot_kinematics</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./baseframe</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_ROB" />
  </InternalElement>
</InternalElement>
</InstanceHierarchy>

```

Figure B.25 – XML representation of the AML document

Finally, the connection is established by the use of an InternalLink, which connects both published frames. Figure B.26 shows the relevant part of the AML document and how to define an InternalLink. The link is stored in the InternalElement "RobotOnLinearUnit", which is the lowest common parent of the connected AML objects according to IEC 62714-1.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  [...]
  <InternalElement Name="RobotOnLinearUnit" ID="GUID0">
    <InternalLink Name="AttachmentLinearUnitMovesRobot"
      RefPartnerSideA="GUID3:Attachment"
      RefPartnerSideB="GUID8:Attachment" />
  </InternalElement>
</InstanceHierarchy>

```

Figure B.26 – XML representation of the AML document

Figure B.27 depicts the resulting scene with the robot attached to the linear unit.

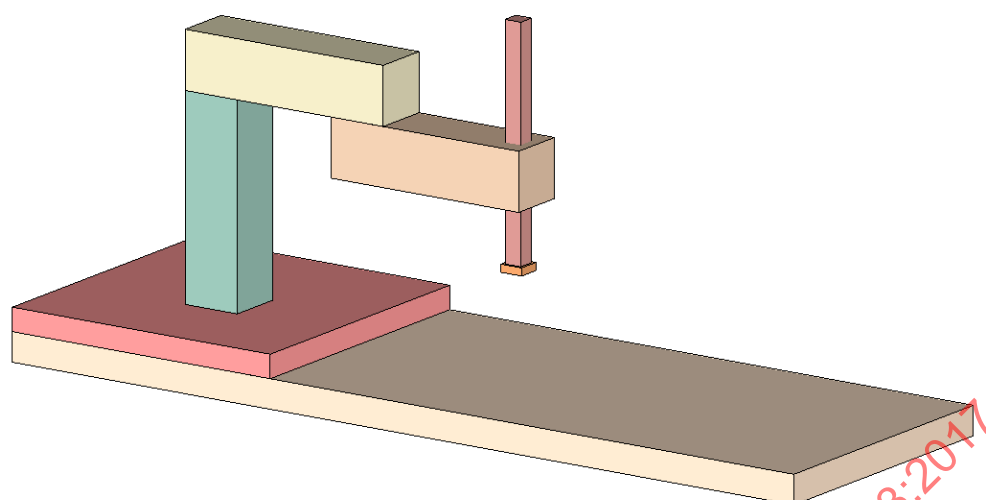


Figure B.27 – Visualization of the robot attached to the linear unit

B.5 Modelling an AML document of a gripper connected to robot in CAEX and COLLADA

B.5.1 General

Clause B.5 shows how to model a gripping device which represents a simplified two jaw parallel gripper. To open or close the gripper by means of moving the fingers, a kinematic system will be applied to the gripper. To achieve a parallel movement of both fingers, the first finger can be moved actively and the second finger will be dependent on the first one. This dependency and the direction of the movement will be expressed using MathML.

In this example, three COLLADA documents are used: a main document and two subdocuments. The geometry and kinematic definition is separated into two subdocuments. The main document defines the scene and combines the geometry and kinematic representation. Figure B.28 depicts the hierarchy of the documents.

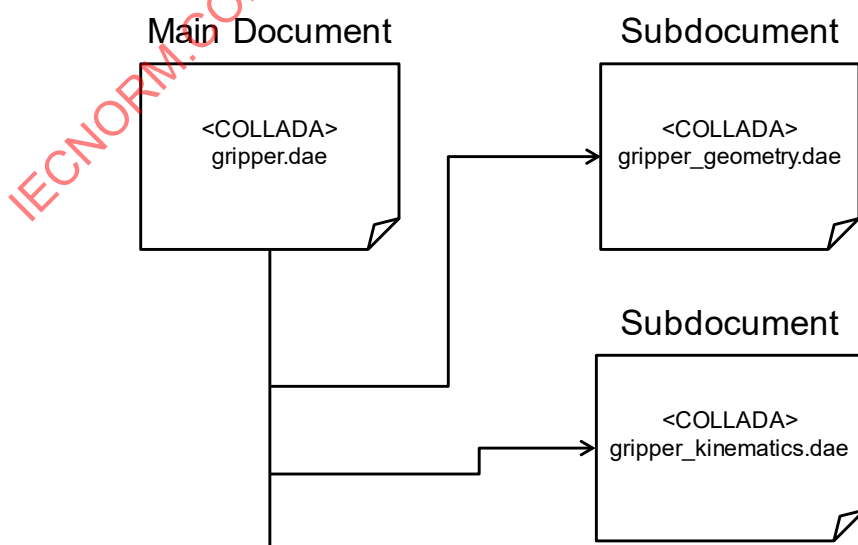


Figure B.28 – Hierarchy of COLLADA documents

According to Clauses B.2 and B.3, B.5.2 describes the visual scene, which is defined in the subdocument “gripper_geometry.dae”. In B.5.3, the kinematics scene is defined, which is part of the subdocument “gripper_kinematics”. The combination of both the geometry and

kinematic and in particular the definition of the scene is defined in the main document “gripper.dae” and is subject of B.5.3.6. In B.5.5, the combination of CAEX and COLLADA into AML is shown.

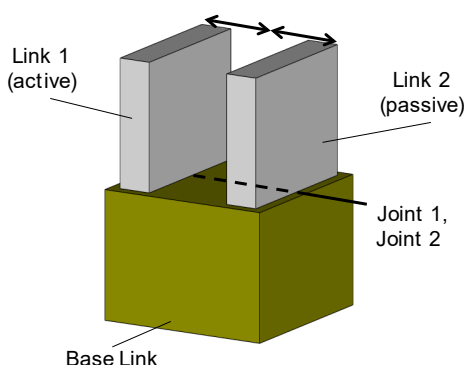


Figure B.29 – Visualization of the gripper

B.5.2 Definition of the visual scene

The visual scene of the gripper consists of three geometric elements: a gripping module (“BaseLink”) and two fingers (“Link1”, “Link2”) as depicted in Figure B.29. To keep the COLLADA content to a minimum, the geometries of the gripper are represented by rectangular cuboids, which are defined within the COLLADA document “cube.dae” as introduced in Figure A.7. Figure B.30 depicts the content of the COLLADA document “gripper_geometry.dae” which defines the geometric representation of the gripper.

```
<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the node hierarchy of a gripper with two
jaws</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="base-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0.5 0.5 0 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="link-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0.75 0.75 0.75 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="base-mat" name="BaseMat">
      <instance_effect url="#base-fx" />
    </material>
    <material id="link-mat" name="LinkMat">
      <instance_effect url="#link-fx" />
    </material>
  </library_materials>
</COLLADA>
```

```

<library_visual_scenes>
  <visual_scene id="visualscene">
    <node id="gripper" name="Gripper">
      <node id="geombase" name="GeomBase">
        <!-- the base link of the gripper -->
        <translate>-0.02 -0.02 0</translate>
        <scale>0.04 0.04 0.03</scale>
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#base-mat" />
            </technique_common>
          </bind_material>
        </instance_geometry>
        <!-- the first (active) link of the gripper -->
        <node id="geomlink1" name="GeomLink1">
          <scale>25 25 33.3333</scale>
          <translate>0.0025 0.0025 0.03</translate>
          <scale>0.0075 0.035 0.03</scale>
          <instance_geometry url="./cube.dae#cube-geom">
            <bind_material>
              <technique_common>
                <instance_material symbol="mat" target="#link-mat" />
              </technique_common>
            </bind_material>
          </instance_geometry>
        </node>
        <!-- the second (passive) link of the gripper -->
        <node id="geomlink2" name="GeomLink2">
          <scale>25 25 33.3333</scale>
          <translate>0.03 0.0025 0.03</translate>
          <scale>0.0075 0.035 0.03</scale>
          <instance_geometry url="./cube.dae#cube-geom">
            <bind_material>
              <technique_common>
                <instance_material symbol="mat" target="#link-mat" />
              </technique_common>
            </bind_material>
          </instance_geometry>
        </node>
      </node>
    </node>
  </visual_scene>
</library_visual_scenes>
</COLLADA>

```

Figure B.30 – Definition of the visual scene

B.5.3 Definition of the kinematic system

B.5.3.1 General

In Subclause B.5.3, the kinematic system will be defined. According to Clauses B.2 or Clause B.3, the steps include the definition of joints, the kinematic model, articulated system, and the kinematic scene. The exact procedure is not explained any more in detail as it is already explained in Clause B.2 and Clause B.3. Figure B.31 shows the template of the COLLADA document "gripper.dae". Missing elements are denoted by XML comments and will be defined in the subsequent subclauses.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the kinematics of a gripper with two jaws</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_joints>
    <!-- Step 1: Definition of joint elements -->
  </library_joints>
  <library_kinematics_models>
    <!-- Step 2: Definition of kinematic model element -->
  </library_kinematics_models>
  <library_articulated_systems>
    <!-- Step 3: Definition of articulated system (kinematics) element -->
    <!-- Step 4: Definition of articulated system (motion) element -->
  </library_articulated_systems>
  <library_kinematics_scenes>
    <!-- Step 5: Definition of kinematics scene element -->
  </library_kinematics_scenes>
  <library_formulas>
    <!-- Step 6: Definition of formula elements (joint dependencies) -->
  </library_formulas>
</COLLADA>

```

Figure B.31 – Definition of the kinematics

B.5.3.2 Definition of joints

In this example, the kinematic system has two prismatic joints (“Joint1”, “Joint2”) which are defined in a library and which will be instantiated in a subsequent step. “Joint1” will represent the active, leading joint while “Joint2” will represent the passive, dependent one. The extract of the COLLADA document is shown in Figure B.32.

```

<joint id="joint1" name="Joint1">
  <prismatic sid="axis0">
    <axis>0 0 1</axis>
    <limits>
      <min>0</min>
      <max>3</max>
    </limits>
  </prismatic>
</joint>
<joint id="joint2" name="Joint2">
  <prismatic sid="axis0">
    <axis>0 0 1</axis>
    <limits>
      <min>0</min>
      <max>3</max>
    </limits>
  </prismatic>
</joint>

```

Figure B.32 – Definition of joints

B.5.3.3 Definition of the kinematic model

In Subclause B.5.3.3, the kinematic model and its hierarchy is defined. Firstly all joints are defined or instantiated from a joints library. The following kinematic chain consists of links and joints and their hierarchical relationship. Any link or joint may provide a transformation. The transformation of a link or a joint is always relative to its predecessor and is described through the elements “translate” and “rotate”. The kinematic model is defined within a library and will be instantiated in a subsequent step. Figure B.33 shows the relevant part of the COLLADA document.

```

<kinematics_model id="gripper_kinmodel">
  <technique common>
    <instance_joint sid="inst_joint1" url="#joint1" />
    <instance_joint sid="inst_joint2" url="#joint2" />
    <link name="BaseLink" sid="baselink">
      <attachment_full joint="gripper_kinmodel/inst_joint1">
        <rotate>0 1 0 90</rotate>
        <link name="Link1" sid="link1" />
      </attachment_full>
      <attachment_full joint="gripper_kinmodel/inst_joint2">
        <rotate>0 1 0 90</rotate>
        <link name="Link2" sid="link2" />
      </attachment_full>
    </link>
  </technique_common>
</kinematics_model>

```

Figure B.33 – Definition of kinematic model

B.5.3.4 Definition of the articulated system

Kinematic and dynamic properties are assembled within “articulated_system” elements. Since each articulated system definition only contains either kinematic or dynamic information, two separate “articulated_system” systems are defined within a library.

The kinematic section of an articulated system contains information about the kinematic behaviour of an articulated model, for instance activity of joints, limits or dependencies between joints. Figure B.34 shows the relevant part of the COLLADA document and its additional kinematic parameters used in this example.

```

<articulated_system id="gripper_kinematics">
  <kinematics>
    <instance_kinematics_model sid="inst_gripper_kinmodel" url="#gripper_kinmodel">
      <newparam sid="gripper_kinematics_inst_gripper_kinmodel">
        <SIDREF>gripper_kinematics/inst_gripper_kinmodel</SIDREF>
      </newparam>
      <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint1_axis0">
        <SIDREF>gripper_kinematics/inst_gripper_kinmodel/inst_joint1/axis0</SIDREF>
      </newparam>
      <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint1_value">
        <float>0.0</float>
      </newparam>
      <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint2_axis0">
        <SIDREF>gripper_kinematics/inst_gripper_kinmodel/inst_joint2/axis0</SIDREF>
      </newparam>
      <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint2_value">
        <float>0.0</float>
      </newparam>
    </instance_kinematics_model>
    <technique_common>
      <axis_info axis="gripper_kinematics/inst_gripper_kinmodel/inst_joint1/axis0" sid="inst
joint1_info">
        <active>
          <bool>true</bool>
        </active>
        <locked>
          <bool>false</bool>
        </locked>
        <limits>
          <min>
            <float>0.0</float>
          </min>
          <max>
            <float>3.0</float>
          </max>
        </limits>
      </axis_info>
      <axis_info axis="gripper_kinematics/inst_gripper_kinmodel/inst_joint2/axis0" sid="inst
joint2_info">
        <active>
          <bool>false</bool>
        </active>
        <locked>
          <bool>false</bool>
        </locked>
        <limits>
          <min>
            <float>0.0</float>
          </min>
          <max>
            <float>3.0</float>
          </max>
        </limits>
      </axis_info>
      <frame_origin link="gripper_kinematics/inst_gripper_kinmodel/baselink" sid="baseframe"
/>
      <frame_tip link="gripper_kinematics/inst_gripper_kinmodel/link1" sid="flangeframe">
        <translate>1 2 3</translate>
      </frame_tip>
    </technique_common>
  </kinematics>
</articulated_system>

```

Figure B.34 – Definition of the articulated system

To provide dynamics information, an additional articulated system shall be defined. The motion section of an articulated system contains information about the dynamics behaviour of an articulated model, for instance velocities of joints, acceleration or deceleration of joints. Figure B.35 depicts the relevant part of the COLLADA document and its additional dynamics parameters used in this example.

```

<articulated_system id="gripper_motion">
  <motion>
    <instance_articulated_system sid="inst_gripper_kinematics" url="#gripper_kinematics">
      <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinematics_model">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel</SIDREF>
      </newparam>
      <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1_axis0">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint1_axis0</SIDREF>
      </newparam>
      <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1_value">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint1_value</SIDREF>
      </newparam>
      <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2_axis0">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint2_axis0</SIDREF>
      </newparam>
      <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2_value">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint2_value</SIDREF>
      </newparam>
    </instance_articulated_system>
    <technique_common>
      <axis_info axis="gripper_kinematics/inst_joint1_info">
        <speed>
          <float>0.0</float>
        </speed>
        <acceleration>
          <float>0.0</float>
        </acceleration>
        <deceleration>
          <float>0.0</float>
        </deceleration>
        <jerk>
          <float>0.0</float>
        </jerk>
      </axis_info>
      <axis_info axis="gripper_kinematics/inst_joint2_info">
        <speed>
          <float>0.0</float>
        </speed>
        <acceleration>
          <float>0.0</float>
        </acceleration>
        <deceleration>
          <float>0.0</float>
        </deceleration>
        <jerk>
          <float>0.0</float>
        </jerk>
      </axis_info>
    </technique_common>
  </motion>
</articulated_system>

```

Figure B.35 – Definition of the articulated system

B.5.3.5 Definition of the kinematic scene

A kinematic scene assembles different kinematic systems. All parameters needed for the whole scene are published within a kinematic scene. In this example the kinematic scene instantiates the articulated system which has been defined in B.5.3.4. Additionally, the kinematic model, the joint axis, and its values are published. Figure B.36 shows the relevant part of the COLLADA document.

```

<kinematics_scene id="gripper_kinscene">
  <instance_articulated_system sid="inst_gripper_motion" url="#gripper_motion">
    <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinematics_model">
      <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinematics_model" />
    </bind>
    <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1_axis0">
      <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1_axis0" />
    </bind>
    <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1_value">
      <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1_value" />
    </bind>
    <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2_axis0">
      <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2_axis0" />
    </bind>
    <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2_value">
      <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2_value" />
    </bind>
  </instance_articulated_system>
</kinematics_scene>

```

Figure B.36 – Definition of the kinematic scene

B.5.3.6 Definition of the joint dependency using MathML

For opening and closing the gripper, both fingers need to move towards one another simultaneously. To move the passive joint “Joint2” when moving the active joint “Joint1” a dependency is required, that is modelled using a mathematical formula. Formula (B.1) shows the according formula to define the value on “Joint2” depending on the value of “Joint1”.

$$Joint2(Joint1) = (-1) * Joint1 \quad (B.1)$$

The sign depends on the orientation of the joint defined in the kinematic model. Figure B.37 depicts the relevant MathML section to define a formula within the COLLADA document. First, the target is defined based on the SID “gripper_kinmodel/inst_joint2”. Using the SID of “Joint1” within the formula, the value of the active joint “Joint1” is applied to the passive joint “Joint2” according to the mathematical expression.

```

<formula sid="front_dir.formula">
  <target>
    <param>gripper_kinmodel/inst_joint2</param>
  </target>
  <technique_common>
    <mathml:math>
      <mathml:apply>
        <mathml:times />
        <mathml:cn type="integer">-1</mathml:cn>
        <mathml:ci>gripper_kinmodel/inst_joint1</mathml:ci>
      </mathml:apply>
    </mathml:math>
  </technique_common>
</formula>

```

Figure B.37 – Definition of the joint dependency using MathML

B.5.3.7 Final COLLADA document

Figure B.38 depicts the resulting COLLADA document “gripper_kinematics.dae”.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the kinematics of a gripper with two jaws</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_joints>
    <!-- Step 1: Definition of joint elements -->
    <joint id="joint1" name="Joint1">
      <prismatic sid="axis0">
        <axis>0 0 1</axis>
        <limits>
          <min>0</min>
          <max>3</max>
        </limits>
      </prismatic>
    </joint>
    <joint id="joint2" name="Joint2">
      <prismatic sid="axis0">
        <axis>0 0 1</axis>
        <limits>
          <min>0</min>
          <max>3</max>
        </limits>
      </prismatic>
    </joint>
  </library_joints>
  <library_kinematics_models>
    <!-- Step 2: Definition of kinematic model element -->
    <kinematics_model id="gripper_kinmodel">
      <technique_common>
        <instance_joint sid="inst_joint1" url="#joint1" />
        <instance_joint sid="inst_joint2" url="#joint2" />
        <link name="BaseLink" sid="baselink">
          <attachment_full joint="gripper_kinmodel/inst_joint1">
            <rotate>0 1 0 90</rotate>
            <link name="Link1" sid="link1" />
          </attachment_full>
          <attachment_full joint="gripper_kinmodel/inst_joint2">
            <rotate>0 1 0 90</rotate>
            <link name="Link2" sid="link2" />
          </attachment_full>
        </link>
      </technique_common>
    </kinematics_model>
  </library_kinematics_models>
  <library_articulated_systems>
    <!-- Step 3: Definition of articulated system (kinematics) element -->
    <articulated_system id="gripper_kinematics">
      <kinematics>
        <instance_kinematics_model sid="inst_gripper_kinmodel" url="#gripper_kinmodel">
          <newparam sid="gripper_kinematics_inst_gripper_kinmodel">
            <SIDREF>gripper_kinematics/inst_gripper_kinmodel</SIDREF>
          </newparam>
          <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint1_axis0">
            <SIDREF>gripper_kinematics/inst_gripper_kinmodel/inst_joint1/axis0</SIDREF>
          </newparam>
          <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint1_value">
            <float>0.0</float>
          </newparam>
          <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint2_axis0">
            <SIDREF>gripper_kinematics/inst_gripper_kinmodel/inst_joint2/axis0</SIDREF>
          </newparam>
          <newparam sid="gripper_kinematics_inst_gripper_kinmodel_inst_joint2_value">
            <float>0.0</float>
          </newparam>
        </instance_kinematics_model>
        <technique_common>
          <axis_info axis="gripper_kinematics/inst_gripper_kinmodel/inst_joint1/axis0" sid="
inst_joint1_info">
            <active>
              <bool>true</bool>
            </active>
            <locked>
              <bool>false</bool>
            </locked>

```

```

        <limits>
        <min>
        <float>0.0</float>
        </min>
        <max>
        <float>3.0</float>
        </max>
        </limits>
        </axis_info>
        <axis_info axis="gripper_kinematics/inst_gripper_kinmodel/inst_joint2/axis0" sid="
inst_joint2_info">
        <active>
        <bool>>false</bool>
        </active>
        <locked>
        <bool>>false</bool>
        </locked>
        <limits>
        <min>
        <float>0.0</float>
        </min>
        <max>
        <float>3.0</float>
        </max>
        </limits>
        </axis_info>
        <frame_origin link="gripper_kinematics/inst_gripper_kinmodel/baselink" sid="basefr
ame" />
        <frame_tip link="gripper_kinematics/inst_gripper_kinmodel/link1" sid="flangeframe"
>
        <translate>1 2 3</translate>
        </frame_tip>
        </technique_common>
        </kinematics>
        </articulated_system>
        <!-- Step 4: Definition of articulated system (motion) element -->
        <articulated_system id="gripper_motion">
        <motion>
        <instance_articulated_system sid="inst_gripper_kinematics" url="#gripper_kinematics"
>
        <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinematics_model">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel</SIDREF>
        </newparam>
        <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1
axis0">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint1_
axis0</SIDREF>
        </newparam>
        <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint1
value">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint1_
value</SIDREF>
        </newparam>
        <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2
axis0">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint2_
axis0</SIDREF>
        </newparam>
        <newparam sid="gripper_motion_gripper_kinematics_inst_gripper_kinmodel_inst_joint2
value">
        <SIDREF>gripper_kinematics/gripper_kinematics_inst_gripper_kinmodel_inst_joint2_
value</SIDREF>
        </newparam>
        </instance_articulated_system>
        </technique_common>
        <axis_info axis="gripper_kinematics/inst_joint1_info">
        <speed>
        <float>0.0</float>
        </speed>
        <acceleration>
        <float>0.0</float>
        </acceleration>
        <deceleration>
        <float>0.0</float>
        </deceleration>
        <jerk>
        <float>0.0</float>
        </jerk>
        </axis_info>
        <axis_info axis="gripper_kinematics/inst_joint2_info">
        <speed>
        <float>0.0</float>
        </speed>

```

```

        <acceleration>
          <float>0.0</float>
        </acceleration>
        <deceleration>
          <float>0.0</float>
        </deceleration>
        <jerk>
          <float>0.0</float>
        </jerk>
      </axis_info>
    </technique_common>
  </motion>
</articulated_system>
</library_articulated_systems>
<library_kinematics_scenes>
  <!-- Step 5: Definition of kinematics scene element -->
  <kinematics_scene id="gripper_kinscene">
    <instance_articulated_system sid="inst_gripper_motion" url="#gripper_motion">
      <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinemat
tics_model">
        <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinemat
ics_model" />
      </bind>
      <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint1_axis0">
        <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint1_axis0" />
      </bind>
      <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint1_value">
        <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint1_value" />
      </bind>
      <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint2_axis0">
        <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint2_axis0" />
      </bind>
      <bind symbol="gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint2_value">
        <param ref="gripper_motion/gripper_motion_gripper_kinematics_inst_gripper_kinmod
el_inst_joint2_value" />
      </bind>
    </instance_articulated_system>
  </kinematics_scene>
</library_kinematics_scenes>
<library_formulas>
  <!-- Step 6: Definition of formula elements (joint dependencies) -->
  <formula sid="front_dir_formula">
    <target>
      <param>gripper_kinmodel/inst_joint2</param>
    </target>
    <technique_common>
      <mathml:math>
        <mathml:apply>
          <mathml:times />
          <mathml:cn type="integer">-1</mathml:cn>
          <mathml:ci>gripper_kinmodel/inst_joint1</mathml:ci>
        </mathml:apply>
      </mathml:math>
    </technique_common>
  </formula>
</library_formulas>
</COLLADA>

