INTERNATIONAL STANDARD

ISO 13584-25

First edition
2004-06-15

Industrial automation systems and integration — Parts library —

Part 25: Logical resource: Logical model of supplier library with aggregate values and explicit content

Systèmes d'automatisation industrielle et intégration — Bibliothèque de composants —

Partie 25: Ressource logique: Modèle logique de fournisseur avec des valeurs d'ensemble et un contenu explicite

Reference number
ISO 13584-25:2004(E)

© ISO 2004

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

© ISO 2004

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 ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Contents

--`,``,,-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Figures

Tables

--`,`,,`,`,,`,,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Foreword

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

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

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

ISO 13584-25 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC 4, *Industrial data*.

ISO 13584 consists of the following parts, under the general title *Industrial automation systems and integration — Parts library*:

— *Part 1: Overview and fundamental principles*

— *Part 20: Logical resource: Logical model of expressions*

— *Part 24: Logical resource: Logical model of supplier library*

— *Part 25: Logical resource: Logical model of supplier library with aggregate values and explicit content*

— *Part 26: Logical resource: Information supplier identification*

— *Part 31: Implementation resources: Geometric programming interface*

— *Part 42: Description methodology: Methodology for structuring part families*

— *Part 101: Geometrical view exchange protocol by parametric program*

— *Part 102: View exchange protocol by ISO 10303 conforming specification*

The structure of ISO 13584 is described in ISO 13584-1. The numbering of the parts of ISO 13584 reflects its structure:

— Parts 10 to 19 specify the conceptual descriptions;

— Parts 20 to 29 specify the logical resources;

— Parts 30 to 39 specify the implementation resources;

© ISO 2004 – All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

— Parts 40 to 49 specify the description methodology;

— Parts 100 to 199 specify the view exchange protocol.

Should further parts of ISO 13584 be published, they will follow the same numbering pattern.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Introduction

ISO 13584 is an International Standard for the computer-interpretable representation and exchange of parts library data. The objective is to provide a neutral mechanism capable of transferring parts library data, independent of any application that is using a parts library data system. The nature of this description makes it suitable not only for the exchange of files containing parts, but also as a basis for implementing and sharing databases of parts library data.

ISO 13584 is organized as a series of parts, each published separately. The parts of ISO 13854 fall into one of the following series: conceptual descriptions, logical resources, implementation resources, description methodology and view exchange protocol. The series are described in ISO 13584-1. This part of ISO 13584 is a member of the logical resources series.

This part of ISO 13584 specifies the generic resources needed for modelling supplier libraries that contain properties whose values may be aggregate-structured, and whose possible content is explicitly described as a set of instances. It also provides the EXPRESS integrated information models that permit the exchange of such supplier libraries. Knowledge of EXPRESS as defined in ISO 10303-11 is required to understand this part of ISO 13584. Basic knowledge of ISO 13584-24 and ISO 13584-42 is also required.

The generic resources specified in this document were developed as a joint effort of ISO TC184/SC4/WG2 and IEC SC3D. They are intended to be documented both in this part of ISO 13584 and in IEC 61360-5. Both committees agreed not to change and/or modify the presented EXPRESS schemas independent of each other in order to guarantee the harmonization and the reusability of the work from both committees. Requests for amendments should therefore be sent to both committees. These requests should be adopted by both committees before modifying the EXPRESS schemas.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

--`,``,,-`-`,,`,,`,`,,`---

Not for Resale

Industrial automation systems and integration — Parts library —

Part 25:
Logical resource: Logical model of supplier library with aggregate values and explicit content

1 Scope

This part of ISO 13584 provides generic EXPRESS resource constructs that support the description of aggregate data types and values occurring in supplier libraries. It also contains an integrated EXPRESS information model for representing supplier libraries for the purpose of exchange. This integrated information model integrates the above resource constructs with other EXPRESS resource constructs from different parts of ISO 13584 and ISO 10303 into one single schema. Supplier libraries may consist of definitions and of representations of families of parts. They may also define new representation categories. Supplier libraries may consist only of dictionary elements with or without aggregate data types, or they may also contain explicit specifications of the sets of permitted instances.

When used together with view exchange protocols, this integrated information model also permits the exchange of one or several representation categories for the parts defined in a parts library.

The following are within the scope of this part of ISO 13584:

— generic resource constructs for representing aggregate data types. Aggregate data types and values are modeled according to the definition of aggregate data types of the EXPRESS language (ISO 10303-11);

— generic resource constructs for representing aggregate values;

— generic resource constructs for representing of assembled parts that may contain an unlimited number of constituent components;

— a library integrated information model that provides for modeling and exchanging supplier libraries that contain properties whose values may be aggregate-structured, and whose possible class extensions are explicitly described as sets of instances.

The following are outside the scope of this part of ISO 13584:

— representation of expressions and variables;

— implicit description of the set of permitted instances of a class by means of constraints;

— specification of a software system able to manage supplier libraries represented according to the information models defined in this part of ISO 13584.