```

Figure B.38 – XML representation of the COLLADA document gripper_kinematics.dae

B.5.4 Assembling of the scene

In the element “scene”, all kinematic scenes are instantiated that are used. In addition the linking to the visual scene is done here, as depicted in Figure B.39. The scene in this example instantiates elements from the subdocuments “gripper_geometry.dae” and “gripper_kinematics.dae” as defined in B.5.2 and B.5.3. This linking defines the flow of the joint value from a kinematic calculation to the transformation in the geometry.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the combination of the gripper's geometry and
kinematics</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <scene>
    <instance_visual_scene url="./gripper_geometry.dae#visualscene" />
    <instance_kinematics_scene sid="inst_linearunit_kinscene" url="./gripper_kinematics.dae#
gripper_kinscene">
      <bind_kinematics_model node="gripper">
        <param>gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinematics_mo
del</param>
      </bind_kinematics_model>
      <bind_joint_axis target="geomlink1/transform">
        <axis>
          <param>gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_in
st_joint1_axis0</param>
        </axis>
        <value>
          <param>gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_in
st_joint1_value</param>
        </value>
      </bind_joint_axis>
      <bind_joint_axis target="geomlink1/transform">
        <axis>
          <param>gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_in
st_joint2_axis0</param>
        </axis>
        <value>
          <param>gripper_kinscene_gripper_motion_gripper_kinematics_inst_gripper_kinmodel_in
st_joint2_value</param>
        </value>
      </bind_joint_axis>
    </instance_kinematics_scene>
  </scene>
</COLLADA>

```

Figure B.39 – XML representation of the COLLADA document gripper.dae

B.5.5 Combination of CAEX and COLLADA into AML

This example extends the AML document from Clause B.4. In addition to the existing InternalElements “LinearUnit” and “SCARA Robot” the AML document has an InternalElement “JawGripper” including an ExternalInterface named “FileLinkExplicit” derived from the InterfaceClass “COLLADAInterface”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib”. This ExternalInterface is used to reference the complete scene explicitly and in particular the gripper as depicted in Figure B.40.

The InternalElement “SCARA Robot” has an additional child InternalElement “PublishedFlangeFrame” with applied RoleClass “Frame”. This InternalElement has two ExternalInterfaces: an ExternalInterface derived from the InterfaceClass “COLLADAInterface” to reference a COLLADA node implicitly (“FileLinkImplicit”) and an ExternalInterface derived from the InterfaceClass “AttachmentInterface” to define an attachment (“Attachment”).

The InternalElement “JawGripper” has two child InternalElements (“PublishedBaseFrame”, “PublishedFlangeFrame”) with applied RoleClass “Frame”, which is part of the standard RoleClassLibrary as well. The InternalElements “PublishedBaseFrame” and “PublishedFlangeFrame” have two ExternalInterfaces: an ExternalInterface derived from the InterfaceClass “COLLADAInterface” to reference a COLLADA node implicitly (“FileLinkImplicit”) and an ExternalInterface derived from the InterfaceClass “AttachmentInterface” to define an attachment (“Attachment”).

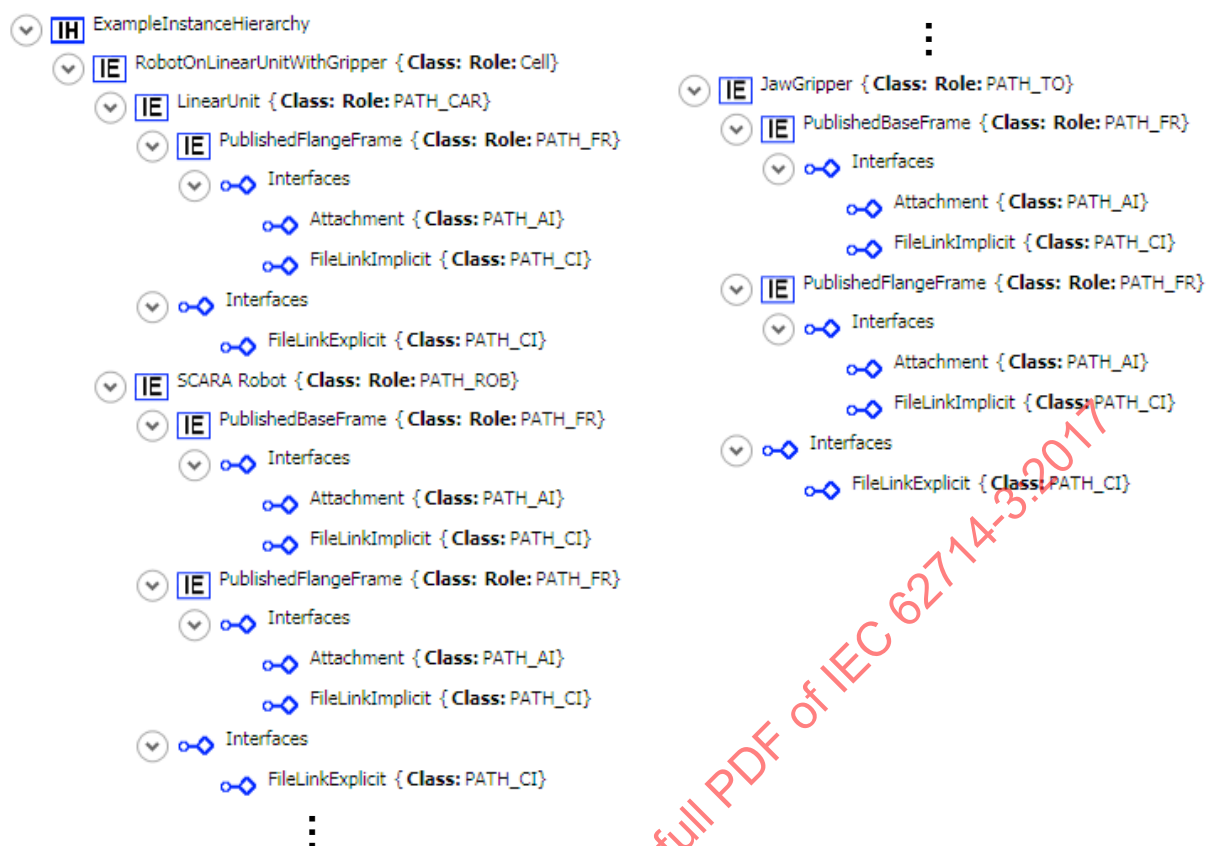


Figure B.40 – Hierarchy of the AML document

Figure B.41 shows the relevant part of the AML document and how to publish frames for attaching the gripper to the robot, which are defined in AML.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RobotOnLinearUnitWithGripper" ID="GUID0">
    <InternalElement Name="LinearUnit" ID="GUID1">
      <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>explicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./linear_unit.dae</Value>
        </Attribute>
      </ExternalInterface>
      <InternalElement Name="PublishedFlangeFrame" ID="GUID3">
        <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID4" />
        <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID5">
          <Attribute Name="refType" AttributeDataType="xs:string">
            <Value>implicit</Value>
          </Attribute>
          <Attribute Name="refURI" AttributeDataType="xs:anyURI">
            <Value>./linear_unit.dae#linearunit_kinematics</Value>
          </Attribute>
          <Attribute Name="target" AttributeDataType="xs:token">
            <Value>./flangeframe</Value>
          </Attribute>
        </ExternalInterface>
        <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
      </InternalElement>
      <RoleRequirements RefBaseRoleClassPath="PATH_CAR" />
    </InternalElement>
    <InternalElement Name="SCARA Robot" ID="GUID6">
      <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID7">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>explicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./robot.dae</Value>
        </Attribute>
      </ExternalInterface>
    </InternalElement>
  </InternalElement>
</InstanceHierarchy>
```

```

<InternalElement Name="PublishedBaseFrame" ID="GUID8">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID9" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID10">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./baseframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalElement Name="PublishedFlangeFrame" ID="GUID11">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID12" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID13">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./flangeframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalElement Name="JawGripper" ID="GUID14">
  <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID15">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>explicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./gripper.dae</Value>
    </Attribute>
  </ExternalInterface>
</InternalElement>
<InternalElement Name="PublishedBaseFrame" ID="GUID16">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID17" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID18">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./baseframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalElement Name="PublishedFlangeFrame" ID="GUID19">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID20" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID21">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./flangeframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalElement Name="PublishedFlangeFrame" ID="GUID19">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID20" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID21">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./flangeframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalElement Name="PublishedFlangeFrame" ID="GUID19">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID20" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID21">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./flangeframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalLink Name="AttachmentLinearUnitMovesRobot" RefPartnerSideA="GUID3:Attachment" RefPartnerSideB="GUID8:Attachment" />
<InternalLink Name="AttachmentRobotMovesGripper" RefPartnerSideA="GUID11:Attachment" RefPartnerSideB="GUID16:Attachment" />
</InternalElement>
</InstanceHierarchy>

```

Figure B.41 – XML representation of the AML document

Finally, the connection is established by the use of an InternalLink, which connects both published frames. Figure B.42 shows the relevant part of the AML document and how to define an InternalLink. The link is stored in the InternalElement “RobotOnLinearUnitWithGripper”, which is the lowest common parent of the connected AML objects according to IEC 62714-1.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RobotOnLinearUnitWithGripper" ID="GUID0">
    [...]
    <InternalLink Name="AttachmentLinearUnitMovesRobot"
      RefPartnerSideA="GUID3:Attachment"
      RefPartnerSideB="GUID8:Attachment" />
    <InternalLink Name="AttachmentRobotMovesGripper"
      RefPartnerSideA="GUID11:Attachment"
      RefPartnerSideB="GUID16:Attachment" />
  </InternalElement>
</InstanceHierarchy>
```

Figure B.42 – XML representation of the AML document

Figure B.43 depicts the resulting scene with the robot attached to the linear unit.

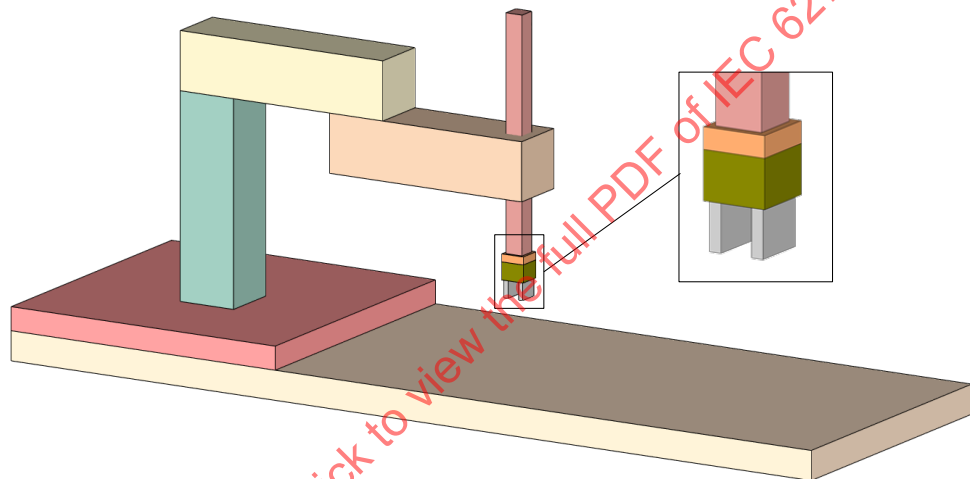


Figure B.43 – Visualization of the robot on a linear unit and attached gripper

B.6 Modelling an AML document of a work piece connected to a gripper in CAEX and COLLADA

B.6.1 General

Clause B.6 describes a specialized use case of a geometric attachment between a parent and a child AML object (both of type InternalElement). In general, any descendant geometry of an attached child AML object is transformed (moved or rotated) by the parent AML object due to the parent child relationship. If the attached child AML object references a COLLADA node whose geometry is defined in an ancestor COLLADA node, some additional requirements shall be met to move the geometry together with the attachment. In this clause, the example from Clauses B.2, B.3, B.4, and B.5 is supplemented by a further work object, which will be attached to the gripper defined in Clause B.5.

B.6.2 Implicit upper boundary

For a better understanding, Figure B.44 shows a simplified example of a robot with two frames (“A”, “B”) and a box with two frames (“C”, “D”). Frame “D” represents a child node of frame “C” and should be attached to frame “B” of the robot. The geometric shape of the box is defined in frame “C” and thus it is not explicitly part of the attachment.

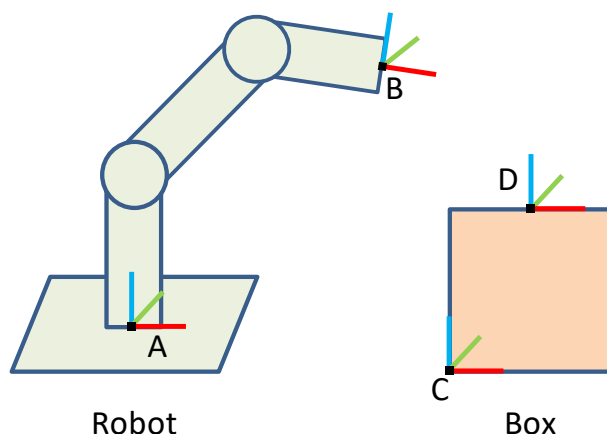


Figure B.44 – Example for implicit upper boundary

In this example, an AML and two COLLADA documents are used. The robot and the box are defined in separated COLLADA documents while the topology and the geometric attachment are defined in a top level AML document. The structure is depicted in Figure B.45.

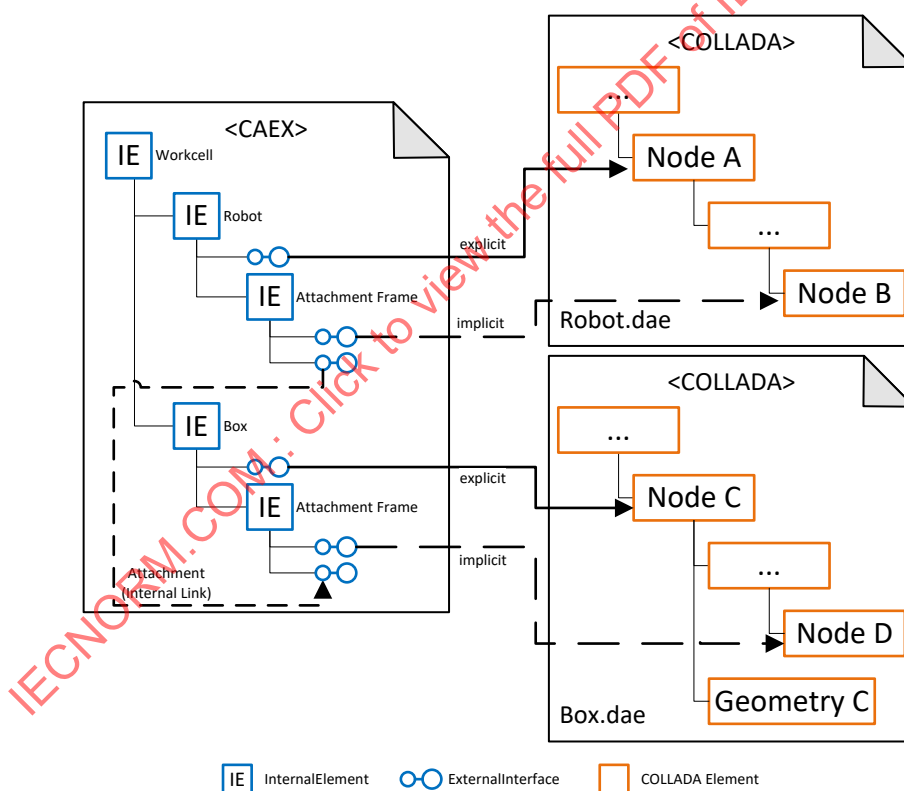


Figure B.45 – Structure for attachments between objects in CAEX

The AML document has two InternalElements (“Robot”, “Box”) by means of AML objects. Both AML objects have a reference to a COLLADA geometry object by using explicit references. The AML object “Robot” references the COLLADA document Robot.dae and the AML object “Box” references the COLLADA document Box.dae. Furthermore, both AML objects have a child InternalElement “Attachment Frame” with assigned RoleClass “Frame”. Each InternalElement “Attachment Frame” has an implicit ExternalInterface derived from the InterfaceClass “COLLADAInterface” which references a node defined in the COLLADA document. Additionally, each InternalElement “Attachment Frame” has ExternalInterface derived from the InterfaceClass “AttachmentInterface”, which may be connected or linked. In this example the implicit reference of the AML object “Box” addresses the COLLADA node

“Node D” which is a child of the COLLADA node “Node C”. The geometry of the box is defined in the scope of the COLLADA node “Node C”.

In accordance with the requirements for an attachment defined in Clause 8, this example is valid and the geometry of the box is moved with the attachment since it fulfils the requirements:

- The “AttachmentInterface” is child of an AML object with the RoleClass “Frame” and the object to be moved is the parent AML object “Box”.
- The AML object owning an “AttachmentInterface” has the RoleClass “Frame” and it has an ExternalInterface derived from the InterfaceClass “COLLADAInterface”. Its parent AML object “Box” has an ExternalInterface derived from the InterfaceClass “COLLADAInterface” as well. Applying this, the “COLLADAInterface” of the parent AML object represents the boundary of the geometry of the attached geometry.

In the subclauses B.6.3 and B.6.4, the example from Clauses B.2, B.3, B.4, and B.5 is continued. According to the definition of the box, a work piece is defined and attached to a gripper through an additional frame. The geometry of the work piece is defined in an ancestor COLLADA node.

B.6.3 Definition of the work piece

The work piece used in this example is defined in the COLLADA document “workpiece.dae”. To keep the COLLADA content to a minimum, the geometry of the work piece is represented by a rectangular cuboid, which is defined within the COLLADA document “cube.dae” as introduced in Figure A.7. The work piece has a different scaling and provides a custom material. Figure B.46 shows the visualization of the work piece (“WorkPiece”) and its additional frame.

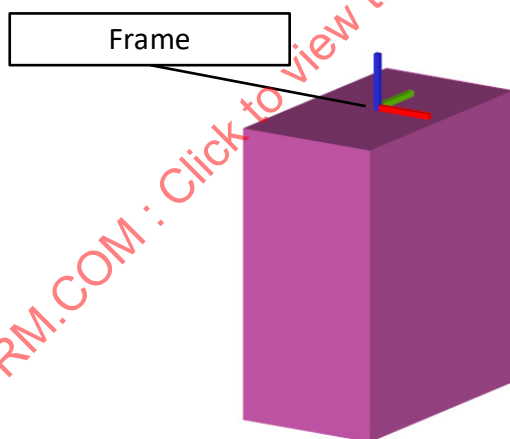


Figure B.46 – Visualization of the work piece with additional frame

The complete COLLADA document of the work piece (“workpiece.dae”) is shown in Figure B.47. It uses the geometry definition from the external COLLADA document “cube.dae” taken from Figure A.7.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the geometry of a work piece.
        An additional node representing a frame is provided.</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="wpeffect-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0.71 0.31 0.61 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="wpeffect" name="WorkPieceEffect">
      <instance_effect url="#wpeffect-fx" />
    </material>
  </library_materials>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="workpiece" name="WorkPiece">
        <scale>0.04 0.07 0.08</scale>
        <node sid="workpieceframe">
          <translate>0.5 0.5 1</translate>
        </node>
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#wpeffect" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#visualscene" />
  </scene>
</COLLADA>

```

Figure B.47 – COLLADA document of work piece with additional frame

B.6.4 Combination of CAEX and COLLADA into AML

This example extends the AML document from Clause B.5. In addition to the existing InternalElements “LinearUnit”, “SCARA Robot” and “JawGripper”, the AML document has an InternalElement “WorkPiece” including an ExternalInterface named “FileLinkExplicit” derived from the InterfaceClass “COLLADAInterface”, which is part of the standard InterfaceClassLibrary “AutomationMLInterfaceClassLib”. This ExternalInterface is used to reference the complete scene explicitly and in particular the work piece as depicted in Figure B.48.

The InternalElement “WorkPiece” has a child InternalElement (“PublishedFrame”) with applied RoleClass “Frame”, which is part of the standard RoleClassLibrary. The InternalElement “PublishedFrame” has an ExternalInterface derived from the InterfaceClass “COLLADAInterface” to reference a COLLADA node implicitly (“FileLinkImplicit”) and an ExternalInterface derived from the InterfaceClass “AttachmentInterface” to define an attachment (“Attachment”) as depicted in Figure B.48.

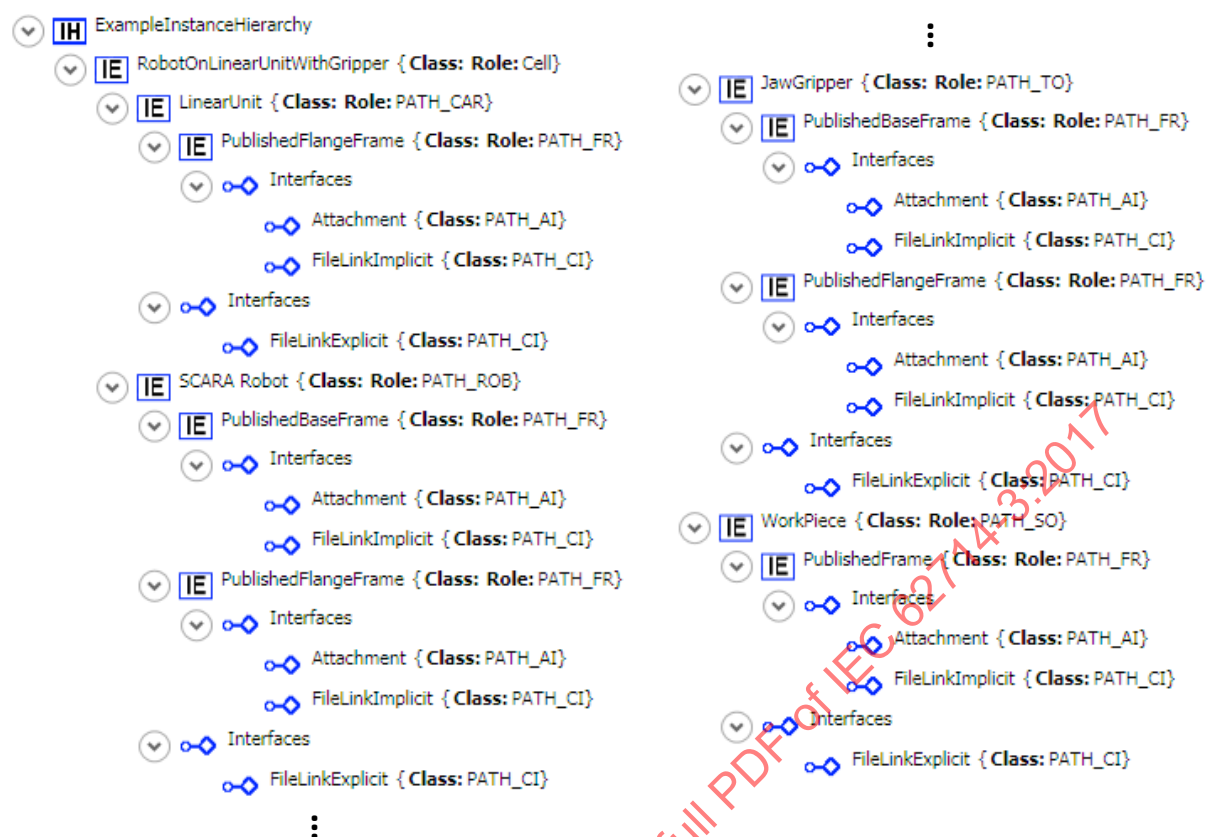


Figure B.48 – Hierarchy of the AML document

Figure B.49 shows the relevant part of the AML document.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RobotOnLinearUnitWithGripperAndWorkpiece" ID="GUID0">
    <InternalElement Name="LinearUnit" ID="GUID1">
      <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>explicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./linear_unit.dae</Value>
        </Attribute>
      </ExternalInterface>
      <InternalElement Name="PublishedFlangeFrame" ID="GUID3">
        <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID4" />
        <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID5">
          <Attribute Name="refType" AttributeDataType="xs:string">
            <Value>implicit</Value>
          </Attribute>
          <Attribute Name="refURI" AttributeDataType="xs:anyURI">
            <Value>./linear_unit.dae#linearunit_kinematics</Value>
          </Attribute>
          <Attribute Name="target" AttributeDataType="xs:token">
            <Value>./flangeframe</Value>
          </Attribute>
        </ExternalInterface>
        <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
      </InternalElement>
      <RoleRequirements RefBaseRoleClassPath="PATH_CAR" />
    </InternalElement>
    <InternalElement Name="SCARA Robot" ID="GUID6">
      <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID7">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>explicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./robot.dae</Value>
        </Attribute>
      </ExternalInterface>
      <InternalElement Name="PublishedBaseFrame" ID="GUID8">
```

```

<ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID9" />
<ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID10">
  <Attribute Name="refType" AttributeDataType="xs:string">
    <Value>implicit</Value>
  </Attribute>
  <Attribute Name="refURI" AttributeDataType="xs:anyURI">
    <Value>./robot.dae#robot_kinematics</Value>
  </Attribute>
  <Attribute Name="target" AttributeDataType="xs:token">
    <Value>./baseframe</Value>
  </Attribute>
</ExternalInterface>
<RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalElement Name="PublishedFlangeFrame" ID="GUID11">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID12" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID13">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./flangeframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<RoleRequirements RefBaseRoleClassPath="PATH_ROB" />
</InternalElement>
<InternalElement Name="JawGripper" ID="GUID14">
  <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID15">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>explicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./gripper.dae</Value>
    </Attribute>
  </ExternalInterface>
</InternalElement>
<InternalElement Name="PublishedBaseFrame" ID="GUID16">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID17" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID18">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./baseframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<InternalElement Name="PublishedFlangeFrame" ID="GUID19">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID20" />
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID21">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./robot.dae#robot_kinematics</Value>
    </Attribute>
    <Attribute Name="target" AttributeDataType="xs:token">
      <Value>./flangeframe</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<RoleRequirements RefBaseRoleClassPath="PATH_TO" />
</InternalElement>
<InternalElement Name="WorkPiece" ID="GUID22">
  <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID23">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>explicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./workpiece.dae</Value>
    </Attribute>
  </ExternalInterface>
</InternalElement>
<InternalElement Name="PublishedFrame" ID="GUID24">
  <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID25" />

```

```

<ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID26">
  <Attribute Name="refType" AttributeDataType="xs:string">
    <Value>implicit</Value>
  </Attribute>
  <Attribute Name="refURI" AttributeDataType="xs:anyURI">
    <Value>./workpiece.dae#workpiece</Value>
  </Attribute>
  <Attribute Name="target" AttributeDataType="xs:token">
    <Value>./workpieceframe</Value>
  </Attribute>
</ExternalInterface>
<RoleRequirements RefBaseRoleClassPath="PATH_FR" />
</InternalElement>
<RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InternalElement>
</InternalElement>
</InstanceHierarchy>

```

Figure B.49 – XML representation of the AML document

As a result, the work piece is attached to the gripper as depicted in Figure A.41. In opposite to mounting, the attaching does not include any additional transformation of the attached object.

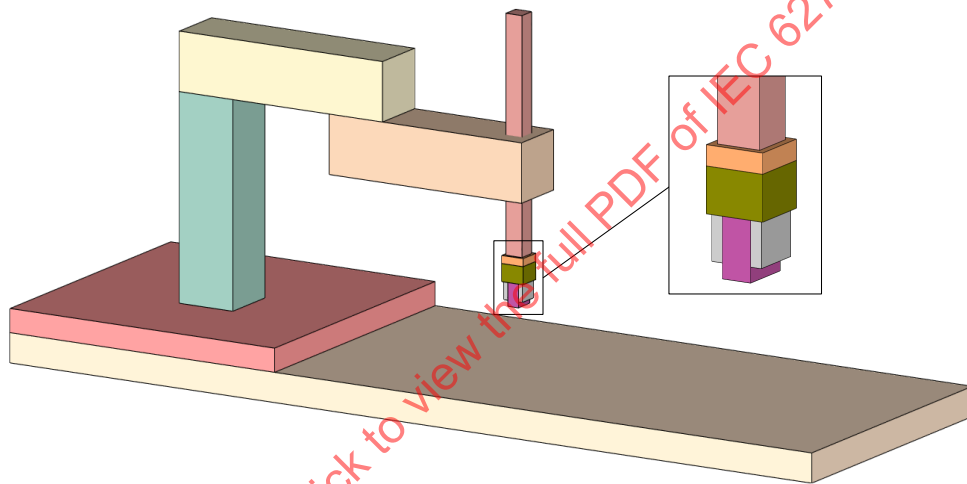


Figure B.50 – Attachment between geometric AML objects

Finally, the connection is established by the use of an InternalLink, which connects both published frames. Figure B.51 shows the relevant part of the AML document and how to define an InternalLink. The link is stored at the CAEX InternalElement “Workcell”, which is the lowest common parent of the connected CAEX objects according to IEC 62714-1.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RobotOnLinearUnitWithGripperAndWorkpiece" ID="GUID0">
    <InternalLink Name="AttachmentLinearUnitMovesRobot"
      RefPartnerSideA="GUID3:Attachment"
      RefPartnerSideB="GUID8:Attachment" />
    <InternalLink Name="AttachmentRobotMovesGripper"
      RefPartnerSideA="GUID11:Attachment"
      RefPartnerSideB="GUID16:Attachment" />
    <InternalLink Name="AttachmentGripperMovesWorkpiece"
      RefPartnerSideA="GUID19:Attachment"
      RefPartnerSideB="GUID24:Attachment" />
  </InternalElement>
</InstanceHierarchy>

```

Figure B.51 – XML representation of the AML document

Annex C (informative)

XML representation of AML libraries

C.1 AutomationMLBaseRoleClassLib

Figure C.1 shows the AML libraries AutomationMLBaseRoleClassLib which contains the RoleClass “Frame” specified in this document.

```
<CAEXFile xsi:noNamespaceSchemaLocation="CAEX_ClassModel_V2.15.xsd" FileName="AutomationMLBaseRoleClassLib.aml" SchemaVersion="2.15">
  <AdditionalInformation AutomationMLVersion="2.0" />
  <AdditionalInformation>
    <WriterHeader>
      <WriterName>AutomationML e.V.</WriterName>
      <WriterID>AutomationML e.V.</WriterID>
      <WriterVendor>AutomationML e.V.</WriterVendor>
      <WriterVendorURL>www.automationml.org</WriterVendorURL>
      <WriterVersion>1.0</WriterVersion>
      <WriterRelease>1.0.0</WriterRelease>
      <LastWritingDateTime>2013-03-01</LastWritingDateTime>
      <WriterProjectTitle>Automation Markup Language Standard Libraries</WriterProjectTitle>
      <WriterProjectID>Automation Markup Language Standard Libraries</WriterProjectID>
    </WriterHeader>
  </AdditionalInformation>
  <ExternalReference Path="AutomationMLInterfaceClassLib.aml" Alias="AutomationMLInterfaceClassLib" />
  <RoleClassLib Name="AutomationMLBaseRoleClassLib">
    <Description>Automation Markup Language Base Role class Library - Part 1 Content extended with Part 3 Content</Description>
    <Version>2.2.1</Version>
    <RoleClass Name="AutomationMLBaseRole">
      <RoleClass Name="Group" RefBaseClassPath="AutomationMLBaseRole">
        <Attribute Name="AssociatedFacet" AttributeDataType="xs:string" />
      </RoleClass>
      <RoleClass Name="Facet" RefBaseClassPath="AutomationMLBaseRole" />
      <RoleClass Name="Port" RefBaseClassPath="AutomationMLBaseRole">
        <Attribute Name="Direction" AttributeDataType="xs:string" />
        <Attribute Name="Cardinality">
          <Attribute Name="MinOccur" AttributeDataType="xs:unsignedInt" />
          <Attribute Name="MaxOccur" AttributeDataType="xs:unsignedInt" />
        </Attribute>
        <Attribute Name="Category" AttributeDataType="xs:string" />
        <ExternalInterface Name="ConnectionPoint" ID="9942bd9c-c19d-44e4-a197-11b9edf264e7" RefBaseClassPath="AutomationMLInterfaceClassLib@AutomationMLInterfaceClassLib/AutomationMLBaseInterface/PortConnector" />
      </RoleClass>
      <RoleClass Name="Resource" RefBaseClassPath="AutomationMLBaseRole" />
      <RoleClass Name="Product" RefBaseClassPath="AutomationMLBaseRole" />
      <RoleClass Name="Process" RefBaseClassPath="AutomationMLBaseRole" />
      <RoleClass Name="Structure" RefBaseClassPath="AutomationMLBaseRole">
        <RoleClass Name="ProductStructure" RefBaseClassPath="Structure" />
        <RoleClass Name="ProcessStructure" RefBaseClassPath="Structure" />
        <RoleClass Name="ResourceStructure" RefBaseClassPath="Structure" />
      </RoleClass>
      <RoleClass Name="PropertySet" RefBaseClassPath="AutomationMLBaseRole" />
      <RoleClass Name="Frame" RefBaseClassPath="AutomationMLBaseRole" />
    </RoleClass>
  </RoleClassLib>
</CAEXFile>
```

Figure C.1 – XML representation of AML libraries AutomationMLBaseRoleClassLib

C.2 AutomationMLInterfaceClassLib

Figure C.2 shows the AML libraries AutomationMLInterfaceClassLib which contains the InterfaceClass “COLLADAInterface” specified in this document.

```

<CAEXFile xsi:noNamespaceSchemaLocation="CAEX_ClassModel_V2.15.xsd" FileName="AutomationMLInterfaceClassLib.aml" SchemaVersion="2.15">
  <AdditionalInformation AutomationMLVersion="2.0" />
  <AdditionalInformation>
    <WriterHeader>
      <WriterName>AutomationML e.V.</WriterName>
      <WriterID>AutomationML e.V.</WriterID>
      <WriterVendor>AutomationML e.V.</WriterVendor>
      <WriterVendorURL>www.automationml.org</WriterVendorURL>
      <WriterVersion>1.0</WriterVersion>
      <WriterRelease>1.0.0</WriterRelease>
      <LastWritingDateTime>2013-03-01</LastWritingDateTime>
      <WriterProjectTitle>Automation Markup Language Standard Libraries</WriterProjectTitle>
      <WriterProjectID>Automation Markup Language Standard Libraries</WriterProjectID>
    </WriterHeader>
  </AdditionalInformation>
  <InterfaceClassLib Name="AutomationMLInterfaceClassLib">
    <Description>Standard Automation Markup Language Interface Class Library - Part 1 Content extended with Part 3 Content</Description>
    <Version>2.2.1</Version>
    <InterfaceClass Name="AutomationMLBaseInterface">
      <InterfaceClass Name="Order" RefBaseClassPath="AutomationMLBaseInterface">
        <Attribute Name="Direction" AttributeDataType="xs:string" />
      </InterfaceClass>
      <InterfaceClass Name="PortConnector" RefBaseClassPath="AutomationMLBaseInterface" />
      <InterfaceClass Name="InterlockingConnector" RefBaseClassPath="AutomationMLBaseInterface" />
      <InterfaceClass Name="PPRConnector" RefBaseClassPath="AutomationMLBaseInterface" />
      <InterfaceClass Name="ExternalDataConnector" RefBaseClassPath="AutomationMLBaseInterface">
        <Attribute Name="refURI" AttributeDataType="xs:anyURI" />
        <InterfaceClass Name="COLLADAInterface" RefBaseClassPath="ExternalDataConnector">
          <Attribute Name="refType" AttributeDataType="xs:string" />
          <Attribute Name="target" AttributeDataType="xs:token" />
        </InterfaceClass>
        <InterfaceClass Name="PLCopenXMLInterface" RefBaseClassPath="ExternalDataConnector" />
      </InterfaceClass>
      <InterfaceClass Name="Communication" RefBaseClassPath="AutomationMLBaseInterface">
        <InterfaceClass Name="SignalInterface" RefBaseClassPath="Communication" />
      </InterfaceClass>
      <InterfaceClass Name="AttachmentInterface" RefBaseClassPath="AutomationMLBaseInterface" />
    </InterfaceClass>
  </InterfaceClassLib>
</CAEXFile>

```

Figure C.2 – XML representation of AML libraries AutomationMLInterfaceClassLib

SOMMAIRE

AVANT-PROPOS	89
INTRODUCTION.....	91
1 Domaine d'application	94
2 Références normatives	94
3 Termes, définitions et abréviations	94
3.1 Termes et définitions	94
3.2 Abréviations	95
4 Conformité.....	95
5 Extensions des bibliothèques AML pour la géométrie et la cinématique	95
5.1 Généralités	95
5.2 AutomationMLBaseRoleClassLib – RoleClass Frame	95
5.3 AutomationMLInterfaceClassLib	96
5.3.1 InterfaceClass COLLADAInterface	96
5.3.2 InterfaceClass AttachmentInterface	96
6 Attribut «Frame» (cadre).....	96
7 Intégration de documents COLLADA	97
8 Adjonction de deux objets AML.....	98
9 Méta-informations relatives à l'outil source COLLADA	99
Annexe A (informative) Méthodes de référencement de descriptions géométriques/cinématiques	101
A.1 Intégration d'un document COLLADA commun par référencement explicite.....	101
A.1.1 Généralités	101
A.1.2 Définition de l'attribut Frame (cadre).....	102
A.1.3 Structure des documents COLLADA	105
A.1.4 Référencement à l'aide de l'URI et de fragments sans cible et sans ID	109
A.1.5 Référencement à l'aide de l'URI et de fragments comprenant une cible sans ID	109
A.1.6 Référencement à l'aide d'un URI sans fragment comprenant une cible et un ID	110
A.1.7 Référencement à l'aide de l'URI et de fragments comprenant une cible et un ID	111
A.1.8 Référencement à l'aide de l'URI sans fragment, sans cible et sans ID	112
A.2 Référencement implicite des éléments COLLADA	113
A.2.1 Généralités	113
A.2.2 Référencement implicite	114
A.2.3 Référencement implicite de sous-documents COLLADA	115
A.2.4 Publication d'éléments d'un document COLLADA en CAEX	121
A.3 Adjonction entre objets en CAEX	123
Annexe B (informative) Modélisation des systèmes cinématiques et leur combinaison en AML	129
B.1 Généralités	129
B.2 Modélisation d'un document AML d'une unité linéaire en CAEX et COLLADA	129
B.2.1 Généralités	129
B.2.2 Définition de la scène visuelle	129
B.2.3 Définition de l'articulation	131
B.2.4 Définition du modèle cinématique	131

B.2.5	Définition du système articulé	132
B.2.6	Définition de la scène cinématique	134
B.2.7	Assemblage de la scène	134
B.2.8	Combinaison de CAEX et COLLADA en AML	135
B.3	Modélisation d'un document AML d'un robot en CAEX et COLLADA	136
B.3.1	Généralités	136
B.3.2	Définition de la scène visuelle	137
B.3.3	Définition des articulations	139
B.3.4	Définition du modèle cinématique	140
B.3.5	Définition du système articulé	141
B.3.6	Définition de la scène cinématique	144
B.3.7	Assemblage de la scène	145
B.3.8	Combinaison de CAEX et de COLLADA en AML	145
B.4	Modélisation d'un document AML d'un système combiné comprenant un robot et un axe linéaire en CAEX et COLLADA	147
B.5	Modélisation d'un document AML d'un outil de préhension connecté au robot en CAEX et en COLLADA	150
B.5.1	Généralités	150
B.5.2	Définition de la scène visuelle	151
B.5.3	Définition du système cinématique	153
B.5.4	Assemblage de la scène	161
B.5.5	Combinaison de CAEX et de COLLADA en AML	162
B.6	Modélisation d'un document AML d'une pièce connectée à un outil de préhension en CAEX et en COLLADA	165
B.6.1	Généralités	165
B.6.2	Limite implicite supérieure	165
B.6.3	Définition de la pièce	167
B.6.4	Combinaison de CAEX et de COLLADA en AML	169
Annexe C (informative)	Représentation XML des bibliothèques AML	173
C.1	AutomationMLBaseRoleClassLib	173
C.2	AutomationMLInterfaceClassLib	174
Figure 1	– Vue d'ensemble du format AML d'échange de données techniques	92
Figure 2	– Texte XML exigé dans le cadre de l'ISO/PAS 17506	100
Figure 3	– Texte XML exigé dans le cadre du document COLLADA 1.4.1	100
Figure A.1	– Arbre de décision pour les différentes méthodes de référencement	101
Figure A.2	– Deux cadres représentés dans l'InstanceHierarchy d'un document AML	102
Figure A.3	– Représentation XML du document AML	103
Figure A.4	– Translation et rotation plane	104
Figure A.5	– Scène COLLADA utilisée dans cet exemple	105
Figure A.6	– Structure et références	106
Figure A.7	– Contenu du document COLLADA cube.dae	107
Figure A.8	– Contenu du document COLLADA red_blue_cubes.dae	108
Figure A.9	– “RedCube” – Hiérarchie du document AML	109
Figure A.10	– Représentation XML du document AML	109
Figure A.11	– Référencement du cube rouge par ID	109
Figure A.12	– “BlueCube” – Hiérarchie du document AML	110

Figure A.13 – Représentation XML du document AML	110
Figure A.14 – Référencement du cube bleu	110
Figure A.15 – Hiérarchie du document AML	110
Figure A.16 – Représentation XML du document AML	111
Figure A.17 – Référencement du cube bleu à partir de l'élément "subpart"	111
Figure A.18 – Hiérarchie du document AML	111
Figure A.19 – Représentation XML du document AML	112
Figure A.20 – Référencement du cube bleu	112
Figure A.21 – Hiérarchie du document AML	112
Figure A.22 – Représentation XML du document AML	113
Figure A.23 – Référencement de la scène COLLADA complète	113
Figure A.24 – Référencement implicite: Hiérarchie du document AML	114
Figure A.25 – Représentation XML du document AML	115
Figure A.26 – Structure et relations des sous-documents COLLADA référencés	116
Figure A.27 – Contenu du document COLLADA modifié red_blue_cubes.dae	117
Figure A.28 – Contenu du document COLLADA red_cube.dae	118
Figure A.29 – Contenu du document COLLADA blue_cube.dae	119
Figure A.30 – Référencement implicite: Hiérarchie du document AML	120
Figure A.31 – Représentation XML du document AML	120
Figure A.32 – Scène COLLADA modifiée avec nœud supplémentaire	121
Figure A.33 – Élément cadre supplémentaire du document COLLADA	122
Figure A.34 – Cadres de publication: Hiérarchie du document AML	122
Figure A.35 – Représentation XML du document AML	123
Figure A.36 – Structure pour les adjonctions entre objets en CAEX	124
Figure A.37 – Visualisation du cube jaune avec un cadre supplémentaire	124
Figure A.38 – Document COLLADA du cube jaune avec un cadre supplémentaire	125
Figure A.39 – Hiérarchie du document AML	126
Figure A.40 – Représentation XML du document AML	127
Figure A.41 – Adjonction entre des objets AML géométriques	128
Figure A.42 – Représentation XML du document AML	128
Figure B.1 – Visualisation de l'unité linéaire	129
Figure B.2 – Définition de la scène visuelle	131
Figure B.3 – Définition de l'articulation	131
Figure B.4 – Définition de modèle cinématique	132
Figure B.5 – Définition de la bibliothèque du système articulé	132
Figure B.6 – Définition du système cinématique articulé	133
Figure B.7 – Définition du système de mouvement articulé	134
Figure B.8 – Définition de la scène cinématique	134
Figure B.9 – Instanciation de la scène cinématique	135
Figure B.10 – Hiérarchie du document AML	135
Figure B.11 – Représentation XML du document AML	136
Figure B.12 – Visualisation du robot	137
Figure B.13 – Définition de la scène visuelle	139

Figure B.14 – Définition d'articulations	140
Figure B.15 – Définition de modèle cinématique	140
Figure B.16 – Définition de la bibliothèque du système articulé	141
Figure B.17 – Définition du système cinématique articulé	142
Figure B.18 – Définition du système de mouvement articulé	144
Figure B.19 – Définition de la scène cinématique	144
Figure B.20 – Instanciation de la scène cinématique	145
Figure B.21 – Hiérarchie du document AML	146
Figure B.22 – Représentation XML du document AML	146
Figure B.23 – Structure du référencement de sous-documents COLLADA	147
Figure B.24 – Hiérarchie du document AML	148
Figure B.25 – Représentation XML du document AML	149
Figure B.26 – Représentation XML du document AML	149
Figure B.27 – Visualisation du robot adjoint à l'unité linéaire	150
Figure B.28 – Hiérarchie des documents COLLADA	150
Figure B.29 – Visualisation de l'outil de préhension	151
Figure B.30 – Définition de la scène visuelle	152
Figure B.31 – Définition de la cinématique	153
Figure B.32 – Définition des articulations	154
Figure B.33 – Définition du modèle cinématique	154
Figure B.34 – Définition du système articulé	155
Figure B.35 – Définition du système articulé	156
Figure B.36 – Définition de la scène cinématique	157
Figure B.37 – Définition de la dépendance des articulations en MathML	157
Figure B.38 – Représentation XML du document COLLADA gripper_kinematics.dae	161
Figure B.39 – Représentation XML du document COLLADA gripper.dae	161
Figure B.40 – Hiérarchie du document AML	162
Figure B.41 – Représentation XML du document AML	164
Figure B.42 – Représentation XML du document AML	164
Figure B.43 – Visualisation du robot sur une unité linéaire et l'outil de préhension adjoint	165
Figure B.44 – Exemple de limite implicite supérieure	166
Figure B.45 – Structure des adjonctions entre objets en CAEX	167
Figure B.46 – Visualisation de la pièce avec un cadre supplémentaire	168
Figure B.47 – Document COLLADA d'une pièce avec un cadre supplémentaire	169
Figure B.48 – Hiérarchie du document AML	170
Figure B.49 – Représentation XML du document AML	172
Figure B.50 – Adjonctions entre objets AML géométriques	172
Figure B.51 – Représentation XML du document AML	172
Figure C.1 – Représentation XML des bibliothèques AML AutomationMLBaseRoleClassLib	173
Figure C.2 – Représentation XML de bibliothèques AML AutomationMLInterfaceClassLib	174

Tableau 1 – Abréviations	95
Tableau 2 – RoleClass Frame	95
Tableau 3 – InterfaceClass COLLADAInterface	96
Tableau 4 – InterfaceClass AttachmentInterface	96
Tableau 5 – Attribut “Frame”	97
Tableau 6 – Sous-attributs de l’attribut “Frame”	97
Tableau 7 – Règles pour la résolution de documents et de points d’entrée	98
Tableau 8 – Méta-informations relatives à l’outil source COLLADA	100

IECNORM.COM : Click to view the full PDF of IEC 62714-3:2017

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

**FORMAT D'ÉCHANGE DE DONNÉES TECHNIQUES
POUR UNE UTILISATION DANS L'INGÉNIERIE
DES SYSTÈMES D'AUTOMATISATION INDUSTRIELLE –
AUTOMATION MARKUP LANGUAGE –****Partie 3: Géométrie et cinématique****AVANT-PROPOS**

- 1) La Commission Électrotechnique Internationale (IEC) est une organisation mondiale de normalisation composée de l'ensemble des comités électrotechniques nationaux (Comités nationaux de l'IEC). L'IEC a pour objet de favoriser la coopération internationale pour toutes les questions de normalisation dans les domaines de l'électricité et de l'électronique. À cet effet, l'IEC – 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 l'IEC"). 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 l'IEC, participent également aux travaux. L'IEC 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 l'IEC concernant les questions techniques représentent, dans la mesure du possible, un accord international sur les sujets étudiés, étant donné que les Comités nationaux de l'IEC intéressés sont représentés dans chaque comité d'études.
- 3) Les Publications de l'IEC se présentent sous la forme de recommandations internationales et sont agréées comme telles par les Comités nationaux de l'IEC. Tous les efforts raisonnables sont entrepris afin que l'IEC s'assure de l'exactitude du contenu technique de ses publications; l'IEC 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 l'IEC s'engagent, dans toute la mesure possible, à appliquer de façon transparente les Publications de l'IEC dans leurs publications nationales et régionales. Toutes divergences entre toutes Publications de l'IEC et toutes publications nationales ou régionales correspondantes doivent être indiquées en termes clairs dans ces dernières.
- 5) L'IEC 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 l'IEC. L'IEC 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 l'IEC, 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 l'IEC, 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 l'IEC ou de toute autre Publication de l'IEC, 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 l'IEC peuvent faire l'objet de droits de brevet. L'IEC ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de brevets et de ne pas avoir signalé leur existence.

La Norme internationale IEC 62714-3 a été établie par le sous-comité 65E: Les dispositifs et leur intégration dans les systèmes de l'entreprise, du comité d'études 65 de l'IEC: Mesure, commande et automation dans les processus industriels.

Le texte de cette norme est issu des documents suivants:

CDV	Rapport de vote
65E/497/CDV	65E/508/RVC

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

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

Une liste de toutes les parties de la série IEC 62714, publiées sous le titre général *Format d'échange de données techniques pour une utilisation dans l'ingénierie des systèmes d'automatisation industrielle – Automation markup language*, peut être consultée sur le site web de l'IEC.

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 l'IEC 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.

INTRODUCTION

Le format d'échange de données défini dans l'IEC 62714 (Automation Markup Language, AML) est un format de données de type schéma XML mis au point afin de faciliter l'échange de données entre des outils techniques dans un environnement hétérogène d'outils techniques. L'IEC 62714-1 propose une vue d'ensemble de ce format.

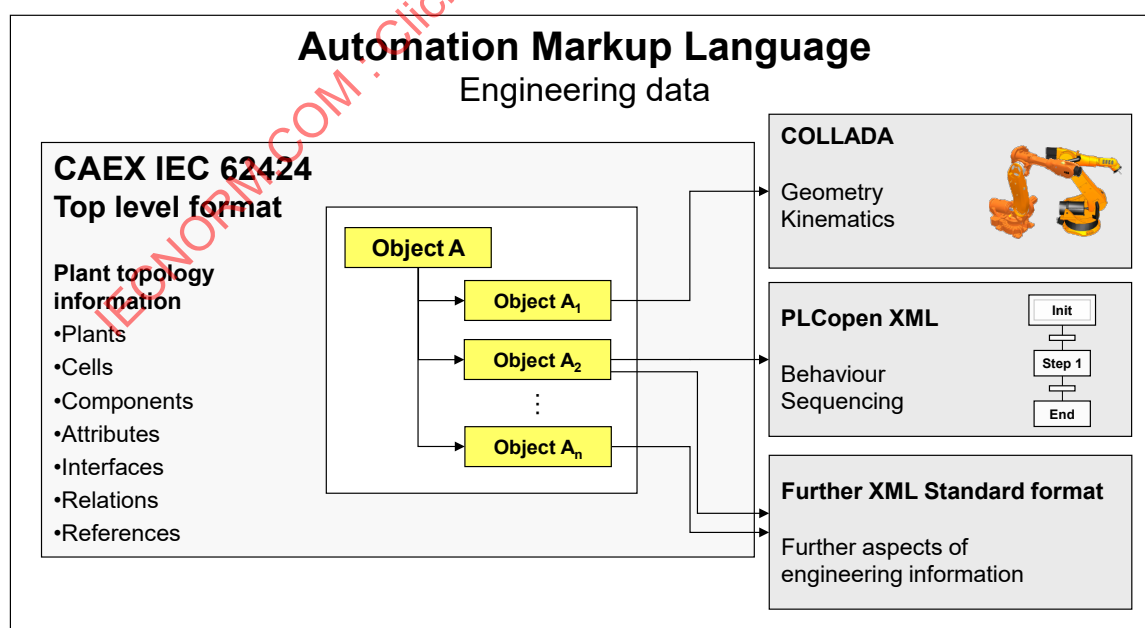
L'objectif de l'AML est l'interconnexion d'outils techniques provenant d'environnements hétérogènes d'outils techniques existants dans leurs différentes disciplines, par exemple l'ingénierie des installations mécaniques, l'étude d'électricité, l'ingénierie de procédés, l'ingénierie de commande de processus, le développement des IHM, la programmation PLC, la programmation de robots, etc.