2 Normative references

The following referenced documents are indispensable for the application 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.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

IEC 61360-2:1998, *Standard data element types with associated classification scheme for electric components — Part 2: EXPRESS dictionary schema*

ISO/IEC 8824-1, *Information technology — Abstract Syntax Notation One (ASN.1): Specification of basic notation*

ISO 8859-1, *Information technology — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1*

ISO 10303-11:1994, *Industrial automation systems and integration — Product data representation and exchange — Part 11: Description methods: The EXPRESS language reference manual*

ISO 10303-21:2002, *Industrial automation systems and integration — Product data representation and exchange — Part 21: Implementation methods: Clear text encoding of the exchange structure*

ISO 10303-41:2000, *Industrial automation systems and integration — Product data representation and exchange — Part 41: Integrated generic resource: Fundamentals of product description and support*

ISO 10303-42: 2003, *Industrial automation systems and integration — Product data representation and exchange — Part 42: Integrated generic resource: Geometric and topological representation*

ISO 10303-43: 2000, *Industrial automation systems: and integration — Product data representation and exchange — Part 43: Integrated generic resource: Representation structures*

ISO 13584-24:2003, *Industrial automation systems and integration — Parts library — Part 24: Logical resource: Logical model of supplier library*

ISO 13584-42:1998, *Industrial automation systems and integration — Parts library — Part 42: Description methodology: Methodology for structuring part families*

IAB RFC 2068, *Hypertext transfer protocol HTTP/1.1 (HTTP-1.1)*, Internet architecture board proposed standard protocol

3 Terms, definitions and abbreviations

For the purposes of this document, the following terms and definitions apply. Some of these terms and definitions are repeated for convenience from:

— ISO 10303-1:1994;

— ISO 10303-11:1994;

— ISO 13584-1:2001;

— ISO 13584-24:2003;

— ISO 13584-42:1998.

NOTE Definitions copied verbatim from other standards are followed by a reference to the source standard in brackets. Definitions that have been adapted from other standards are followed by an explanatory note.

--`,``,,-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

3.1

applicable property
a property that is defined for some family of parts and that shall apply to any part that belongs to this family of parts
[ISO 13584-24:2003]

EXAMPLE For a screw generic family of parts, the threaded diameter is an applicable property: this characteristic applies to any screw.

3.2

basic semantic unit
BSU
the entity that provides an absolute and universal identification of certain objects of the application domain (e.g. classes, data element types)
[ISO 13584-42:1998]

3.3

class extension
set of all the different possible instances conforming to the specification defined by a class
[ISO 13584-24:2003]

3.4

common dictionary schema
the information model for a dictionary, using the EXPRESS modeling language, resulting from a joint effort between ISO TC184/SC4/WG2 and IEC SC3D
[ISO 13584-42:1998]

NOTE The common dictionary schema is formally named ISO13584_IEC61360_dictionary_schema and is specified in IEC 61360-2. This schema is duplicated in Annex D of ISO 13584-42:1998.

3.5

conformance class
a subset of a standard for which conformance may be claimed
[ISO 13584-24:2003]

3.6

conformance requirement
a precise, text definition of a characteristic required to be present in a conforming implementation
[ISO 10303-1:1994]

3.7

dictionary element
the set of attributes that constitutes the dictionary description of certain objects of the application domain (e.g. classes, data element types)
[ISO 13584-42:1998]

3.8

data element type
DET
unit of data for which the identification, the description and value representation have been specified
[ISO 13584-42:1998]

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

3.9

data type
a domain of values
[ISO 10303-11:1994]

3.10

family of parts
a simple or generic family of parts
[ISO 13584-24:2003]

3.11

functional model of a part
the library data that represent one representation category of a part in an integrated library
[ISO 13584-1:2001]

3.12

functional view of a part
the data that represent one representation category of a part in product data
[ISO 13584-1:2001]

NOTE The structure of a functional view does not depend on the part it represents.

3.13

general model of a part
the library data that carries the definition and identity of a part in an integrated library
[ISO 13584-1:2001]

3.14

generic family of parts
a grouping of simple or generic families of parts done for purposes of classification or for factoring common information
[ISO 13584-24:2003]

3.15

library delivery file
a population of EXPRESS entity instances conforming to a library integrated information model and represented according to one of the implementation methods specified in ISO 10303
[ISO 13584-24:2003]

NOTE A library delivery file specifies the structure and the content of a supplier library. It may reference library external files.

3.16

library part
a part associated with a set of data that represents it in a library
[ISO 13584-1:2001]

--`,``,,-`-`,,`,,`,`,,`---
Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

3.17

library part data
the data that represent a part in a library
[ISO 13584-1:2001]

3.18

library exchange context
the set of one library delivery file and zero, one or several library external files that represent together a supplier library
[ISO 13584-24:2003]

3.19

library external file
a file, referenced from a library delivery file, that contributes to the definition of a supplier library
[ISO 13584-24:2003]

NOTE The structure and the format of a library external file is specified in the library delivery file that references it.