L'AML archive les informations techniques en respectant le paradigme orienté objet et permet la modélisation des composants d'installations physiques et logiques sous forme d'objets de données qui englobent différents aspects. Un objet peut comporter d'autres sous-objets et peut lui-même faire partie intégrante d'une composition ou d'une agrégation plus importante. Les objets typiques qui constituent l'automatisation d'installations comprennent les informations relatives à la topologie, à la géométrie, à la cinématique et à la logique, tandis que la logique comprend pour sa part le séquençement, le comportement et la commande.

L'AML combine les formats de données industrielles existants, conçus pour l'archivage et l'échange des informations techniques sous différents aspects. Ces formats de données sont utilisés « en l'état » dans le cadre de leurs propres spécifications et ne sont pas associés aux besoins du langage AML.

La caractéristique centrale de l'AML est le format de données central CAEX qui connecte les différents formats de données. Le langage AML a, par conséquent, une architecture de document répartie intrinsèque.

La Figure 1 représente l'architecture AML de base et la répartition des informations relatives à la topologie, à la géométrie, à la cinématique et à la logique.



Anglais	Français
Engineering data	Données techniques
CAEX IEC 62424 top level format	Format central CAEX défini dans l'IEC 62424
Object	Objet

Anglais	Français
Plant topology information	Informations relatives à la topologie de l'installation
Plants	Installations
Cells	Cellules
Components	Composants
Attributes	Attributs
Interfaces	Interfaces
Relations	Relations
References	Références
Geometry	Géométrie
Kinematics	Cinématique
Behaviour	Comportement
Sequencing	Séquencement
Init	Début
Step	Étape
End	Fin
Further XML standard format	Autre format XML standard
Further aspects of engineering information	Autres aspects des informations techniques

Figure 1 – Vue d'ensemble du format AML d'échange de données techniques

Du fait des différents aspects d'AML, l'IEC 62714 comporte différentes parties portant sur différents aspects:

- IEC 62714-1: Architecture et exigences générales
Cette partie spécifie l'architecture AML générale, et la modélisation des données techniques, classes, instances, relations, références, hiérarchies, bibliothèques AML de base et concepts AML étendus.
- IEC 62714-2: Bibliothèques de classes de rôles
Cette partie spécifie d'autres bibliothèques AML.
- IEC 62714-3: Géométrie et cinématique
Cette partie spécifie la modélisation des informations relatives à la géométrie et à la cinématique.

D'autres parties pourront être ajoutées à l'avenir afin d'interconnecter d'autres normes de données avec l'AML.

L'Article 5 décrit les extensions de géométrie des bibliothèques de classes de rôles.

L'Article 6 décrit l'attribut cadre ("Frame") pouvant être utilisé pour représenter la position géométrique d'un objet InternalElement, InstanceHierarchy, SystemUnitClass ou SystemUnitClassLibrary par rapport à un autre objet CAEX.

L'Article 7 donne une description normative du référencement des documents COLLADA.

L'Article 8 spécifie les dispositions normatives relatives à l'adjonction de deux objets géométriques AML.

L'Article 9 définit la manière d'archiver des méta-informations relatives à l'outil source directement dans le document COLLADA.

L'Annexe A décrit les méthodes de référencement des modèles géométriques et cinématiques.

L'Annexe B donne un exemple de modélisation de systèmes cinématiques et leur combinaison en AML.

L'Annexe C donne une représentation XML informative des bibliothèques définies dans la présente partie de l'IEC 62714.

IECNORM.COM : Click to view the full PDF of IEC 62714-3:2017

FORMAT D'ÉCHANGE DE DONNÉES TECHNIQUES POUR UNE UTILISATION DANS L'INGÉNIERIE DES SYSTÈMES D'AUTOMATISATION INDUSTRIELLE – AUTOMATION MARKUP LANGUAGE –

Partie 3: Géométrie et cinématique

1 Domaine d'application

La présente partie de l'IEC 62714 spécifie l'intégration des informations relatives à la géométrie et à la cinématique pour l'échange entre les outils techniques dans la zone d'automatisation de l'installation par le biais du langage AML.

Elle ne définit pas les détails de la procédure d'échange de données ou des exigences de mise en œuvre pour les outils d'importation/exportation.

2 Références normatives

Les documents suivants cités dans le texte constituent pour tout ou partie de leur contenu, des exigences du présent document. 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).

IEC 62714-1:2014, *Format d'échange de données techniques pour une utilisation dans l'ingénierie des systèmes d'automatisation industrielle – Automation markup language – Partie 1: Architecture et exigences générales*

IEC 62714-2:2015, *Format d'échange de données techniques pour une utilisation dans l'ingénierie des systèmes d'automatisation industrielle – Automation markup language – Partie 2: Bibliothèques de classes de rôles*

ISO/PAS 17506, *Systèmes d'automatisation industrielle et intégration – Spécifications du schéma des actifs numériques COLLADA pour la visualisation 3D des données industrielles*

COLLADA 1.4.1: March 2008 COLLADA – Digital Asset Schema Release 1.4.1
(disponible sous <http://www.khronos.org/files/collada_spec_1_4.pdf>)

Extensible Markup Language (XML) 1.0:2004, W3C Recommendation
(disponible sous <<http://www.w3.org/TR/2004/REC-xml-20040204/>>)

3 Termes, définitions et abréviations

3.1 Termes et définitions

Pour les besoins du présent document, les termes et les définitions de l'IEC 62714-1:2014 et de l'IEC 62714-2:2015 ainsi que les suivants, s'appliquent.

L'ISO et l'IEC tiennent à jour des bases de données terminologiques destinées à être utilisées en normalisation, consultables aux adresses suivantes:

- IEC Electropedia: disponible à l'adresse <http://www.electropedia.org/> [consulté le 26 septembre 2016]

- ISO Online browsing platform: disponible à l'adresse <http://www.iso.org/obp> [consulté le 26 septembre 2016]

3.1.1

chemin SID

référence du type `sidref_type` d'un élément arbitraire au sein d'un document COLLADA, conformément à l'ISO/PAS 17506 ou au document COLLADA 1.4.1

3.2 Abréviations

Pour les besoins du présent document, les abréviations données dans l'IEC 62714-1:2014 et l'IEC 62714-2:2015 ainsi que les abréviations énoncées dans le Tableau 1 s'appliquent.

Tableau 1 – Abréviations

SCARA	Selective Compliance Assembly Robot Arm (Bras robotisé d'assemblage à conformité sélective)
SID	Scoped Identifier (Identifiant à portée)

4 Conformité

Les exigences des Articles 5, 6, 7, 8 et 9 doivent être respectées pour revendiquer la conformité au présent document, relatif à la prise en charge du langage AML. Dans le domaine d'application du langage AML, un document COLLADA doit être conforme à la spécification de l'ISO/PAS 17506 ou au document COLLADA 1.4.1.

5 Extensions des bibliothèques AML pour la géométrie et la cinématique

5.1 Généralités

L'Article 5 définit les extensions des classes AML standard `RoleClasses` et `InterfaceClasses`. Ces classes font partie des bibliothèques AML standard et d'une extension particulière de l'IEC 62714-1 pour la présente partie de l'IEC 62714. Tous les attributs décrits font partie des bibliothèques standard et peuvent être retirés de l'`InstanceHierarchy` s'ils ne sont pas nécessaires.

5.2 AutomationMLBaseRoleClassLib – RoleClass Frame

La `RoleClass` (classe de rôle) "Frame" doit être utilisée, comme spécifié au Tableau 2.

Tableau 2 – RoleClass Frame

Nom de la classe	Frame
Description	Ce rôle désigne un système direct de coordonnées cartésiennes.
Classe parente	AutomationMLBaseRoleClassLib/AutomationMLBaseRole
Chemin pour le référencement de l'élément	AutomationMLBaseRoleClassLib/AutomationMLBaseRole/Frame

NOTE 1 Il est possible d'utiliser un objet AML qui référence la `RoleClass` "Frame" pour définir les systèmes de coordonnées de référence comme les points d'adjonction.

NOTE 2 Pour définir les systèmes de coordonnées de référence comme les points d'adjonction, il est possible d'utiliser un objet AML qui référence la `RoleClass` "Frame".

5.3 AutomationMLInterfaceClassLib

5.3.1 InterfaceClass COLLADAInterface

L'InterfaceClass (classe d'interface) "COLLADAInterface" doit être utilisée, comme spécifié au Tableau 3.

Tableau 3 – InterfaceClass COLLADAInterface

Nom de la classe	COLLADAInterface	
Description	L'InterfaceClass "COLLADAInterface" doit être utilisée pour le référencement des documents COLLADA externes et pour la publication des interfaces définies dans le document COLLADA externe.	
Classe parente	AutomationMLInterfaceClassLib/AutomationMLBaseInterface/ExternalDataConnector	
Chemin pour le référencement de l'élément	AutomationMLInterfaceClassLib/AutomationMLBaseInterface/ExternalDataConnector/COLLADAInterface	
Attributs	refType (AttributeDataType="xs:string")	L'attribut "refType" spécifie si le document COLLADA référence un caractère explicite ou implicite. Les valeurs admises sont "explicit" ou "implicit". L'attribut est obligatoire.
	target (AttributeDataType="xs:token")	L'attribut "target" spécifie le chemin SID d'un élément COLLADA dans le document référencé. L'attribut est facultatif.

NOTE L'InterfaceClass "COLLADAInterface" hérite en plus de l'attribut "refURI" de l'InterfaceClass parente "ExternalDataConnector".

5.3.2 InterfaceClass AttachmentInterface

L'InterfaceClass "AttachmentInterface" doit être utilisée, comme spécifié au Tableau 4.

Tableau 4 – InterfaceClass AttachmentInterface

Nom de la classe	AttachmentInterface
Description	L'InterfaceClass "AttachmentInterface" spécifie une interface pour les liaisons géométriques ou cinématiques entre les objets AML de la RoleClass "Frame".
Classe parente	AutomationMLInterfaceClassLib/AutomationMLBaseInterface
Chemin pour le référencement de l'élément	AutomationMLInterfaceClassLib/AutomationMLBaseInterface/AttachmentInterface

6 Attribut «Frame» (cadre)

Un InternalElement (élément interne) ou une SystemUnitClass (classe d'unité système) peuvent exiger un attribut cadre représentant sa position géométrique par rapport aux autres objets. À cet effet, les dispositions suivantes s'appliquent:

- Chaque cadre doit être fondé sur un système direct tridimensionnel et orthogonal de coordonnées.
- Les éléments InstanceHierarchy et SystemUnitClassLib spécifient un système direct tridimensionnel et orthogonal de coordonnées de base standard. L'axe positif z est considéré comme l'axe de profondeur, la direction positive x définit l'axe horizontal et la direction négative y définit l'axe vertical.
- Les translations relatives x , y et z ainsi que les rotations rx , ry et rz doivent être spécifiées comme sous-attributs d'un attribut AML "Frame" défini au Tableau 5 et au Tableau 6. Les

translations relatives x , y et z doivent être données par rapport au système parent de coordonnées spécifié au point précédent. Les rotations rx , ry et rz doivent être effectuées dans l'ordre rx , ry et rz par rapport aux axes fixes du système parent de coordonnées. L'origine du Frame translaté doit conserver sa position (x, y, z) au cours de la rotation plane. L'attribut "Frame" doit affecter son objet AML ainsi que tous ses enfants.

NOTE Cela signifie que le point de rotation pivote tandis que la position du cadre ne varie pas. Cela évite les doubles modifications.

- Si l'attribut "Frame" n'est pas spécifié, les valeurs par défaut de x , y , z , rx , ry et rz doivent être de 0.
- Si l'attribut "Frame" est spécifié, tous les sous-attributs x , y , z , rx , ry et rz doivent être énoncés. Tout sous-attribut non utilisé doit avoir une valeur par défaut de 0.

L'attribut "Frame" doit être utilisé comme attribut complexe spécifié au Tableau 5 et au Tableau 6. Un exemple est donné en A.1.2.

Tableau 5 – Attribut "Frame"

Attribut	AttributeDataType	Description
Frame	Cet attribut n'a pas d'AttributeDataType étant donné qu'il n'a aucune valeur.	L'attribut "Frame" doit être utilisé comme élément de structure pour l'archivage des sous-attributs spécifiés au Tableau 6.

Tableau 6 – Sous-attributs de l'attribut "Frame"

Attribut	AttributeDataType	Description
x	xs:double	L'attribut "x" doit être utilisé pour spécifier la position relative, en mètres, le long de l'axe x du système de coordonnées spécifié par l'élément parent. La valeur de l'attribut "Unit" de l'élément CAEX "Attribute" doit être "m".
y	xs:double	L'attribut "y" doit être utilisé pour spécifier la position relative, en mètres, le long de l'axe y du système de coordonnées spécifié par l'élément parent. La valeur de l'attribut "Unit" de l'élément CAEX "Attribute" doit être "m".
z	xs:double	L'attribut "z" doit être utilisé pour spécifier la position relative, en mètres, le long de l'axe z du système de coordonnées spécifié par l'élément parent. La valeur de l'attribut "Unit" de l'élément CAEX "Attribute" doit être "m".
rx	xs:double	L'attribut "rx" doit être utilisé pour spécifier la rotation relative, en degrés, autour de l'axe x du système de coordonnées spécifié par l'élément parent. La valeur de l'attribut "Unit" de l'élément CAEX "Attribute" doit être "deg".
ry	xs:double	L'attribut "ry" doit être utilisé pour spécifier la rotation relative, en degrés, autour de l'axe y du système de coordonnées spécifié par l'élément parent. La valeur de l'attribut "Unit" de l'élément CAEX "Attribute" doit être "deg".
rz	xs:double	L'attribut "rz" doit être utilisé pour spécifier la rotation relative, en degrés, autour de l'axe z du système de coordonnées spécifié par l'élément parent. La valeur de l'attribut "Unit" de l'élément CAEX "Attribute" doit être "deg".

7 Intégration de documents COLLADA

En ce qui concerne le référencement de documents COLLADA, les dispositions suivantes s'appliquent:

- Une référence d'un objet AML à un document COLLADA doit être modélisée par le biais d'une ExternalInterface CAEX déduite de l'InterfaceClass AML "COLLADAInterface", tel que défini en 5.3.1.
- Le document COLLADA doit être référencé par son URI dans l'attribut "refURI" de ladite ExternalInterface CAEX.
- La valeur de l'attribut "target" doit renvoyer à un élément se trouvant dans le document COLLADA et doit suivre la syntaxe d'un chemin SID.

- Si l'attribut "target" n'est pas présent ou que son contenu est vide et que l'URI n'a pas de fragment, l'élément "scene" du document COLLADA doit être pris en compte.
- Le document COLLADA et son point d'entrée doivent être résolus comme spécifié au Tableau 7. Un arbre de décision visant à résoudre les différentes entrées est représenté à la Figure A.1.
- La valeur "explicit" de l'attribut "refType" doit être utilisée pour les objets AML (ci-après nommés "objets AML explicites") qui font partie d'une représentation de données.

NOTE 1 La géométrie d'un robot et d'un transporteur dans une scène géométrique commune est un exemple d'objets AML explicites.

- La valeur "implicit" de l'attribut "refType" doit être utilisée pour les objets AML (ci-après nommés "objets AML implicites") qui sont utilisés à des fins particulières pour le référencement d'objets déjà représentés, par exemple pour la publication de cadres ou le référencement d'objets déjà représentés.

NOTE 2 La référence d'une géométrie de robot déjà explicitement référencée est un exemple d'objet AML implicite. Le référencement implicite évite la double représentation de géométries.

- Les objets se trouvant dans le document COLLADA peuvent uniquement être référencés à l'aide de la valeur "implicit" s'ils font partie d'un objet COLLADA ou s'ils sont identiques à un objet COLLADA qui a déjà été référencé avec la valeur "explicit". Les objets AML implicites doivent être modélisés comme des enfants directs ou indirects d'objets AML explicites.
- Conformément aux règles d'instanciation et de référencement externe spécifiées dans l'ISO/PAS 17506, lorsqu'un document COLLADA (ci-après nommé "document principal") référence d'autres documents COLLADA (ci-après nommés "sous-documents"), toutes les références aux éléments au sein des sous-documents COLLADA référencés doivent uniquement être effectuées par le biais du document COLLADA principal. Un exemple est donné en A.2.3.
- Un objet AML doit avoir zéro ou une "COLLADAInterface" comme enfant direct.

Tableau 7 – Règles pour la résolution de documents et de points d'entrée

		Attribut "refURI"	
		URI avec fragment	URI sans fragment
Attribut "target"	Chemin SID qui commence avec un ID	L'URI spécifie le document. Le chemin SID spécifie le point d'entrée.	L'URI spécifie le document. Le chemin SID spécifie le point d'entrée.
	Chemin SID qui commence sans ID	L'URI spécifie le document et un élément correspondant pour le point d'entrée. Le chemin SID spécifie le point d'entrée.	Statut non défini. Cette combinaison ne doit pas être utilisée.
	Valeur manquante ou champ vide	L'URI spécifie le document et le point d'entrée.	L'URI spécifie le document. L'élément "scene" spécifie le point d'entrée.

8 Adjonction de deux objets AML

L'Article 8 spécifie les dispositions normatives relatives à l'adjonction de deux objets AML géométriques. Un exemple est donné à l'Article A.3.

Pour l'adjonction de deux objets AML, les dispositions suivantes s'appliquent:

- Un objet AML doit avoir zéro ou une "AttachmentInterface" comme enfant direct.
- Une liaison entre deux ExternalInterfaces affectées à une InterfaceClass "AttachmentInterface" doit être une connexion unidirectionnelle avec couplage géométrique fixe à l'aide d'un élément InternalLink CAEX.

- La transformation relative à partir d'un objet AML référencé par l'attribut RefPartnerSideA à un objet référencé par l'attribut RefPartnerSideB doit rester constante même si l'objet de RefPartnerSideA est transformé, donnant lieu à une transformation subséquente de RefPartnerSideB en cas de transformation de RefPartnerSideA. Une transformation de RefPartnerSideB ne modifie pas la transformation de RefPartnerSideA.

NOTE 1 RefPartnerSideA et RefPartnerSideB sont des attributs de l'élément InternalLink CAEX.

- Étant donné qu'un point d'adjonction fait partie d'un objet AML, l'interface d'adjonction "AttachmentInterface" doit faire partie de l'objet AML correspondant.
- Si une "AttachmentInterface" est une interface enfant d'un objet AML de la RoleClass "Frame" ou d'une RoleClass héritée, l'objet à déplacer doit être l'objet AML parent de l'objet AML de la RoleClass "Frame" ou d'une RoleClass héritée (même si l'objet AML parent a la RoleClass "Frame" ou une RoleClass héritée).
- Si un objet AML disposant d'une "AttachmentInterface" a la RoleClass "Frame" ou une RoleClass héritée et qu'il a une interface de type "COLLADAInterface", l'objet AML parent doit également avoir une interface de type "COLLADAInterface". L'interface "COLLADAInterface" de l'objet AML parent représente la limite de la géométrie de la géométrie adjointe.
- Si une adjonction fait référence à un élément géométrique, une "COLLADAInterface" doit référencer un nœud COLLADA de la scène visuelle. L'adjonction doit déplacer la géométrie complète (de manière récursive jusqu'à l'attribut de nœud d'un élément bind_kinematics_model) de l'objet adjoint sans modification de l'état cinématique de l'objet adjoint.
- Si une adjonction fait référence à un élément cinématique, une "COLLADAInterface" doit référencer un élément COLLADA se trouvant dans un élément COLLADA kinematics_scene. L'adjonction doit affecter la configuration de l'articulation de l'objet cinématique adjoint lors du déplacement de l'objet AML parent de l'objet adjoint. L'outil d'importation est responsable du calcul des éléments cinématiques couplés.

NOTE 2 Pour établir une connexion bidirectionnelle, deux liaisons unidirectionnelles de sens opposé sont nécessaires.

NOTE 3 Il est possible d'associer plusieurs liaisons à "AttachmentInterface".

9 Méta-informations relatives à l'outil source COLLADA

Afin de simplifier l'échange de données entre un outil source et un outil cible, il est utile d'archiver les informations relatives à l'outil source directement dans le document COLLADA. Par conséquent, les dispositions suivantes s'appliquent:

- Chaque document COLLADA doit fournir des informations relatives à l'outil qui a écrit le document COLLADA.
- Dans une chaîne d'outils d'échange de données, le dernier outil d'édition du document doit archiver ces informations dans le document COLLADA.
- Ces informations doivent être archivées dans l'élément enfant "asset" de l'élément "COLLADA".
- Les méta-informations doivent comporter les éléments suivants:
 - le fournisseur de l'outil d'écriture;
 - l'adresse universelle de l'outil d'écriture;
 - le nom de l'outil d'écriture;
 - la version de produit de l'outil d'écriture;
 - le numéro de la version finale de produit de l'outil d'écriture;
 - l'horodatage de création et l'horodatage de la dernière modification du document COLLADA;
 - la définition de l'unité et de l'axe vertical.

- Les méta-informations doivent être archivées à l'aide des éléments présentés au Tableau 8.

Tableau 8 – Méta-informations relatives à l'outil source COLLADA

Nom d'étiquette XML COLLADA 1.4.1	Nom d'étiquette XML ISO/PAS 17506	Type	Niveau	Exemple
"author"	"author"	xs:string	Obligatoire	ToolX Vendor (fournisseur de l'outil X)
"authoring_tool"	"authoring_tool"	xs:string	Obligatoire	"ToolX" "0.2" "123 prealpha"
"comments"	"author_website"	xs:anyURI	Obligatoire	http://www.ToolX-Vendor.org
"created"	"created"	xs:dateTime	Obligatoire	2014-04-14T11:37:18.1875000
"modified"	"modified"	xs:dateTime	Obligatoire	2014-04-14T11:37:18.1875000
"unit meter=... name=..."	"unit meter=... name=..."	xs:string	Obligatoire	name="meter" meter="1.0"
"up_axis"	"up_axis"	xs:string	Obligatoire	Z_UP

Les éléments XML doivent comporter les informations suivantes:

- "author" indique le fournisseur de l'outil d'écriture;
- "authoring_tool" indique le nom, la version de produit et le numéro de la version finale de produit de l'outil d'écriture séparés par un blanc et chacun apparaissant entre guillemets;
- dans le cadre de l'ISO/PAS 17506, "author_website" indique l'adresse universelle de l'outil d'écriture, dans le cadre de COLLADA 1.4.1, "comments" indique l'adresse universelle de l'outil d'écriture;
- "created" et "modified" indiquent l'horodatage de création et l'horodatage de la dernière modification, conformément aux règles spécifiées dans l'ISO/PAS 17506;
- "unit" et "up_axis" doivent être définis comme spécifié dans l'ISO/PAS 17506 ou dans le document COLLADA 1.4.1.

La Figure 2 et la Figure 3 donnent respectivement un exemple de texte XML exigé dans le cadre de l'ISO/PAS 17506 et dans le cadre du document COLLADA 1.4.1.

```
<asset>
  <contributor>
    <author>ToolX Vendor</author>
    <author_website>http://www.ToolX-Vendor.org</author_website>
    <authoring_tool>"ToolX" "0.2" "123 prealpha"</authoring_tool>
  </contributor>
  <created>2014-04-14T11:37:18.1875000</created>
  <modified>2014-04-14T11:37:18.1875000</modified>
  <!-- .. -->
  <unit name="meter" meter="1.0" />
  <up_axis>Z_UP</up_axis>
</asset>
```

Figure 2 – Texte XML exigé dans le cadre de l'ISO/PAS 17506

```
<asset>
  <contributor>
    <author>ToolX Vendor</author>
    <comments>http://www.ToolX-Vendor.org</comments>
    <authoring_tool>"ToolX" "0.2" "123 prealpha"</authoring_tool>
  </contributor>
  <created>2014-04-14T11:37:18.1875000</created>
  <modified>2014-04-14T11:37:18.1875000</modified>
  <!-- .. -->
  <unit name="meter" meter="1.0" />
  <up_axis>Z_UP</up_axis>
</asset>
```

Figure 3 – Texte XML exigé dans le cadre du document COLLADA 1.4.1

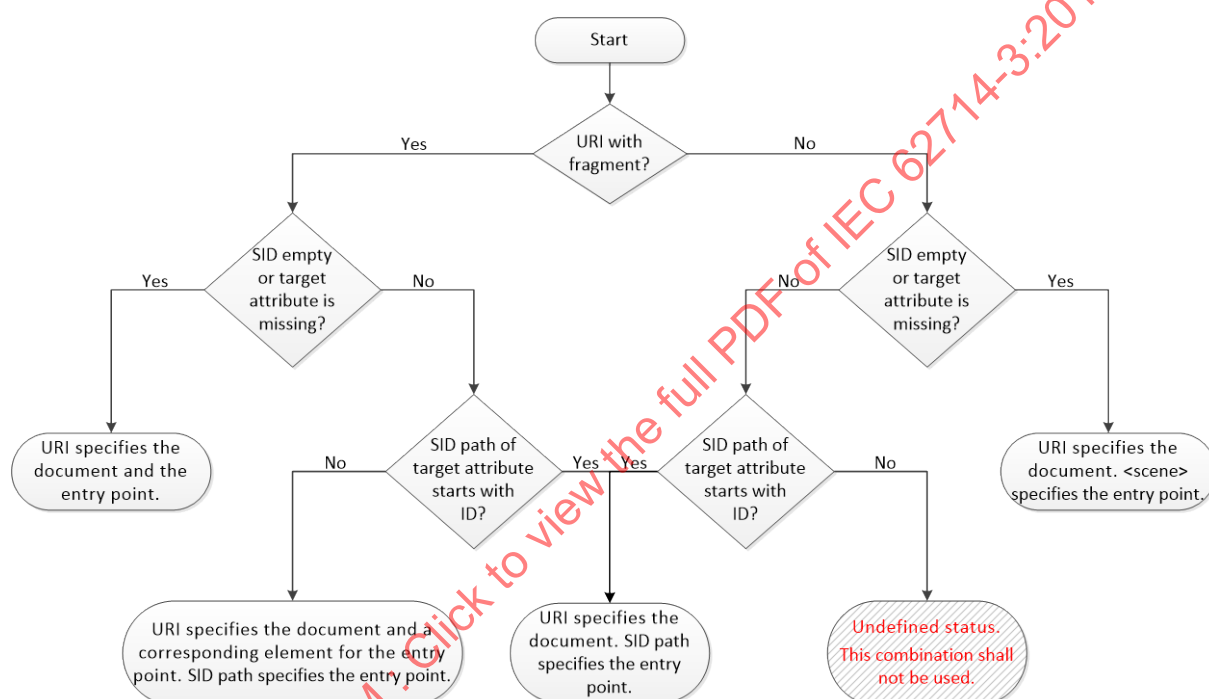
Annexe A (informative)

Méthodes de référencement de descriptions géométriques/cinématiques

A.1 Intégration d'un document COLLADA commun par référencement explicite

A.1.1 Généralités

L'Article A.1 décrit les méthodes de référencement explicite pour les modèles géométriques et cinématiques situés dans des documents COLLADA. La Figure A.1 représente les différentes méthodes utilisées dans les paragraphes qui suivent au moyen d'un arbre de décision.



Anglais	Français
Start	Début
Yes	Oui
URI with fragment?	URI avec fragment?
No	Non
SID empty or target attribute is missing?	SID vide ou attribut cible manquant?
URI specifies the document and the entry point.	L'URI spécifie le document et le point d'entrée.
SID path of target attribute starts with ID?	Le chemin SID de l'attribut cible commence-t-il par un ID?
URI specifies the document. <scene> specifies the entry point.	L'URI spécifie le document. <scene> spécifie le point d'entrée.
URI specifies the document and a corresponding element for the entry point. SID path specifies the entry point.	L'URI spécifie le document et un élément correspondant pour le point d'entrée. Le chemin SID spécifie le point d'entrée.
URI specifies the document. SID path specifies the entry point.	L'URI spécifie le document. Le chemin SID spécifie le point d'entrée.
Undefined status. This combination shall not be used.	Statut non défini. Cette combinaison ne doit pas être utilisée.

Figure A.1 – Arbre de décision pour les différentes méthodes de référencement

NOTE 1 Dans le présent document, les chaînes de chemins ou de sous-chemins sont représentées sous une forme abrégée de type "PATH1", "PATH2", etc. Cela simplifie la lisibilité et permet un chemin ou un sous-chemin réel. Les numéros identiques indiquent un chemin identique et, ainsi, une égalité sémantique.

NOTE 2 Afin d'améliorer la lisibilité, seules les parties applicables des documents AML/COLLADA sont présentées. Les parties manquantes sont signalées par "[...]".

NOTE 3 Afin d'améliorer la lisibilité, tout attribut RefBaseClassPath qui renvoie à la classe "AutomationMLInterfaceClassLib/AutomationMLBaseInterface/ExternalDataConnector/COLLADAInterface" est remplacé par "PATH_CI".

NOTE 4 Afin d'améliorer la lisibilité, tout attribut RefBaseClassPath qui renvoie à la classe "AutomationMLInterfaceClassLib/AutomationMLBaseInterface/AttachmentInterface" est remplacé par "PATH_AI".

NOTE 5 Afin d'améliorer la lisibilité, tout attribut RefBaseRoleClassPath qui renvoie à la classe "AutomationMLDMIRoleClassLib/DiscManufacturingEquipment/StaticObject" est remplacé par "PATH_SO".

NOTE 6 Afin d'améliorer la lisibilité, tout attribut RefBaseRoleClassPath qui renvoie à la classe "AutomationMLBaseRoleClassLib/AutomationMLBaseRole/Frame" est remplacé par "PATH_FRAME".

A.1.2 Définition de l'attribut Frame (cadre)

L'exemple suivant représente la manière de définir la position et l'orientation géométriques de deux objets CAEX par le biais de l'attribut Frame mentionné à l'Article 6. Le document AML correspondant est constitué d'une InstanceHierarchy (hiérarchie d'instances) nommée "ExampleInstanceHierarchy" avec deux InternalElements "BaseFrame" et "ChildFrame" (voir Figure A.2).

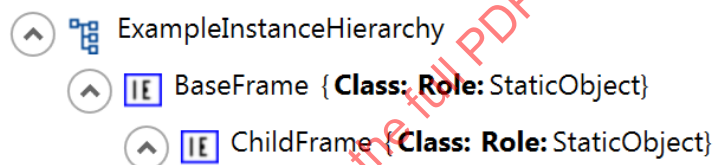


Figure A.2 – Deux cadres représentés dans l'InstanceHierarchy d'un document AML

Pour décrire la transformation complète de coordonnées de l'InternalElement "ChildFrame" par rapport à son nœud parent "BaseFrame", l'ancien objet détient l'attribut Frame. La Figure A.3 représente la partie pertinente du document AML et la manière de définir l'attribut Frame et ses sous-attributs.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <Version>1.0.0</Version>
  <InternalElement Name="BaseFrame" ID="d0235958-bce3-4b48-bec9-08d1033cc49e">
    <InternalElement Name="ChildFrame" ID="07218e99-f4d8-4dd9-ba92-16c6f0dd4f46">
      <Attribute Name="Frame">
        <Attribute Name="x" AttributeDataType="xs:double">
          <Value>1.00</Value>
        </Attribute>
        <Attribute Name="y" AttributeDataType="xs:double">
          <Value>1.00</Value>
        </Attribute>
        <Attribute Name="z" AttributeDataType="xs:double">
          <Value>1.00</Value>
        </Attribute>
        <Attribute Name="rx" AttributeDataType="xs:double">
          <Value>45.00</Value>
        </Attribute>
        <Attribute Name="ry" AttributeDataType="xs:double">
          <Value>45.00</Value>
        </Attribute>
        <Attribute Name="rz" AttributeDataType="xs:double">
          <Value>45.00</Value>
        </Attribute>
      </Attribute>
      <RoleRequirements RefBaseRoleClassPath="AutomationMLDMIRoleClassLib/./StaticObject"
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="AutomationMLDMIRoleClassLib/./StaticObject" />
  </InternalElement>
</InstanceHierarchy>

```

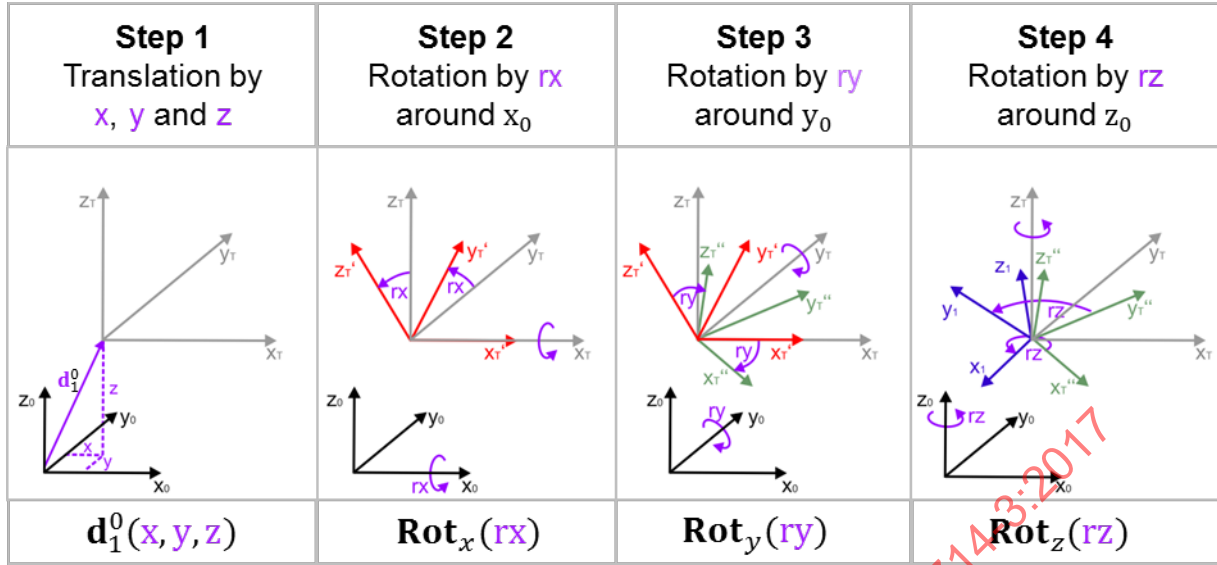
Figure A.3 – Représentation XML du document AML

En partant de l'hypothèse que le système de coordonnées $K_1(x_1, y_1, z_1)$ appartient à l'InternalElement "ChildFrame", la Figure A.4 représente sa translation relative par x , y et z ainsi que sa rotation plane par rx , ry et rz par rapport aux trois axes du système de coordonnées $K_0(x_0, y_0, z_0)$ spécifié par l'élément parent "BaseFrame". À l'aide de la matrice de transformation homogène H_1^0 et des six sous-attributs mentionnés dans le Tableau 6, la translation relative et l'orientation entre K_1 et K_0 peuvent être décrites dans un format compact, comme suit:

$$H_1^0(x, y, z, rx, ry, rz) = \begin{bmatrix} R_1^0 & d_1^0 \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{4 \times 4}. \quad (\text{A.1})$$

Ici, la matrice de rotation R_1^0 exprime l'orientation de K_1 par rapport à K_0 . L'Article 6 détermine l'ordre de rotation: rx , ry et rz . Dans le cadre de rotations planes, il est rappelé que les matrices de rotation appropriées doivent être multipliées en partant de la gauche. Ainsi, la partie rotative de H_1^0 est donnée par l'équation suivante

$$R_1^0 = \text{Rot}_z(rz) \text{Rot}_y(ry) \text{Rot}_x(rx). \quad (\text{A.2})$$



Anglais	Français
Step 1 Translation by x, y and z	Étape 1 Translation par x, y et z
Step 2 Rotation by rx around x ₀	Étape 2 Rotation par rx autour de x ₀
Step 3 Rotation by ry around y ₀	Étape 3 Rotation par ry autour de y ₀
Step 4 Rotation by rz around z ₀	Étape 4 Rotation par rz autour de z ₀

Figure A.4 – Translation et rotation plane

Chaque matrice de rotation est caractérisée par

$$Rot_x(rx) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(rx) & -\sin(rx) \\ 0 & \sin(rx) & \cos(rx) \end{bmatrix}, \quad Rot_y(ry) = \begin{bmatrix} \cos(ry) & 0 & \sin(ry) \\ 0 & 1 & 0 \\ -\sin(ry) & 0 & \cos(ry) \end{bmatrix} \text{ and}$$

$$Rot_z(rz) = \begin{bmatrix} \cos(rz) & -\sin(rz) & 0 \\ \sin(rz) & \cos(rz) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Le vecteur de déplacement est décrit par

$$d_1^0 = \begin{bmatrix} x \\ y \\ z \end{bmatrix}.$$

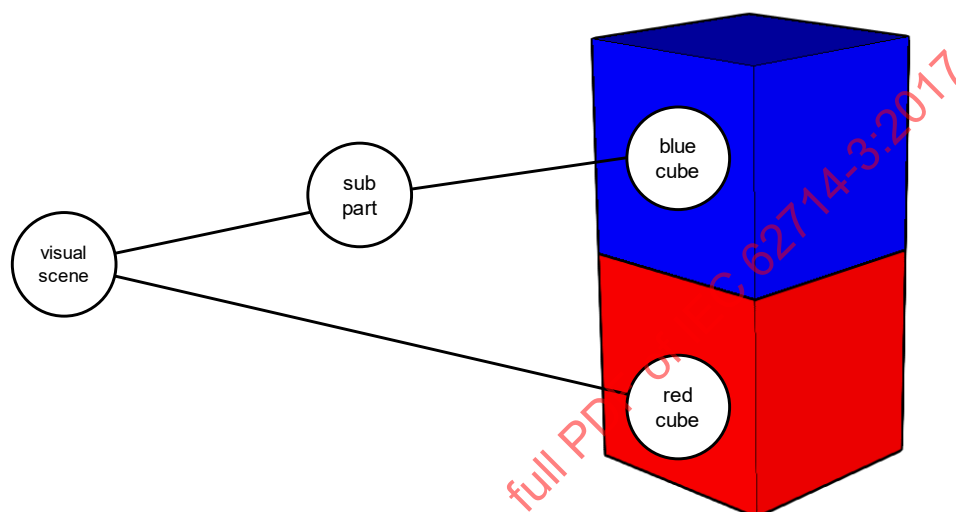
Comme représenté à la Figure A.4, l'origine du Frame translaté conserve ses coordonnées cartésiennes x, y et z au cours de la rotation plane. En remplaçant $\sin(ri) = sr_i$ et $\cos(ri) = cr_i$, l'équation (A.2) et la multiplication de l'équation (A.1) donnent le format compact

$$H_1^0 = \begin{bmatrix} cr_z cr_y & cr_z sr_y sr_x - sr_z cr_x & cr_z sr_y cr_x + sr_z sr_x & x \\ sr_z cr_y & sr_z sr_y sr_x + cr_z cr_x & sr_z sr_y cr_x - cr_z sr_x & y \\ -sr_y & cr_y sr_x & cr_y sr_x & z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

qui décrit la translation et l'orientation de l'InternalElement "ChildFrame" par rapport à son nœud parent "BaseFrame".

A.1.3 Structure des documents COLLADA

La scène utilisée dans les échantillons suivants est constituée de deux cubes, l'un au-dessus de l'autre. Dans la scène visuelle, le nœud de chaque cube est instancié. Chaque nœud référence le même élément géométrique, mais applique un nom ("RedCube", "BlueCube"), une matière (rouge, bleue) et une transformation différents. La Figure A.5 représente la scène obtenue avec le graphe de scène COLLADA correspondant.



Anglais	Français
visual scene	scène visuelle
sub part	sous-partie
blue cube	cube bleu
red cube	cube rouge

Figure A.5 – Scène COLLADA utilisée dans cet exemple

Dans cet exemple, deux documents COLLADA sont utilisés pour définir la scène visuelle complète. La hiérarchie des nœuds est définie dans le document COLLADA principal "red_blue_cubes.dae". La géométrie du cube est définie dans une bibliothèque dans le sous-document COLLADA séparé "cube.dae". Chaque nœud référence le sous-document "cube.dae" et instancie une géométrie de cube conformément aux règles d'instanciation et de référencement externe spécifiées dans l'ISO/PAS 17506. La Figure A.6 représente la structure et les références utilisées dans cet exemple.

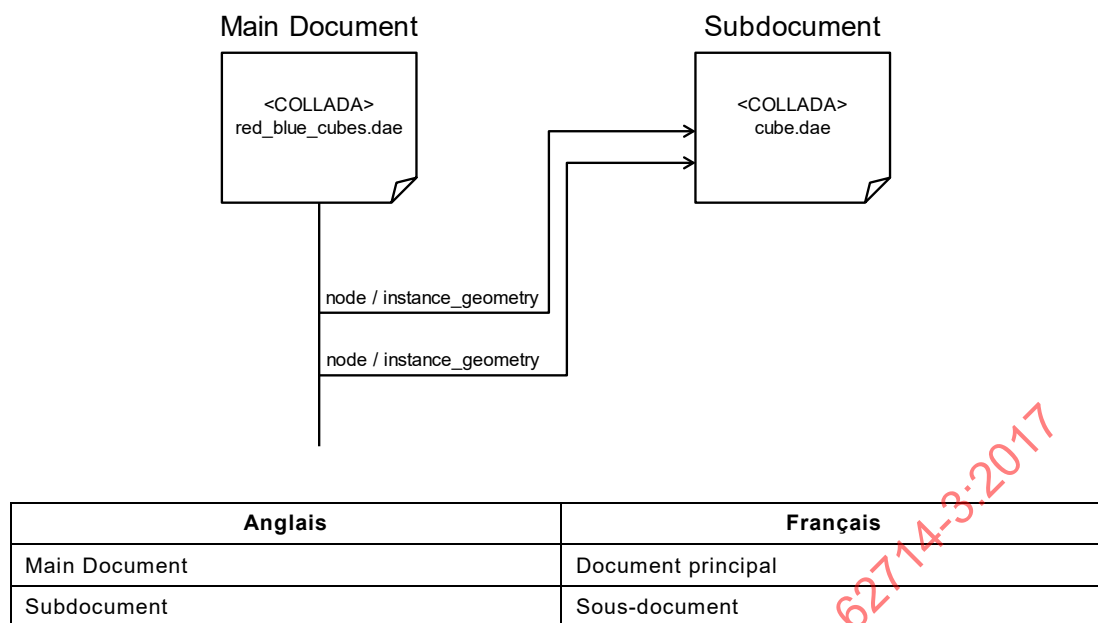


Figure A.6 – Structure et références

Le contenu du sous-document COLLADA “cube.dae” est représenté à la Figure A.7.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the geometry of a unit cube stored as reusable library
element</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_geometries>
    <geometry id="cube-geom" name="cube">
      <mesh>
        <source id="cube-geom-positions">
          <float_array id="cube-geom-positions-array" count="144"> 1 1 0 1 0 1 1 0 0 1 1 1
1 1 1 1 0 1 0 0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 0 1 1 0 0 0 0 0 1 0 0 1 1 1 0 0 0 1 0 1 1
1 1 1 1 0 1 1 1 1 0 0 1 0 1 0 1 0 1 1 0 0 1 1 1 1 1 1 1 0 1 0 1 1 0 0 1 1 0 0 1 0 0 0
0 0 1 0 0 1 1 0 1 0 0 0 1 0 0 0 1 1 0 0 0 0 0 1 0 1 0 0 1 0 0 1 1 0 0 0 0 0 1</float_array>
          <technique_common>
            <accessor count="48" source="#cube-geom-positions-array" stride="3">
              <param name="X" type="float" />
              <param name="Y" type="float" />
              <param name="Z" type="float" />
            </accessor>
          </technique_common>
        </source>
        <source id="cube-geom-normals">
          <float_array id="cube-geom-normals-array" count="144"> 1 0 0 1 0 0 1 0 0 1 0 0 -1
0 0 -1 0 0 -1 0 0 -1 0 0 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1
0 0 1 0 0 -1 0 0 -1 0 0 -1 0 0 0 1 0 0 1 0 0 1 0 0 1 0 0 1 0 0 -1 0 0 -1 0 0 -1 0 0 -1 0
0 0 -1 0 0 -1 0 0 -1 0 0 1 0 0 1 0 0 1 0 0 1 0 -1 0 0 -1 0 0 -1 0 0 -1 0 0 1 0 0 1 0 0 1 0 0
1 0 0</float_array>
          <technique_common>
            <accessor count="48" source="#cube-geom-normals-array" stride="3">
              <param name="X" type="float" />
              <param name="Y" type="float" />
              <param name="Z" type="float" />
            </accessor>
          </technique_common>
        </source>
        <vertices id="cube-geom-vertices">
          <input semantic="POSITION" source="#cube-geom-positions" />
          <input semantic="NORMAL" source="#cube-geom-normals" />
        </vertices>
        <triangles count="24" material="mat">
          <input offset="0" semantic="VERTEX" source="#cube-geom-vertices" />
          <p>0 1 2 1 0 3 4 5 6 7 6 5 8 9 10 9 8 11 12 13 14 15 14 13 16 17 18 17 16 19 20 21
22 23 22 21 24 25 26 25 24 27 28 29 30 31 30 29 32 33 34 33 32 35 36 37 38 39 38 37 40 41 42
41 40 43 44 45 46 47 46 45</p>
        </triangles>
      </mesh>
    </geometry>
  </library_geometries>
</COLLADA>

```

Anglais	Français
This document defines the geometry of a unit cube stored as reusable library element	Le présent document définit la géométrie d'un cube archivé comme élément de bibliothèque réutilisable

Figure A.7 – Contenu du document COLLADA cube.dae

Le document principal ("red_blue_cubes.dae") comprenant la hiérarchie des nœuds et les références externes à la géométrie du cube ("cube.dae") est représenté à la Figure A.8. Chaque géométrie instanciée écrase le symbole matière avec sa propre définition matière.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the node hierarchy and instantiates cubes from
subdocuments</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="red-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0 0 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="blue-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0 0 1 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="red" name="Red">
      <instance_effect url="#red-fx" />
    </material>
    <material id="blue" name="Blue">
      <instance_effect url="#blue-fx" />
    </material>
  </library_materials>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="redcube" name="RedCube">
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#red" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
      <node id="subpart">
        <translate>0 0 1</translate>
        <node name="bluecube" sid="bluecube">
          <instance_geometry url="./cube.dae#cube-geom">
            <bind_material>
              <technique_common>
                <instance_material symbol="mat" target="#blue" />
              </technique_common>
            </bind_material>
          </instance_geometry>
        </node>
      </node>
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#visualscene" />
  </scene>
</COLLADA>

```

Anglais	Français
This document defines the node hierarchy and instantiates cubes from subdocuments	Le présent document définit la hiérarchie des nœuds et instancie des cubes à partir de sous-documents

Figure A.8 – Contenu du document COLLADA red_blue_cubes.dae

A.1.4 Référencement à l'aide de l'URI et de fragments sans cible et sans ID

L'exemple de document AML suivant est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec un InternalElement "RedCube" unique. Ce dernier comprend une ExternalInterface CAEX de type "COLLADAInterface" nommée "FileLinkExplicit", qui fait partie de la bibliothèque InterfaceClassLibrary "AutomationMLInterfaceClassLib" standard, comme représenté à la Figure A.9.

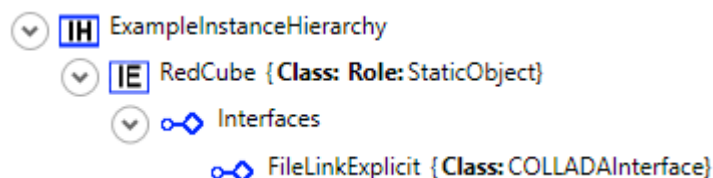


Figure A.9 – "RedCube" – Hiérarchie du document AML

La Figure A.10 représente la partie pertinente du document AML et la manière de définir les attributs pour le référencement.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#redcube</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>
```

Figure A.10 – Représentation XML du document AML

L'ExternalInterface "FileLinkExplicit" renvoie au document COLLADA, qui est représenté à la Figure A.11. Le cube rouge est référencé de manière non ambiguë par l'utilisation de l'URI relatif "./red_blue_cubes.dae#redcube".

```
<visual_scene id="visualscene">
  <node id="redcube" name="RedCube">
    <instance_geometry url="./cube.dae#cube-geom">
      <bind_material>
        <technique common>
          <instance_material symbol="mat" target="#red" />
        </technique common>
      </bind_material>
    </instance_geometry>
  </node>
  <node id="subpart"> [...] </node>
</visual_scene>
```

Figure A.11 – Référencement du cube rouge par ID

A.1.5 Référencement à l'aide de l'URI et de fragments comprenant une cible sans ID

Le document AML est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec un InternalElement "BlueCube" unique. Ce dernier comprend une ExternalInterface "COLLADAInterface" nommée "FileLinkExplicit", qui fait partie de la bibliothèque InterfaceClassLibrary "AutomationMLInterfaceClassLib" standard, comme représenté à la Figure A.12.

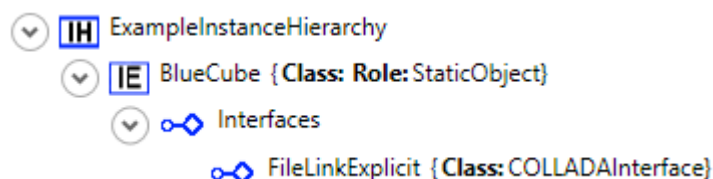


Figure A.12 – “BlueCube” – Hiérarchie du document AML

La Figure A.13 représente la partie pertinente du document AML et la manière de définir les attributs pour le référencement.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="BlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#subpart</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./bluecube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>
  
```

Figure A.13 – Représentation XML du document AML

L’ExternalInterface “FileLinkExplicit” renvoie au document COLLADA, qui est représenté à la Figure A.14. Tout d’abord, le point d’entrée du SID est spécifié par l’URI relatif “./red_blue_cubes.dae#subpart”. L’élément cible défini dans le document COLLADA nommé “bluecube” est ensuite résolu par exploration depuis l’élément référencé “subpart”, conformément aux règles d’un SID.

```

<visual_scene id="visualscene">
  <node id="redcube" name="RedCube"> [...] </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube"> [...] </node>
  </node>
</visual_scene>
  
```

Figure A.14 – Référencement du cube bleu

A.1.6 Référencement à l’aide d’un URI sans fragment comprenant une cible et un ID

Le document AML est constitué d’une InstanceHierarchy nommée “ExampleInstanceHierarchy” avec un InternalElement “BlueCube” unique. Ce dernier comprend une ExternalInterface “COLLADAInterface” nommée “FileLinkExplicit”, qui fait partie de la bibliothèque InterfaceClassLibrary “AutomationMLInterfaceClassLib” standard, comme représentée à la Figure A.15.

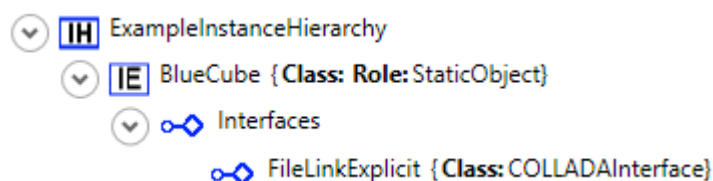


Figure A.15 – Hiérarchie du document AML

La Figure A.16 représente la partie pertinente du document AML et la manière de définir les attributs pour le référencement.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="BlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>subpart/bluecube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>

```

Figure A.16 – Représentation XML du document AML

L'ExternalInterface "FileLinkExplicit" renvoie au document COLLADA qui est représenté à la Figure A.17. Tout d'abord, le point d'entrée du SID est spécifié par l'URI relatif "./red_blue_cubes.dae". L'élément cible défini dans le document COLLADA nommé "bluecube" est ensuite résolu par exploration depuis le point d'entrée du SID "subpart/bluecube".

```

<visual_scene id="visualscene">
  <node id="redcube" name="RedCube"> [...] </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube"> [...] </node>
  </node>
</visual_scene>

```

Figure A.17 – Référencement du cube bleu à partir de l'élément "subpart"

A.1.7 Référencement à l'aide de l'URI et de fragments comprenant une cible et un ID

Le document AML est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec un InternalElement "BlueCube" unique. Ce dernier comprend une ExternalInterface "COLLADAInterface" nommée "FileLinkExplicit", qui fait partie de la bibliothèque InterfaceClassLibrary "AutomationMLInterfaceClassLib" standard, comme représenté à la Figure A.18.

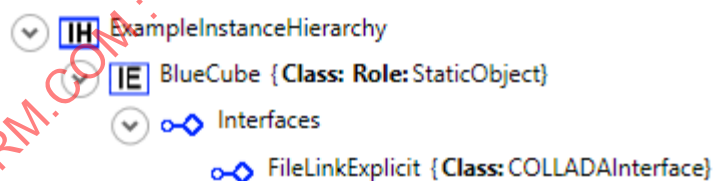


Figure A.18 – Hiérarchie du document AML

La Figure A.19 représente la partie pertinente du document AML et la manière de définir les attributs pour le référencement.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="BlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#redcube</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>subpart/bluecube</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>

```

Figure A.19 – Représentation XML du document AML

L'ExternalInterface "FileLinkExplicit" renvoie au document COLLADA qui est représenté à la Figure A.20. Dans cet exemple, le cube bleu est adressé comme déjà représenté en A.1.6. Le point d'entrée du SID est d'abord spécifié par l'URI relatif "./red_blue_cubes.dae#redcube". À ce stade, il est démontré que le fragment de l'URI spécifié n'est pas réellement nécessaire pour le référencement du cube bleu. Dans ce cas, le cube bleu est résolu par l'évaluation du SID commençant par un ID. Étant donné que l'ID de tout SID doit être unique dans l'intégralité du document, le référencement proposé n'est pas ambigu non plus.

```

<visual_scene id="visualscene">
  <node id="redcube" name="RedCube"> [...] </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube"> [...] </node>
  </node>
</visual_scene>

```

Figure A.20 – Référencement du cube bleu

A.1.8 Référencement à l'aide de l'URI sans fragment, sans cible et sans ID

Le document AML est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec un InternalElement "RedAndBlueCube" unique. Ce dernier comprend une ExternalInterface "COLLADAInterface" nommée "FileLinkExplicit", qui fait partie de la bibliothèque InterfaceClassLibrary "AutomationMLInterfaceClassLib" standard, comme représenté à la Figure A.21.

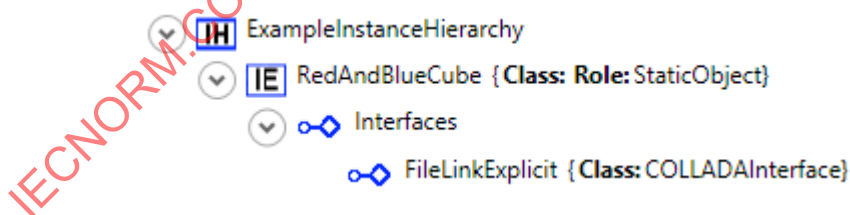


Figure A.21 – Hiérarchie du document AML

La Figure A.22 représente la partie pertinente du document AML et la manière de définir les attributs pour le référencement.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCube" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>

```

Figure A.22 – Représentation XML du document AML

L'ExternalInterface "FileLinkExplicit" renvoie au document COLLADA qui est représenté à la Figure A.23. Dans cet échantillon, l'élément COLLADA "scene" complet est adressé. Le cube bleu et le cube rouge sont tous les deux référencés de manière non ambiguë par l'utilisation de l'URI relatif "./red_blue_cubes.dae" sans fragment ou cible supplémentaire.

```

<COLLADA>
[... ]
<library_visual_scenes>
  <visual_scene id="visualscene">
    <node id="redcube" name="RedCube">
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#red" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
    <node id="subpart">
      <translate>0 0 1</translate>
      <node name="bluecube" sid="bluecube">
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#blue" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
    </node>
  </visual_scene>
</library_visual_scenes>
<scene>
  <instance_visual_scene url="#visualscene" />
</scene>
</COLLADA>

```

Figure A.23 – Référencement de la scène COLLADA complète

A.2 Référencement implicite des éléments COLLADA

A.2.1 Généralités

L'Article A.2 décrit la méthode de référencement implicite des modèles géométriques et cinématiques situés dans des documents COLLADA externes. Le référencement implicite est utilisé pour référencer un objet COLLADA à partir d'un objet CAEX et pour obtenir des informations complémentaires relatives à l'objet COLLADA au niveau CAEX, par exemple un cadre fournissant des possibilités d'adjonction. Dans ce cas, l'objet COLLADA est déjà connu par rapport à un contexte différent, par exemple un référencement explicite (voir l'Article A.1). Le Paragraphe A.2.2 décrit le référencement implicite d'un document COLLADA unique, tandis qu'A.2.3 décrit le référencement implicite des documents COLLADA imbriqués. Le Paragraphe A.2.4 décrit la manière de publier des cadres d'éléments respectifs d'un document COLLADA.

A.2.2 Référencement implicite

L'échantillon suivant applique la scène visuelle de A.1.3 (document COLLADA "red_blue_cubes.dae"). Le document AML est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec trois InternalElements, comme représenté à la Figure A.24. L'InternalElement "RedAndBlueCubeExplicit" comprend une ExternalInterface "COLLADAInterface" nommée "FileLinkExplicit". Les deux InternalElements "BlueCubeImplicit" et "RedCubeImplicit" comprennent une ExternalInterface "COLLADAInterface" nommée "FileLinkImplicit", qui fait partie de la bibliothèque InterfaceClassLibrary "AutomationMLInterfaceClassLib" standard.

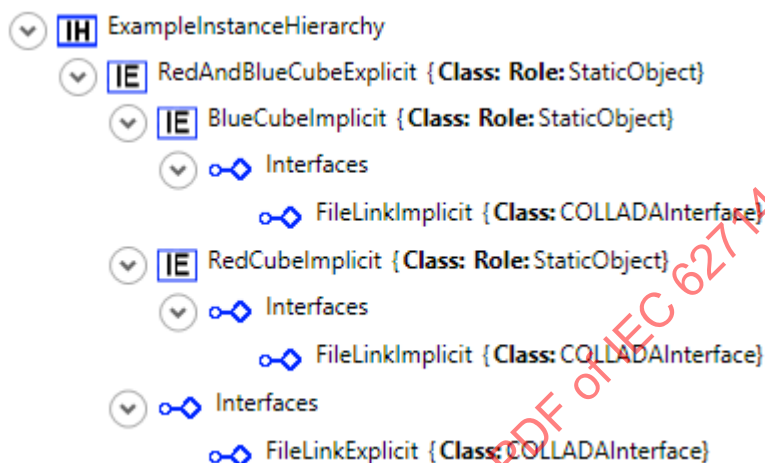


Figure A.24 – Référencement implicite: Hiérarchie du document AML

L'ExternalInterface "FileLinkExplicit" de l'InternalElement "RedAndBlueCubeExplicit" renvoie au document COLLADA et adresse la scène COLLADA complète de manière explicite (voir A.1.8). Le cube rouge et le cube bleu sont tous les deux référencés de manière non ambiguë par l'utilisation de l'URI relatif "./red_blue_cubes.dae" sans fragment ou cible supplémentaire.

Les InternalElements "RedCubeImplicit" et "BlueCubeImplicit" référencent le cube rouge et le cube bleu de manière implicite. La Figure A.25 représente la partie pertinente du document AML et la manière de définir les attributs pour le référencement implicite.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="BlueCubeImplicit" ID="GUID5">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#bluecube</Value>
      </Attribute>
    </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InternalElement>
<InternalElement Name="RedCubeImplicit" ID="GUID3">
  <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
    <Attribute Name="refType" AttributeDataType="xs:string">
      <Value>implicit</Value>
    </Attribute>
    <Attribute Name="refURI" AttributeDataType="xs:anyURI">
      <Value>./red_blue_cubes.dae#redcube</Value>
    </Attribute>
  </ExternalInterface>
  <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InternalElement>
<RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InternalElement>
</InstanceHierarchy>

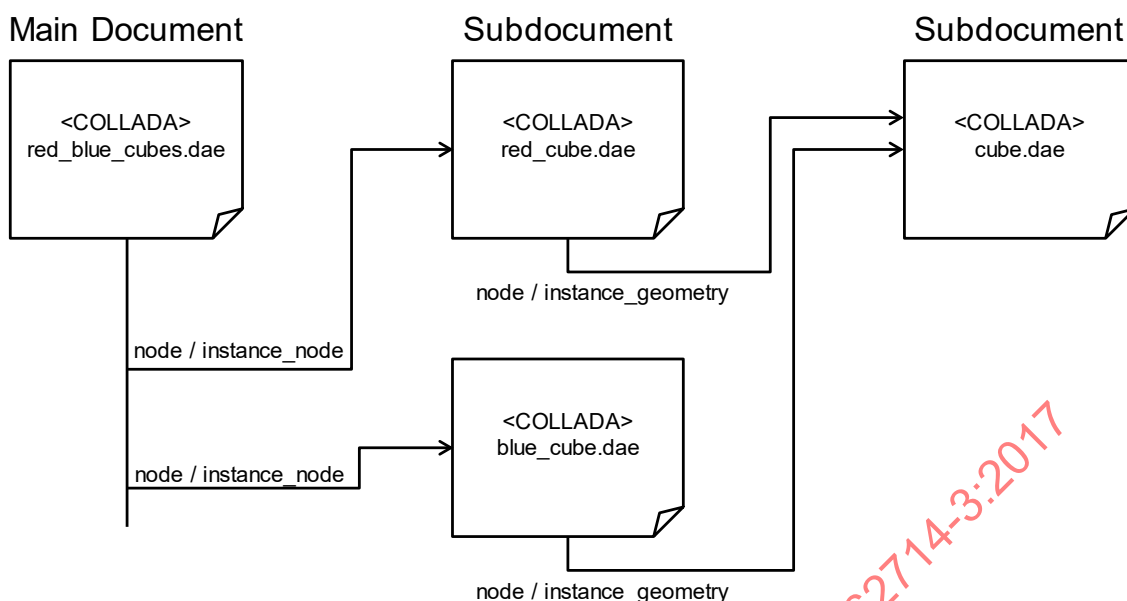
```

Figure A.25 – Représentation XML du document AML

Lors d'une visualisation, il n'est pas nécessaire d'obtenir un nouveau rendu du cube rouge et du cube bleu car ils ont déjà été visualisés lors du référencement explicite de l'élément "RedAndBlueCubeExplicit". Dans ce cas, la scène référencée de manière explicite ("RedAndBlueCubeExplicit") est visualisée.

A.2.3 Référencement implicite de sous-documents COLLADA

Selon le référencement externe, tel que défini dans l'ISO/PAS 17506 ou dans le document COLLADA 1.4.1, la description géométrique d'une scène peut être divisée en documents COLLADA séparés. Dans ce cas, un document COLLADA principal référence les éléments archivés dans les sous-documents COLLADA référencés. Le Paragraphe A.2.3 décrit le référencement implicite des éléments COLLADA situés dans les sous-documents COLLADA et référencés à partir d'un document COLLADA principal. La Figure A.26 représente la structure et les relations utilisées dans cet exemple.



Anglais	Français
Main Document	Document principal
Subdocument	Sous-document

Figure A.26 – Structure et relations des sous-documents COLLADA référencés

L'échantillon suivant applique la scène visuelle de A.1.3, la structure des documents COLLADA étant modifiée. Le document COLLADA principal ("red_blue_cubes.dae") est constitué de deux nœuds, chacun d'entre eux instanciant un nœud situé dans un sous-document COLLADA ("red_cube.dae" et "blue_cube.dae"). Ces sous-documents COLLADA sont constitués d'une bibliothèque qui définit un nœud unique et une matière. Chaque nœud des deux documents instancie une géométrie de cube qui est définie dans un sous-document COLLADA ("cube.dae"). Dans cet exemple, tandis que la définition de la géométrie reste la même que celle utilisée en A.1.3, le document COLLADA principal ("red_blue_cubes.dae") est modifié et deux sous-documents COLLADA supplémentaires sont introduits ("red_cube.dae" et "blue_cube.dae"). La Figure A.27 représente le contenu du document COLLADA principal modifié ("red_blue_cubes.dae").

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the node hierarchy and instantiates nodes from
subdocuments</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="redcube" name="RedCubeInstance">
        <instance_node sid="redcuberef" url="./red_cube.dae#redcube" />
      </node>
      <node id="bluecube" name="blueCubeInstance">
        <translate>0 0 1</translate>
        <instance_node sid="bluecuberef" url="./blue_cube.dae#bluecube" />
      </node>
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#visualscene" />
  </scene>
</COLLADA>

```

Anglais	Français
This document defines the node hierarchy and instantiates nodes from subdocuments	Le présent document définit la hiérarchie des nœuds et instancie des nœuds à partir de sous-documents

Figure A.27 – Contenu du document COLLADA modifié red_blue_cubes.dae

La scène visuelle est constituée de deux éléments de nœuds (redcube et bluecube), chacun instanciant un nœud dans un sous-document ("red_cube.dae" et "blue_cube.dae"). Le contenu des deux sous-documents COLLADA est représenté à la Figure A.28 et à la Figure A.29.

```
<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines a node which instantiates a cube geometry from a
COLLADA subdocument and applies a red material</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="red-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0 0 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="red" name="Red">
      <instance_effect url="#red-fx" />
    </material>
  </library_materials>
  <library_nodes>
    <node id="redcube" name="RedCube">
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#red" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
  </library_nodes>
</COLLADA>
```

Anglais	Français
This document defines a node which instantiates a cube geometry from a COLLADA subdocument and applies a red material	Le présent document définit un nœud qui instancie une géométrie de cube à partir d'un sous-document COLLADA et applique une matière rouge

Figure A.28 – Contenu du document COLLADA red_cube.dae

Le document COLLADA "red_cube.dae" définit un nœud nommé "RedCube" situé dans une bibliothèque et pouvant être référencé et instancié. En outre, une matière rouge est définie pour adjoindre la géométrie instanciée dans le sous-document COLLADA "cube.dae". De manière similaire, le nœud et la géométrie du cube bleu nommé "BlueCube" sont définis dans le sous-document COLLADA "blue_cube.dae", comme représenté à la Figure A.29.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines a node which instantiates a cube geometry from a
COLLADA subdocument and applies a blue material</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="blue-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0 0 1 1 </color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="blue" name="Blue">
      <instance_effect url="#blue-fx" />
    </material>
  </library_materials>
  <library_nodes>
    <node id="bluecube" name="BlueCube">
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#blue" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
  </library_nodes>
</COLLADA>

```

Anglais	Français
This document defines a node which instantiates a cube geometry from a COLLADA subdocument and applies a blue material	Le présent document définit un nœud qui instancie une géométrie de cube à partir d'un sous-document COLLADA et applique une matière bleue

Figure A.29 – Contenu du document COLLADA blue_cube.dae

Pour référencer le cube bleu ou le cube rouge dans les sous-documents COLLADA à partir d'un document AML, le chemin commençant à partir du document COLLADA racine doit être spécifié dans son intégralité. L'exemple suivant présente un extrait d'un document AML pour démontrer le référencement implicite d'éléments situés dans des sous-documents COLLADA référencés.

Le document AML est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec trois InternalElements, comme représenté à la Figure A.30. Tous les InternalElements comprennent une ExternalInterface de type "COLLADAInterface", qui fait partie de la bibliothèque InterfaceClassLibrary "AutomationMLInterfaceClassLib" standard.

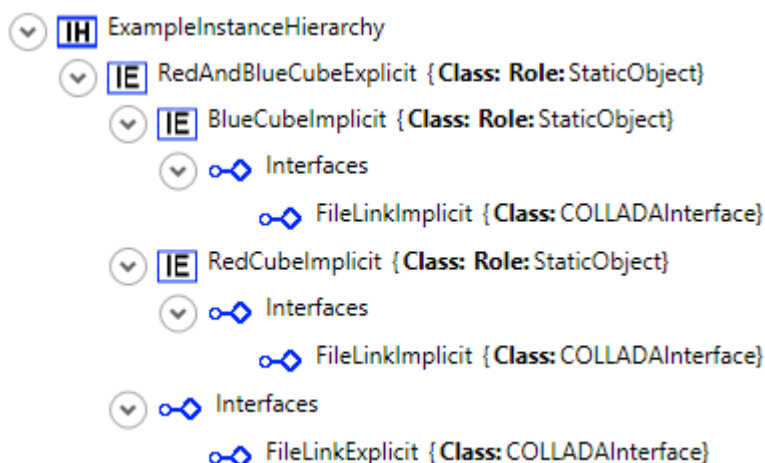


Figure A.30 – Référencement implicite: Hiérarchie du document AML

L'ExternalInterface "FileLinkExplicit" de l'InternalElement "RedAndBlueCubeExplicit" renvoie au document COLLADA et adresse la scène COLLADA complète de manière explicite (voir A.1.8). Le cube rouge et le cube bleu sont tous les deux référencés de manière non ambiguë par l'utilisation de l'URI relatif "./red_blue_cubes.dae" sans fragment ou cible supplémentaire. Les InternalElements "RedCubeImplicit" et "BlueCubeImplicit" référencent le cube rouge et le cube bleu de manière implicite, les deux cubes étant situés dans un sous-document COLLADA. La Figure A.31 représente la partie pertinente du document AML pour un référencement implicite.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
    <InternalElement Name="RedCubeImplicit" ID="GUID3">
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./red_cube.dae</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>redcube/redcuberef</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
    </InternalElement>
    <InternalElement Name="BlueCubeImplicit" ID="GUID5">
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./blue_cube.dae</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>bluecube/bluecuberef</Value>
        </Attribute>
      </ExternalInterface>
      <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
</InstanceHierarchy>
```

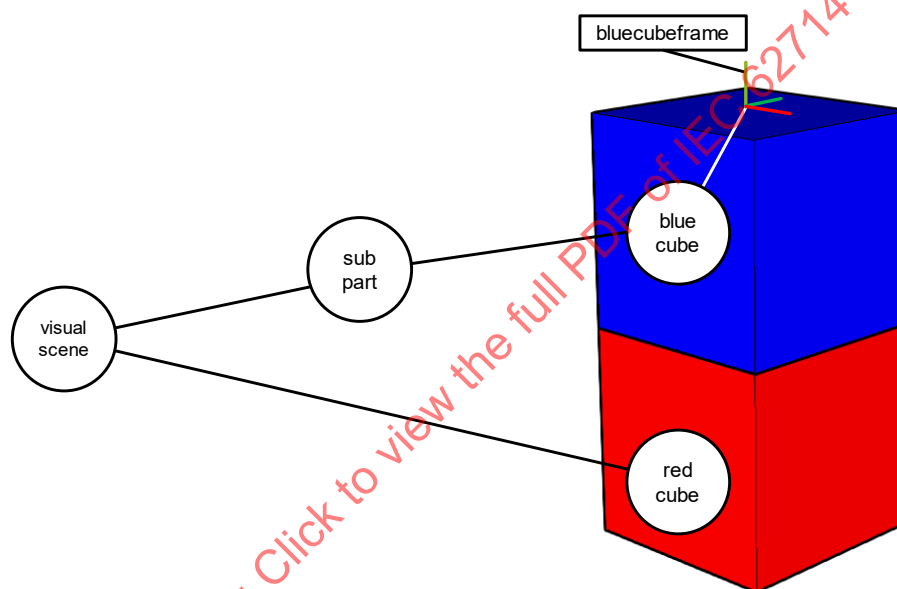
Figure A.31 – Représentation XML du document AML

A.2.4 Publication d'éléments d'un document COLLADA en CAEX

Le Paragraphe A.2.4 présente un cas d'utilisation dans lequel les deux méthodes de référencement (explicite et implicite) sont applicables. Le présent paragraphe décrit la manière de publier des cadres de documents COLLADA et leur référencement dans un document AML.

Les éléments géométriques et cinématiques COLLADA peuvent être publiés au niveau CAEX afin d'être prêts à l'utilisation. Cela permet leur référencement et affecte d'autres informations. Dans cet exemple, le document COLLADA et sa scène visuelle issue de A.1.3 (document COLLADA "red_blue_cubes.dae") sont utilisés avec un élément COLLADA supplémentaire. Le nœud supplémentaire nommé "bluecubeframe" est adjoint au cube bleu et représente un cadre.

La Figure A.32 représente la scène visuelle modifiée ainsi que le graphe correspondant de la scène COLLADA.



Anglais	Français
bluecubeframe	cadre du cube bleu
visual scene	scène visuelle
sub part	sous-partie
blue cube	cube bleu
red cube	cube rouge

Figure A.32 – Scène COLLADA modifiée avec nœud supplémentaire

La partie pertinente du document COLLADA, notamment le nœud cadre supplémentaire, est représentée à la Figure A.33.

```
<visual_scene id="visualscene">
  <node id="redcube" name="RedCube">
    <instance_geometry url="./cube.dae#cube-geom">
      <bind_material>
        <technique_common>
          <instance_material symbol="mat" target="#red" />
        </technique_common>
      </bind_material>
    </instance_geometry>
  </node>
  <node id="subpart">
    <translate>0 0 1</translate>
    <node name="bluecube" sid="bluecube">
      <node sid="bluecubeframe">
        <translate>0.5 0.5 1.0</translate>
      </node>
      <instance_geometry url="./cube.dae#cube-geom">
        <bind_material>
          <technique_common>
            <instance_material symbol="mat" target="#blue" />
          </technique_common>
        </bind_material>
      </instance_geometry>
    </node>
  </node>
</visual_scene>
```

Figure A.33 – Élément cadre supplémentaire du document COLLADA

Dans cet exemple, le document AML est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec un InternalElement "RedAndBlueCubeExplicit" qui comprend une ExternalInterface "COLLADAInterface" nommée "FileLinkExplicit". Cette ExternalInterface est utilisée pour référencer la scène complète de manière explicite, comme représenté à la Figure A.34.

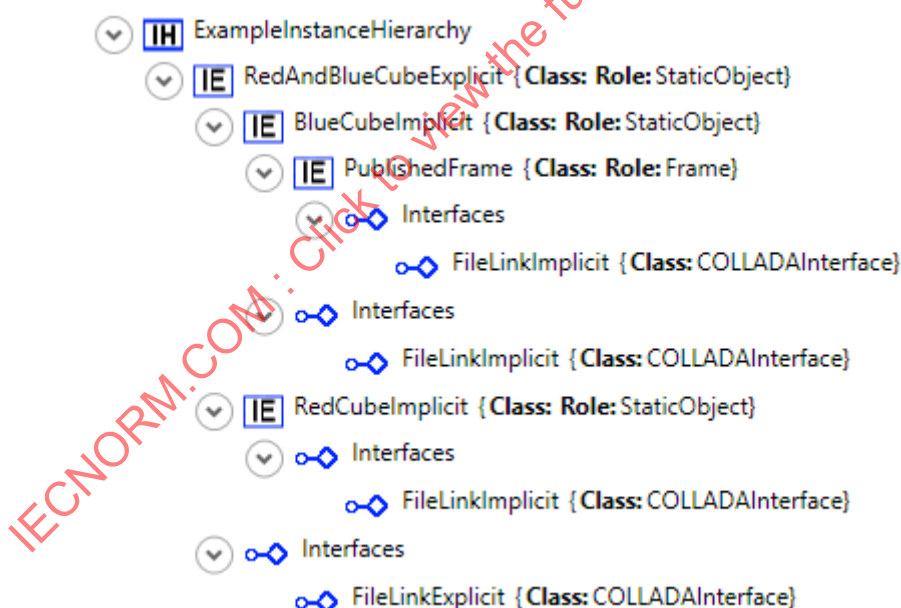


Figure A.34 – Cadres de publication: Hiérarchie du document AML

L'InternalElement "RedAndBlueCubeExplicit" a deux InternalElements nommés "BlueCubeImplicit" et "RedCubeImplicit". L'InternalElement "BlueCubeImplicit" a un InternalElement "PublishedFrame" auquel la RoleClass "Frame" est affectée. L'ExternalElement "FileLinkImplicit" de l'InternalElement "PublishedFrame" référence le nœud COLLADA "bluecubeframe" dans le nœud de document COLLADA de manière implicite.

La Figure A.35 représente la partie pertinente du document AML et la manière de définir les attributs pour le référencement implicite.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae</Value>
      </Attribute>
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="BlueCubeImplicit" ID="GUID3">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#subpart</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./bluecube</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="PublishedFrame" ID="GUID5">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./red_blue_cubes.dae#subpart</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./bluecube/bluecubeframe</Value>
      </Attribute>
      <Attribute Name="AdditionalInformation1" AttributeDataType="xs:token">
        <Value>AdditionalAttributeValue</Value>
      </Attribute>
      <Attribute Name="AdditionalInformation2" AttributeDataType="xs:token">
        <Value>AdditionalAttributeValue</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_FRAME" />
  </InternalElement>
  <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
</InstanceHierarchy>

```

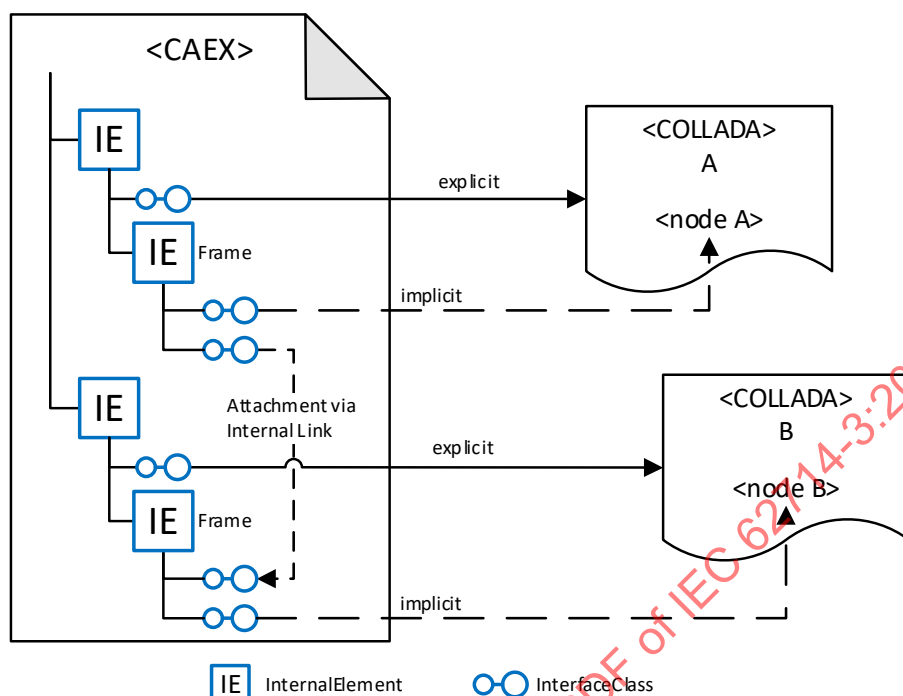
Figure A.35 – Représentation XML du document AML

NOTE Un objet CAEX publié avec la RoleClass "Frame" affectée ne doit pas renvoyer à un objet géométrique COLLADA (il peut renvoyer à un objet COLLADA sans géométrie). Dans le cas précédent, les informations relatives à la transformation proviennent de l'objet COLLADA et sont appliquées à l'objet CAEX publié auquel la RoleClass "Frame" est affectée, même si l'objet cadre CAEX fournit un attribut cadre CAEX. Si l'objet COLLADA ne donne lieu à aucune transformation, celle de l'attribut cadre de l'objet CAEX auquel la RoleClass "Frame" est affectée est utilisée.

A.3 Adjonction entre objets en CAEX

L'Article A.3 décrit la manière d'adjoindre un objet géométrique AML à un autre objet géométrique AML (tous les deux de type InternalElement). Les deux objets AML ont une référence à des objets géométriques COLLADA quelconques par le biais de références explicites. En outre, les deux objets AML ont un enfant InternalElement Frame auquel la RoleClass "Frame" est affectée. Chaque InternalElement Frame a une ExternalInterface implicite déduite de l'InterfaceClass "COLLADAInterface" qui référence un nœud défini dans le document COLLADA. En outre, chaque InternalElement Frame a une ExternalInterface

déduite de l'InterfaceClass "AttachmentInterface", qui peut être connectée ou liée. La structure est représentée à la Figure A.36.



Anglais	Français
Frame	Cadre
Attachment via Internal Link	Adjonction par le biais d'une liaison interne
Internal Element	Élément interne
explicit	explicite
implicit	implicite
node	nœud

Figure A.36 – Structure pour les adjonctions entre objets en CAEX

Dans cet exemple, le document COLLADA est utilisé avec sa scène visuelle issue du Paragraphe A.2.3 avec un nœud supplémentaire. Le nœud supplémentaire, qui est adjoint au cube bleu, représente le cadre publié. Un cube jaune supplémentaire est défini dans un document COLLADA séparé ("yellow_cube.dae") avec un nœud supplémentaire qui est publié. La connexion entre les deux nœuds supplémentaires (cadres d'adjonction) est établie au sein du document AML. La Figure A.37 représente la visualisation du cube jaune ("YellowCube") ainsi que son cadre supplémentaire.

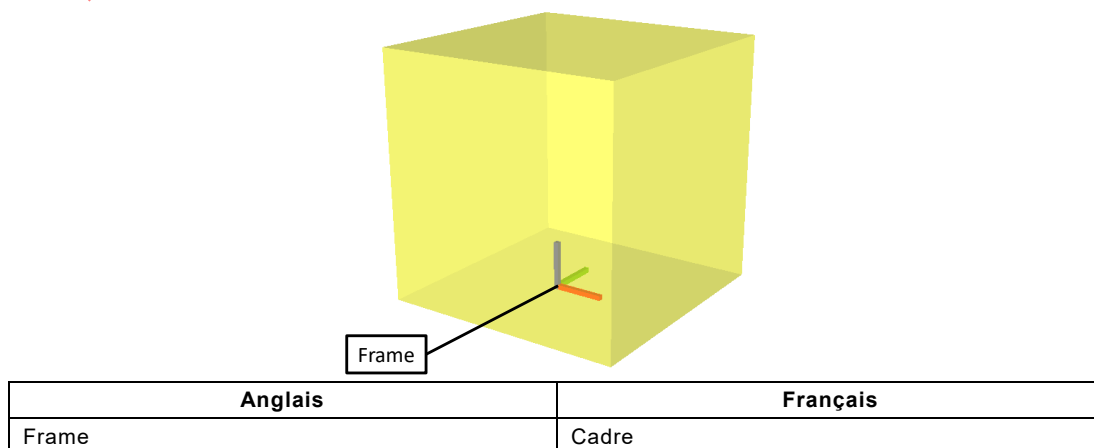


Figure A.37 – Visualisation du cube jaune avec un cadre supplémentaire

Le document COLLADA du cube jaune (“yellow_cube.dae”) est représenté dans son intégralité à la Figure A.38. Il applique la définition géométrique du document COLLADA externe “cube.dae” de A.1.3.

```
<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines a node which instantiates a cube geometry from a
COLLADA subdocument and applies a yellow material. An additional node is
provided.</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="yellow-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 1 0 0.5 </color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="yellow" name="Yellow">
      <instance_effect url="#yellow-fx" />
    </material>
  </library_materials>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="yellowcube" name="YellowCube">
        <translate>0 0 2</translate>
        <node sid="yellowcubeframe">
          <translate>0.5 0.5 0</translate>
        </node>
        <instance_geometry url="./cube.dae#cube-geom">
          <bind material>
            <technique_common>
              <instance_material symbol="mat" target="#yellow" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
    </visual_scene>
  </library_visual_scenes>
  <scene>
    <instance_visual_scene url="#visualscene" />
  </scene>
</COLLADA>
```

Anglais	Français
This document defines a node which instantiates a cube geometry from a COLLADA subdocument and applies a yellow material. An additional node is provided.	Le présent document définit un nœud qui instancie une géométrie de cube à partir d'un sous-document COLLADA et applique une matière jaune. Un nœud supplémentaire est fourni.

Figure A.38 – Document COLLADA du cube jaune avec un cadre supplémentaire

Le document AML est constitué d'une InstanceHierarchy nommée “ExampleInstanceHierarchy” avec deux InternalElements (“RedAndBlueCubeExplicit” et “YellowCubeExplicit”). Dans cet exemple, le cube jaune (“YellowCube”) provenant du document COLLADA séparé est adjoint au cube bleu (“BlueCube”). Les deux InternalElements comprennent une ExternalInterface “COLLADAInterface” nommée “FileLinkExplicit”, qui fait partie de la bibliothèque InterfaceClassLibrary “AutomationMLInterfaceClassLib” standard.

Chaque ExternalInterface nommée “FileLinkExplicit” des deux InternalElements renvoie aux documents COLLADA séparés et adresse la scène COLLADA complète de manière explicite (voir également A.1.8). En outre, les deux InternalElements ont un InternalElement (“PublishedFrame”) enfant auquel la RoleClass “Frame” est affectée et qui fait partie de la bibliothèque RoleClassLibrary standard également. Chaque InternalElement “PublishedFrame” a deux ExternalInterfaces: une ExternalInterface déduite de l’InterfaceClass “COLLADAInterface” pour référencer un nœud COLLADA de manière implicite (“FileLinkImplicit”) et une ExternalInterface déduite de l’InterfaceClass “AttachmentInterface” pour définir une adjonction (“Attachment”). La structure est représentée à la Figure A.39.

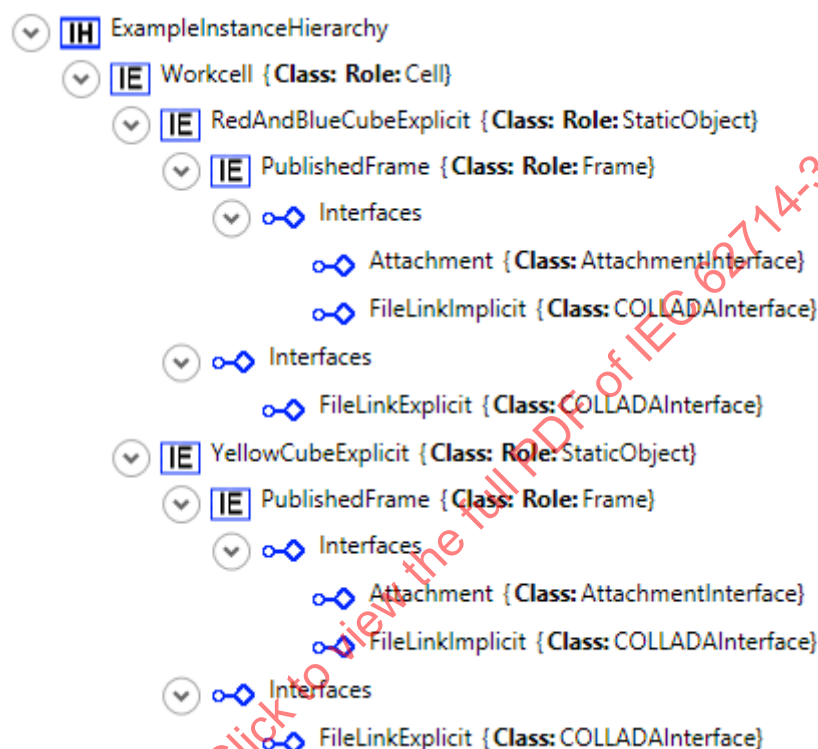


Figure A.39 – Hiérarchie du document AML

La Figure A.40 représente la partie pertinente du document AML et la manière de publier des cadres pour l’adjonction d’objets définis en AML.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="Workcell" ID="GUID0">
    <InternalElement Name="RedAndBlueCubeExplicit" ID="GUID1">
      <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./red_blue_cubes.dae</Value>
        </Attribute>
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>explicit</Value>
        </Attribute>
      </ExternalInterface>
    <InternalElement Name="PublishedFrame" ID="GUID3">
      <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID4" />
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID5">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./red_blue_cubes.dae#subpart</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./bluecube/bluecubeframe</Value>
        </Attribute>
      </InternalElement>
      <RoleRequirements RefBaseRoleClassPath="PATH_FRAME" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <InternalElement Name="YellowCubeExplicit" ID="GUID6">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID7">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./yellow_cube.dae</Value>
      </Attribute>
    </ExternalInterface>
    <InternalElement Name="PublishedFrame" ID="GUID8">
      <ExternalInterface Name="Attachment" RefBaseClassPath="PATH_AI" ID="GUID9" />
      <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID10">
        <Attribute Name="refType" AttributeDataType="xs:string">
          <Value>implicit</Value>
        </Attribute>
        <Attribute Name="refURI" AttributeDataType="xs:anyURI">
          <Value>./yellow_cube.dae#yellowcube</Value>
        </Attribute>
        <Attribute Name="target" AttributeDataType="xs:token">
          <Value>./yellowcube#frame</Value>
        </Attribute>
      </InternalElement>
      <RoleRequirements RefBaseRoleClassPath="PATH_FRAME" />
    </InternalElement>
    <RoleRequirements RefBaseRoleClassPath="PATH_SO" />
  </InternalElement>
  <InternalLink Name="AttachmentBlueMovesYellow" RefPartnerSideA="GUID3:Attachment" RefPartnerSideB="GUID7:Attachment" />
  <RoleRequirements RefBaseRoleClassPath="PATH5/Cell" />
</InternalElement>
</InstanceHierarchy>

```

Figure A.40 – Représentation XML du document AML

En conséquence, le cube jaune est adjoint au cube bleu, comme représenté à la Figure A.41. Par opposition au montage, l'adjonction ne comprend pas de transformation supplémentaire de l'objet adjoint. La connexion est unidirectionnelle. Si le cube bleu est translaté ou pivoté, la transformation du cube bleu affecte le cube jaune également.

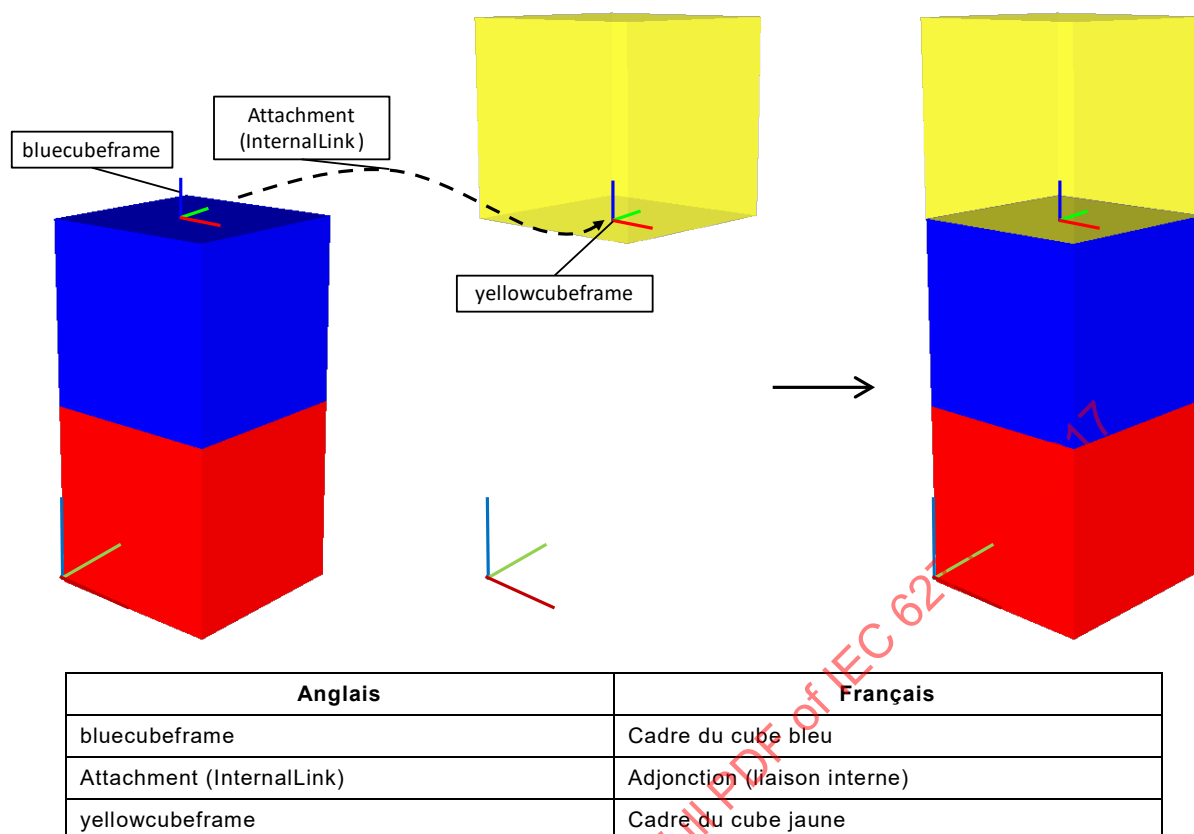


Figure A.41 – Adjonction entre des objets AML géométriques

Pour finir, la connexion est établie par le biais d'une InternalLink (liaison interne) qui connecte les deux cadres publiés. La Figure A.42 représente la partie pertinente du document AML et la manière de définir une InternalLink. La liaison est archivée dans une cellule de travail ("Workcell") de l'InternalElement CAEX qui est le parent commun le plus petit des objets CAEX connectés, conformément à l'IEC 62714-1.

```
<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="Workcell" ID="GUID0">
    [...]
    <InternalLink Name="AttachmentBlueRedMovesYellowCube"
      RefPartnerSideA="GUID3:Attachment"
      RefPartnerSideB="GUID8:Attachment" />
  </InternalElement>
</InstanceHierarchy>
```

Figure A.42 – Représentation XML du document AML

Annexe B (informative)

Modélisation des systèmes cinématiques et leur combinaison en AML

B.1 Généralités

Les articles suivants décrivent la modélisation des systèmes cinématiques et leur combinaison en AML. Dans ces articles, une scène est constituée par étapes d'une unité linéaire (Article B.2) et d'un robot (Article B.3) fixé à l'unité linéaire (Article B.4). En outre, un outil de préhension est défini et monté sur le robot (Article B.5). Pour finir, une pièce est adjointe à l'outil de préhension (Article B.6).

NOTE 1 Afin d'améliorer la lisibilité, tous les remplacements des attributs `RefBaseRoleClassPath` de l'Article A.1 s'appliquent.

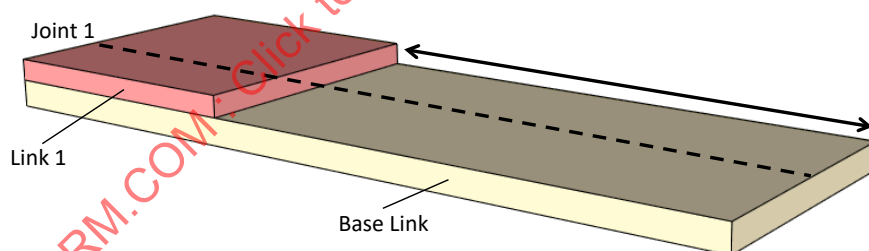
NOTE 2 Afin d'améliorer la lisibilité, tout attribut `RefBaseRoleClassPath` qui renvoie à la classe "AutomationMLMIRoleClassLib/ManufacturingEquipment/Robot" est remplacé par "PATH_ROB".

NOTE 3 Afin d'améliorer la lisibilité, tout attribut `RefBaseRoleClassPath` qui renvoie à la classe "AutomationMLMIRoleClassLib/ManufacturingEquipment/Carrier" est remplacé par "PATH_CAR".

B.2 Modélisation d'un document AML d'une unité linéaire en CAEX et COLLADA

B.2.1 Généralités

L'Article B.2 décrit la modélisation d'une unité linéaire. La scène visuelle est décrite en B.2.2. Dans les paragraphes qui suivent B.2.3 à B.2.6, la scène est complétée par étapes avec des éléments cinématiques. Pour finir, le document AML est défini et combiné avec le système cinématique de B.2.8.



Anglais	Français
Joint	Articulation
Link 1	Liaison 1
Base link	Liaison de base

Figure B.1 – Visualisation de l'unité linéaire

B.2.2 Définition de la scène visuelle

Dans cet exemple, la scène visuelle de l'unité linéaire est constituée de deux éléments géométriques l'un au-dessus de l'autre, comme représenté à la Figure B.1. Pour maintenir le contenu COLLADA à un niveau minimal, la géométrie de l'unité linéaire est représentée par de simples parallélépipèdes rectangulaires qui sont définis dans le document COLLADA "cube.dae", comme représenté à la Figure A.7. Chaque cube a une échelle différente et une matière qui lui est propre. Le cube le plus bas représente la liaison de base de l'unité linéaire alors que le cube le plus élevé représente le curseur.

La Figure B.2 représente le contenu du document COLLADA "linear_unit.dae", qui définit la représentation géométrique de l'unité.

```

<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the geometry and kinematic of a linear unit</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="base-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0.9 0.8 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="link-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>1 0.6 0.6 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
  </library_effects>
  <library_materials>
    <material id="base-mat" name="BaseMat">
      <instance_effect url="#base-fx" />
    </material>
    <material id="link-mat" name="LinkMat">
      <instance_effect url="#link-fx" />
    </material>
  </library_materials>
  <library_visual_scenes>
    <visual_scene id="visualscene">
      <node id="linearunit" name="LinearUnit">
        <node id="geombase" name="GeomBase">
          <scale>3.0 1.0 0.1</scale>
          <instance_geometry url="./cube.dae#cube-geom">
            <bind_material>
              <technique_common>
                <instance_material symbol="mat" target="#base-mat" />
              </technique_common>
            </bind_material>
          </instance_geometry>
          <node id="geomlink1" name="GeomLink1">
            <translate>0 0 1</translate>
            <scale>0.33 1 0.8</scale>
            <rotate sid="transform">0 0 0 0</rotate>
            <instance_geometry url="./cube.dae#cube-geom">
              <bind_material>
                <technique_common>
                  <instance_material symbol="mat" target="#link-mat" />
                </technique_common>
              </bind_material>
            </instance_geometry>
          </node>
        </node>
      </node>
    </visual_scene>
  </library_visual_scenes>
  <!-- Step 1: Definition of joint elements -->
  <!-- Step 2: Definition of kinematic model element -->
  <!-- Step 3: Definition of articulated system (kinematics) element -->
  <!-- Step 4: Definition of kinematics scene element -->
  <scene>
    <instance_visual_scene url="#visualscene" />
    <!-- Step 5: Instantiation of kinematics scene -->
  </scene>
</COLLADA>

```

Anglais	Français
This document defines the geometry and kinematic of a linear unit	Le présent document définit la géométrie et la cinématique de l'unité linéaire
Step 1: Definition of joint elements	Étape 1: Définition des éléments d'articulation
Step 2: Definition of kinematic model element	Étape 2: Définition de l'élément du modèle cinématique
Step 3: Definition of articulated system (kinematics) element	Étape 3: Définition de l'élément (cinématique) du système articulé
Step 4: Definition of kinematics scene element	Étape 4: Définition de l'élément de la scène cinématique
Step 5: Instantiation of kinematics scene	Étape 5: Instanciation de la scène cinématique

Figure B.2 – Définition de la scène visuelle

Chaque parallélépipède est affecté à un nœud (geombase et geomlink1). Dans les paragraphes B.2.3 à B.2.7, les éléments cinématiques exigés sont définis et appliqués à cette scène afin de compléter le système cinématique.

B.2.3 Définition de l'articulation

Dans ce cas, le système cinématique a une articulation prismatique unique nommée "Joint1", qui est définie dans une bibliothèque et qui est instanciée à une étape ultérieure. Les articulations prismatiques sont définies par l'élément "prismatic". En outre, la direction de l'axe de l'articulation et les valeurs limites minimale et maximale facultatives sont spécifiées conformément aux règles spécifiées dans l'ISO/PAS 17506 ou le document COLLADA 1.4.1. L'extrait du document COLLADA est représenté à la Figure B.3.

NOTE Il est possible de spécifier les unités physiques conformément à l'ISO/PAS 17506 ou au document COLLADA 1.4.1. Les degrés (angles) et les mètres (distances) sont utilisables par défaut. Il est possible de redéfinir ces derniers dans l'élément "unit".

```
<library joints>
  <joint id="joint1" name="Joint1">
    <prismatic sid="axis0">
      <axis>0 0 1</axis>
      <limits>
        <min>0</min>
        <max>3</max>
      </limits>
    </prismatic>
  </joint>
</library joints>
```

Figure B.3 – Définition de l'articulation

B.2.4 Définition du modèle cinématique

Le Paragraphe B.2.4 définit le modèle cinématique et sa hiérarchie. Tout d'abord, toutes les articulations du modèle cinématique sont définies ou instanciées à partir d'une bibliothèque d'articulations. La chaîne cinématique suivante est constituée de liaisons et d'une articulation ainsi que de leur relation hiérarchique. Toute liaison ou articulation peut donner lieu à une transformation. La transformation d'une liaison ou d'une articulation dépend toujours de son prédécesseur et est décrite par les éléments "translate" et "rotate". Le modèle cinématique est défini dans une bibliothèque et est instancié à une étape ultérieure. La Figure B.4 représente la partie pertinente du document COLLADA.

```

<library_kinematics_models>
  <kinematics_model id="linearunit_kinmodel">
    <technique_common>
      <instance_joint sid="inst_joint1" url="#joint1" />
      <link name="BaseLink" sid="baselink">
        <attachment_full joint="linearunit_kinmodel/inst_joint1">
          <rotate>0 1 0 90</rotate>
          <link name="Link1" sid="link1" />
        </attachment_full>
      </link>
    </technique_common>
  </kinematics_model>
</library_kinematics_models>

```

Figure B.4 – Définition de modèle cinématique

Dans cet exemple, deux liaisons nommées “BaseLink” et “Link1” sont définies. Celles-ci sont connectées par une articulation unique, qui référence une articulation définie dans la bibliothèque, comme décrit dans le paragraphe précédent.

B.2.5 Définition du système articulé

Les propriétés cinématiques et dynamiques sont réunies dans les éléments “articulated_system”. Étant donné que chaque définition de système articulé comporte uniquement soit des informations cinématiques, soit des informations dynamiques, deux systèmes “articulated_system” séparés sont définis dans une bibliothèque.

```

<library_articulated_systems>
  <articulated_system id="linearunit_kinematics">
    <kinematics>
      <!-- ... ARTICULATED SYSTEM KINEMATICS ... -->
    </kinematics>
  </articulated_system>
  <articulated_system id="linearunit_motion">
    <motion>
      <!-- ... ARTICULATED SYSTEM MOTION ... -->
    </motion>
  </articulated_system>
</library_articulated_systems>

```

Anglais	Français
ARTICULATED SYSTEM KINEMATICS	CINÉMATIQUE DU SYSTÈME ARTICULÉ
ARTICULATED SYSTEM MOTION	MOUVEMENT DU SYSTÈME ARTICULÉ

Figure B.5 – Définition de la bibliothèque du système articulé

La Figure B.5 représente la structure de la définition de la bibliothèque. Les commentaires pour les deux éléments du système articulé indiquent les définitions manquantes. Ce point est expliqué en B.2.6 et B.2.7.

La section “kinematics” d’un système articulé comporte des informations relatives au comportement cinématique d’un modèle articulé, par exemple l’activité des articulations et les limites ou les dépendances entre articulations. La Figure B.6 représente la partie pertinente du document COLLADA et les paramètres cinématiques supplémentaires qui lui sont appliqués dans cet exemple.

```

<articulated_system id="linearunit_kinematics">
  <kinematics>
    <instance_kinematics_model sid="inst_linearunit_kinmodel" url="#linearunit_kinmodel">
      <newparam sid="linearunit_kinematics_inst_linearunit_kinmodel">
        <SIDREF>linearunit_kinematics/inst_linearunit_kinmodel</SIDREF>
      </newparam>
      <newparam sid="linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_axis0">
        <SIDREF>linearunit_kinematics/inst_linearunit_kinmodel/inst_joint1/axis0</SIDREF>
      </newparam>
      <newparam sid="linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_value">
        <float>0.0</float>
      </newparam>
    </instance_kinematics_model>
    <technique_common>
      <axis_info axis="linearunit_kinematics/inst_linearunit_kinmodel/inst_joint1/axis0" sid
="inst_joint1_info">
        <active>
          <bool>true</bool>
        </active>
        <locked>
          <bool>false</bool>
        </locked>
        <limits>
          <min>
            <float>0.0</float>
          </min>
          <max>
            <float>3.0</float>
          </max>
        </limits>
      </axis_info>
      <frame_origin link="linearunit_kinematics/inst_linearunit_kinmodel/baselink" sid="base
frame" />
      <frame_tip link="linearunit_kinematics/inst_linearunit_kinmodel/link1" sid="flangefram
e">
        <translate>1 2 3</translate>
      </frame_tip>
    </technique_common>
  </kinematics>
</articulated_system>

```

Figure B.6 – Définition du système cinématique articulé

Afin d'obtenir des informations dynamiques, un système articulé supplémentaire doit être défini. La section de mouvement d'un système articulé comporte des informations relatives au comportement dynamique d'un modèle articulé, par exemple la vitesse des articulations et l'accélération ou la décélération des articulations. La Figure B.7 représente la partie pertinente du document COLLADA et les paramètres dynamiques supplémentaires qui lui sont appliqués dans cet exemple.

```

<articulated_system id="linearunit_motion">
  <motion>
    <instance_articulated_system sid="inst_linearunit_kinematics" url="#linearunit_kinematic
s">
      <newparam sid="linearunit_motion_linearunit_kinematics_inst_LinearUnit_kinematics_mode
l">
        <SIDREF>linearunit_kinematics/linearunit_kinematics_inst_linearunit_kinmodel</SIDREF
>
      </newparam>
      <newparam sid="linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_j
oint1_axis0">
        <SIDREF>linearunit_kinematics/linearunit_kinematics_inst_linearunit_kinmodel_inst_jo
int1_axis0</SIDREF>
      </newparam>
      <newparam sid="linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_j
oint1_value">
        <SIDREF>linearunit_kinematics/linearunit_kinematics_inst_linearunit_kinmodel_inst_jo
int1_value</SIDREF>
      </newparam>
    </instance_articulated_system>
    <technique_common>
      <axis_info axis="linearunit_kinematics/inst_joint1_info">
        <speed>
          <float>0.0</float>
        </speed>
        <acceleration>
          <float>0.0</float>
        </acceleration>
        <deceleration>
          <float>0.0</float>
        </deceleration>
        <jerk>
          <float>0.0</float>
        </jerk>
      </axis_info>
    </technique_common>
  </motion>
</articulated_system>

```

Figure B.7 – Définition du système de mouvement articulé

B.2.6 Définition de la scène cinématique

Une scène cinématique réunit différents systèmes cinématiques. Tous les paramètres nécessaires pour la scène complète sont publiés dans une scène cinématique. Dans cet exemple, la scène cinématique instancie le système articulé qui est défini en B.2.5. En outre, le modèle cinématique, son axe d'articulation et ses valeurs sont publiés. La Figure B.8 représente la partie pertinente du document COLLADA.

```

<library_kinematics_scenes>
  <kinematics_scene id="linearunit_kinscene">
    <instance_articulated_system sid="inst_linearunit_motion" url="#linearunit_motion">
      <bind symbol="linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_LinearU
nit_kinematics_model">
        <param ref="linearunit_motion/linearunit_motion_linearunit_kinematics_inst_LinearUni
t_kinematics_model" />
      </bind>
      <bind symbol="linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_axis0">
        <param ref="linearunit_motion/linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_axis0" />
      </bind>
      <bind symbol="linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_value">
        <param ref="linearunit_motion/linearunit_motion_linearunit_kinematics_inst_linearu
nit_kinmodel_inst_joint1_value" />
      </bind>
    </instance_articulated_system>
  </kinematics_scene>
</library_kinematics_scenes>

```

Figure B.8 – Définition de la scène cinématique

B.2.7 Assemblage de la scène

Dans l'élément "scene", toutes les scènes cinématiques instanciées sont utilisées. En outre, la liaison avec la scène visuelle est effectuée à ce niveau, comme représenté à la Figure B.9.

Cette liaison définit le flux de la valeur de l'articulation depuis un calcul cinématique jusqu'à la transformation de la géométrie.

```
<scene>
  <instance_visual_scene url="#visualscene" />
  <instance_kinematics_scene sid="inst_linearunit_kinscene" url="#linearunit_kinscene">
    <bind_kinematics_model node="linearunit">
      <param>linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_LinearUnit_kinematics_model</param>
    </bind_kinematics_model>
    <bind_joint_axis target="geomlink1/transform">
      <axis>
        <param>linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_axis0</param>
      </axis>
      <value>
        <param>linearunit_kinscene_linearunit_motion_linearunit_kinematics_inst_linearunit_kinmodel_inst_joint1_value</param>
      </value>
    </bind_joint_axis>
  </instance_kinematics_scene>
</scene>
```

Figure B.9 – Instanciation de la scène cinématique

B.2.8 Combinaison de CAEX et COLLADA en AML

Le document CAEX est constitué d'une InstanceHierarchy nommée "ExampleInstanceHierarchy" avec un InternalElement "LinearUnit" unique. Ce dernier comprend une ExternalInterface nommée "FileLinkExplicit" déduite de l'InterfaceClass "COLLADAInterface", qui fait partie de la bibliothèque InterfaceClassLibrary "AutomationMLInterfaceClassLib" standard. Cette interface est utilisée pour référencer la scène complète de manière explicite, notamment l'unité linéaire, comme représenté à la Figure B.10.

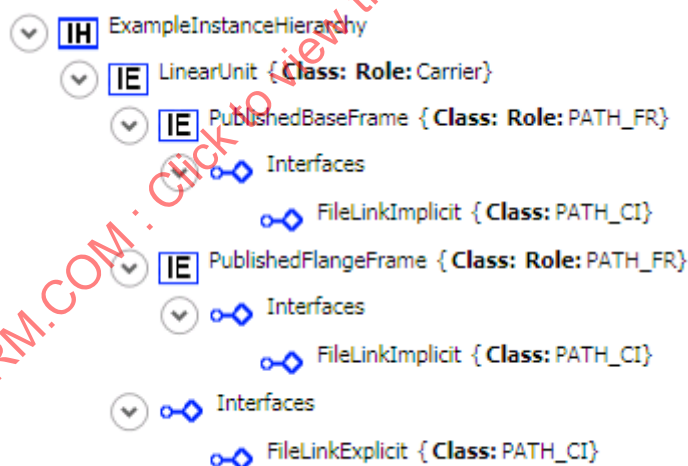


Figure B.10 – Hiérarchie du document AML

L'InternalElement "LinearUnit" a deux InternalElements supplémentaires nommés "PublishedBaseFrame" et "PublishedFlangeFrame" auxquels la RoleClass "Frame" est affectée. Les deux InternalElements référencent les éléments cinématiques de manière implicite dans le document COLLADA. La Figure B.11 représente la partie pertinente du document AML.

```

<InstanceHierarchy Name="ExampleInstanceHierarchy">
  <InternalElement Name="LinearUnit" ID="GUID1">
    <ExternalInterface Name="FileLinkExplicit" RefBaseClassPath="PATH_CI" ID="GUID2">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>explicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./linear_unit.dae</Value>
      </Attribute>
    </ExternalInterface>
  <InternalElement Name="PublishedBaseFrame" ID="GUID3">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID4">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./linear_unit.dae#linearunit_kinematics</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./baseframe</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
  </InternalElement>
  <InternalElement Name="PublishedFlangeFrame" ID="GUID5">
    <ExternalInterface Name="FileLinkImplicit" RefBaseClassPath="PATH_CI" ID="GUID6">
      <Attribute Name="refType" AttributeDataType="xs:string">
        <Value>implicit</Value>
      </Attribute>
      <Attribute Name="refURI" AttributeDataType="xs:anyURI">
        <Value>./linear_unit.dae#linearunit_kinematics</Value>
      </Attribute>
      <Attribute Name="target" AttributeDataType="xs:token">
        <Value>./flangeframe</Value>
      </Attribute>
    </ExternalInterface>
    <RoleRequirements RefBaseRoleClassPath="PATH_FR" />
  </InternalElement>
  <RoleRequirements RefBaseRoleClassPath="PATH_CAR" />
</InternalElement>
</InstanceHierarchy>

```

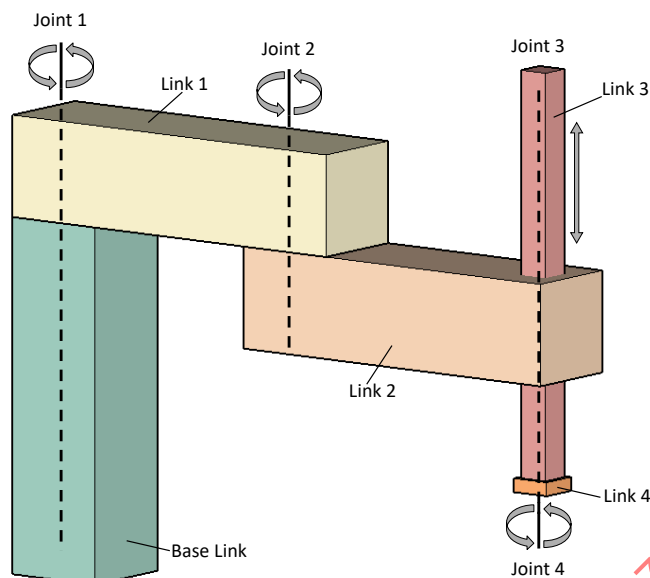
Figure B.11 – Représentation XML du document AML

Chaque cadre publié référence de manière non ambiguë l'élément "articulated_system" dans le document COLLADA par le biais de l'URI relatif "./linear_unit.dae# linearunit_kinematics". Les éléments cibles "frame_tip" et "frame_origin" (voir Figure B.6) sont résolus par exploration depuis le point d'entrée par le biais d'un chemin relatif "./flangeframe" ou "./baseframe", conformément aux règles d'un SID.

B.3 Modélisation d'un document AML d'un robot en CAEX et COLLADA

B.3.1 Généralités

L'Article B.3 décrit la modélisation d'un robot SCARA (Bras robotisé d'assemblage à conformité sélective), qui est un type spécialisé de robot industriel. La scène visuelle est décrite en B.3.2. Dans les paragraphes qui suivent, la scène est complétée par étapes avec des éléments cinématiques. Pour finir, le document AML est défini et combiné avec le système cinématique.



Anglais	Français
Joint	Articulation
Link	Liaison
Base link	Liaison de base

Figure B.12 – Visualisation du robot

B.3.2 Définition de la scène visuelle

Dans cet exemple, la scène visuelle du robot SCARA est constituée de cinq éléments géométriques, comme représenté à la Figure B.12. Chaque élément géométrique représente une liaison du robot. Afin de maintenir le contenu COLLADA à un niveau minimal, chaque liaison est représentée par un parallélépipède rectangle, qui est défini dans le document COLLADA "cube.dae", comme représenté à la Figure A.7. Pour obtenir la forme nécessaire, chaque cube a une échelle différente. Le premier cube représente la liaison de base du robot SCARA tandis que les cubes restants représentent les liaisons subséquentes.

La Figure B.13 représente le contenu du document COLLADA "robot.dae" qui définit la représentation géométrique du robot.

```
<COLLADA version="1.5.0">
  <asset>
    <contributor>
      <author>WriterVendor</author>
      <author_website>http://www.WriterVendor.com</author_website>
      <authoring_tool>"WriterName" "WriterVersion" "WriterRelease"</authoring_tool>
      <comments>This document defines the geometry and kinematic of a SCARA robot</comments>
    </contributor>
    <created>2014-04-14T11:37:18.1875000</created>
    <modified>2014-04-14T11:37:18.1875000</modified>
    <unit meter="1.0" name="meter" />
    <up_axis>Z_UP</up_axis>
  </asset>
  <library_effects>
    <effect id="base-fx">
      <profile_COMMON>
        <technique sid="COMMON">
          <phong>
            <diffuse>
              <color>0.60 0.77 0.72 1</color>
            </diffuse>
          </phong>
        </technique>
      </profile_COMMON>
    </effect>
    <effect id="link1-fx">
      <profile_COMMON>
```

```

<technique sid="COMMON">
  <phong>
    <diffuse>
      <color>0.94 0.91 0.77 1</color>
    </diffuse>
  </phong>
</technique>
</profile_COMMON>
</effect>
<effect id="link2-fx">
  <profile_COMMON>
    <technique sid="COMMON">
      <phong>
        <diffuse>
          <color>0.93 0.80 0.69 1</color>
        </diffuse>
      </phong>
    </technique>
  </profile_COMMON>
</effect>
<effect id="link3-fx">
  <profile_COMMON>
    <technique sid="COMMON">
      <phong>
        <diffuse>
          <color>0.85 0.59 0.57 1</color>
        </diffuse>
      </phong>
    </technique>
  </profile_COMMON>
</effect>
<effect id="link4-fx">
  <profile_COMMON>
    <technique sid="COMMON">
      <phong>
        <diffuse>
          <color>0.95 0.64 0.41 1</color>
        </diffuse>
      </phong>
    </technique>
  </profile_COMMON>
</effect>
</library_effects>
<library_materials>
  <material id="base-mat" name="BaseMat">
    <instance_effect url="#base-fx" />
  </material>
  <material id="link1-mat" name="Link1Mat">
    <instance_effect url="#link1-fx" />
  </material>
  <material id="link2-mat" name="Link2Mat">
    <instance_effect url="#link2-fx" />
  </material>
  <material id="link3-mat" name="Link3Mat">
    <instance_effect url="#link3-fx" />
  </material>
  <material id="link4-mat" name="Link4Mat">
    <instance_effect url="#link4-fx" />
  </material>
</library_materials>
<library_visual_scenes>
  <visual_scene id="visualscene">
    <node id="robot" name="Robot">
      <scale>2 2 2</scale>
      <node id="geombase" name="GeomBase">
        <translate>-0.05 -0.05 0</translate>
        <scale>0.1 0.1 0.35</scale>
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#base-mat" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      <node id="geomlink1" name="GeomLink1">
        <translate>0 0 1</translate>
        <scale>3.8 1 0.2857</scale>
        <instance_geometry url="./cube.dae#cube-geom">
          <bind_material>
            <technique_common>
              <instance_material symbol="mat" target="#link1-mat" />
            </technique_common>
          </bind_material>
        </instance_geometry>
      </node>
    </node>
  </visual_scene>
</library_visual_scenes>

```

```

</instance_geometry>
<node id="geomlink2" name="GeomLink2">
  <translate>0.736848 0 -1</translate>
  <scale>0.947376 1 1</scale>
  <instance_geometry url="./cube.dae#cube-geom">
    <bind_material>
      <technique_common>
        <instance_material symbol="mat" target="#link2-mat" />
      </technique_common>
    </bind_material>
  </instance_geometry>
<node id="geomlink3" name="GeomLink3">
  <translate>0.861 0.35 -1</translate>
  <scale>0.08333 0.3 4</scale>
  <instance_geometry url="./cube.dae#cube-geom">
    <bind_material>
      <technique_common>
        <instance_material symbol="mat" target="#link3-mat" />
      </technique_common>
    </bind_material>
  </instance_geometry>
<node id="geomlink4" name="GeomLink4">
  <translate>-0.1666 -0.166 -0.0333</translate>
  <scale>1.333 1.333 0.0333</scale>
  <instance_geometry url="./cube.dae#cube-geom">
    <bind_material>
      <technique_common>
        <instance_material symbol="mat" target="#link4-mat" />
      </technique_common>
    </bind_material>
  </instance_geometry>
</node>
</node>
</node>
</node>
</node>
</visual_scene>
</library_visual_scenes>
<!-- Step 1: Definition of joint elements -->
<!-- Step 2: Definition of kinematic model element -->
<!-- Step 3: Definition of articulated system (kinematics) element -->
<!-- Step 4: Definition of kinematics scene element -->
<scene>
  <instance_visual_scene url="#visualscene" />
  <!-- Step 5: Instantiation of kinematics scene -->
</scene>
</COLLADA>

```

Anglais	Français
This document defines the geometry and kinematic of a SCARA robot	Le présent document définit la géométrie et la cinématique d'un robot SCARA
Step 1: Definition of joint elements	Étape 1: Définition des éléments d'articulation
Step 2: Definition of kinematic model element	Étape 2: Définition de l'élément du modèle cinématique
Step 3: Definition of articulated system (kinematics) element	Étape 3: Définition de l'élément (cinématique) du système articulé
Step 4: Definition of kinematics scene element	Étape 4: Définition de l'élément de la scène cinématique
Step 5: Instantiation of kinematics scene	Étape 5: Instanciation de la scène cinématique

Figure B.13 – Définition de la scène visuelle

Chaque géométrie est affectée à un nœud (geombase, geomlink1, geomlink2, geomlink3, geomlink4). Les paragraphes suivants définissent les éléments cinématiques exigés et les appliquent à cette scène pour compléter le système cinématique.

B.3.3 Définition des articulations

Le robot utilisé dans cet exemple est constitué de quatre articulations (rotationnelle, rotationnelle, prismatique, rotationnelle). Toutes les articulations sont définies dans une bibliothèque et instanciées à une étape ultérieure. Les articulations prismatiques sont définies par l'élément "prismatic" tandis que les articulations rotationnelles sont définies par l'élément

“revolute”. Tandis que, dans cet exemple, la direction de l’axe d’articulation est spécifiée, les limites inférieure et supérieure de l’articulation sont définies de manière exemplaire à un emplacement différent. L’extrait du document COLLADA est représenté à la Figure B.14.

```
<library_joints>
  <joint id="joint1" name="joint1">
    <revolute sid="axis0">
      <axis>0 0 1</axis>
    </revolute>
  </joint>
  <joint id="joint2" name="joint2">
    <revolute sid="axis0">
      <axis>0 0 1</axis>
    </revolute>
  </joint>
  <joint id="joint3" name="joint3">
    <prismatic sid="axis0">
      <axis>0 0 1</axis>
    </prismatic>
  </joint>
  <joint id="joint4" name="joint4">
    <revolute sid="axis0">
      <axis>0 0 1</axis>
    </revolute>
  </joint>
</library_joints>
```

Figure B.14 – Définition d’articulations

B.3.4 Définition du modèle cinématique

Le Paragraphe B.3.4 définit le modèle cinématique et sa hiérarchie. Tout d’abord, toutes les articulations du modèle cinématique sont instanciées à partir d’une bibliothèque d’articulations. Les chaînes cinématiques suivantes sont constituées de liaisons et d’articulations ainsi que de leur relation hiérarchique. Toute liaison ou articulation peut donner lieu à une transformation. La transformation d’une liaison ou d’une articulation dépend toujours de son prédécesseur et est décrite par les éléments “translate” et “rotate”. Le modèle cinématique est défini dans une bibliothèque et est instancié à une étape ultérieure. La Figure B.15 représente la partie pertinente du document COLLADA.

```
<library_kinematics_models>
  <kinematics_model id="robot_kinmodel">
    <technique_common>
      <instance_joint sid="inst_joint1" url="#joint1" />
      <instance_joint sid="inst_joint2" url="#joint2" />
      <instance_joint sid="inst_joint3" url="#joint3" />
      <instance_joint sid="inst_joint4" url="#joint4" />
      <link name="BaseLink" sid="baselink">
        <attachment_full joint="robot_kinmodel/inst_joint1">
          <translate>0.0 0.0 0.3</translate>
          <link name="Link1" sid="link1">
            <attachment_full joint="robot_kinmodel/inst_joint2">
              <translate>0.28 0.0 0.0</translate>
              <link name="Link2" sid="link2">
                <attachment_full joint="robot_kinmodel/inst_joint3">
                  <translate>0.26 0.0 0.0</translate>
                  <link name="Link3" sid="link3">
                    <attachment_full joint="robot_kinmodel/inst_joint4">
                      <translate>0.0 0.0 -0.15</translate>
                      <link name="Link4" sid="link4" />
                    </attachment_full>
                  </link>
                </attachment_full>
              </link>
            </attachment_full>
          </link>
        </attachment_full>
      </link>
    </technique_common>
  </kinematics_model>
</library_kinematics_models>
```

Figure B.15 – Définition de modèle cinématique