3.20

library integrated information model
LIIM
an EXPRESS schema that integrates resource constructs from different EXPRESS schemas for representing supplier libraries for the purpose of exchange and that is associated with conformance requirements
[ISO 13584-24:2003]

3.21

library specification of a class
explicit representation of a class extension in a supplier library
[ISO 13584-24:2003]

NOTE 1 In the ISO 13584 series, every class is intentionally defined through a dictionary element. Only those classes of which the supplier desires to represent explicitly the possible instances are associated with a library specification.

NOTE 2 In this part of ISO 13584, the library specification of a class consists of a set that contains all the different possible instances.

3.22

part
material or functional element that is intended to constitute a component of different products
[ISO 13584-1:2001]

3.23

property
an information that may be represented by a data element type
[ISO 13584-42:1998]

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

3.24

representation category
an abstraction used to distinguish between different possible user requirements regarding a part representation
[ISO 13584-1:2001]

NOTE In the model defined in the ISO 13584 standard series, this distinction is formally expressed in terms of a view logical name and in terms of the view control variables.

3.25

resource construct
the collection of EXPRESS language entities, types, functions, rules and references that together define a valid description of data
[ISO 13584-24:2003]

3.26

simple family of parts
a set of parts of which each part may be described by the same group of properties
[ISO 13584-24:2003]

3.27

supplier library
a set of data, and possibly of programs, for which the supplier is defined and that describes in the standard format defined in ISO 13584 a set of parts and/or a set of representation of parts
[ISO 13584-1:2001]

3.28

user library
the information that results from the integration of one or more supplier libraries by the library management system and possibly from a later adaptation performed by the user
[ISO 13584-1:2001]

3.29

view exchange protocol
VEP
a part of ISO 13584 that describes the use of resource constructs and of representation transmission interfaces that satisfy the information requirement for the exchange of one representation category of parts
[ISO 13584-24:2003]

3.30

visible property
a property that is defined for some family of parts and that may or not apply to the different parts of this family of parts
[ISO 13584-24:2003]

EXAMPLE For a generic family of screws, the non-threaded length is a visible property: it is clearly defined for any screw, but only those screws with a non-threaded part have a value for this property.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

NOTE The code of the class where a property is defined as visible is part of the identification of the data element type that represents this property.

4 Structure of ISO 13584-25

ISO 13584-25 has two main parts.

The generic resources part provides resource constructs for representing aggregate data types and values. Aggregate data types and values include those defined in the EXPRESS language.

Library integrated information model LIIM 25 gathers the above resource constructs with other generic resource constructs from other parts of ISO 13584 and from ISO 10303 into a single schema for representing supplier libraries. That schema includes aggregate data types and aggregate values and explicitly describes class extensions as sets of instances.

4.1 Generic resources

The generic resources consist of the following EXPRESS schemas:

— **ISO13584_IEC61360_dictionary_aggregate_extension_schema**;

— **ISO13584_aggregate_value_schema**.

These schemas provide resource constructs that are generic in nature. They may be used outside ISO 13584, and particularly in all the applications that use a data dictionary compliant with the IEC 61360 standard series.

4.1.1 ISO13584_IEC61360_dictionary_aggregate_extension_schema

The **ISO13584_IEC61360_dictionary_aggregate_extension_schema** provides the resource constructs needed to describe data types corresponding to aggregate data types that include those defined in the EXPRESS language. It defines resources to describe array, bag, list, set and set of subsets data types. These data types extend the data types already defined in the **ISO13584_IEC61360_dictionary_schema** published in IEC 61360-2:1998 and whose content is duplicated in an informative Annex of ISO 13584-42:1998.

4.1.2 ISO13584_aggregate_value_schema

The **ISO13584_aggregate_value_schema** provides the resource constructs needed to describe values of data types corresponding to aggregate data types as defined in the EXPRESS language. It defines resources to describe array, bag, list and set–structured values. These data values extend the data values already defined in the **ISO13584_instance_resource_schema** specified in ISO 13584-24.

4.2 Library integrated model

The library integrated information model specified in this part of ISO 13584, called LIIM 25, gathers the generic resource constructs defined in this part of ISO 13584 with other generic resource constructs from other parts of ISO 13584 and from ISO 10303 into a single schema for representing supplier libraries for the purpose of exchange. LIIM 25 enables the exchange of seven kinds of dictionaries or libraries and the exchange of a set of library instances without library structure, between a library data supplier and a library end-user.

— Dictionaries that define hierarchies of classes of items, that may be parts, materials or other items, with aggregate-structured properties using only the EXPRESS resource constructs defined in the ISO/IEC common dictionary schema or in the **ISO13584_IEC61360_dictionary_aggregate _extension_schema** defined in this part of ISO 13584 corresponds to conformance class 1;

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

— Dictionaries that define hierarchies of classes of items, that may be parts, materials, features or other items, using the extension of the ISO/IEC common dictionary schema defined in ISO 13584-24, but without description of item representations and of representation categories of items, and without aggregate-structured properties, corresponds to conformance class 2;

— Dictionaries that define hierarchies of classes of items, of item representations, and of representation categories of items, with aggregate-structured properties, corresponds to conformance class 3;

— Dictionaries with the same scope as conformance class 3 but with no more than two levels nesting for aggregate-structured properties, corresponds to conformance class 4;

— Libraries that define the set of instances that belong to some hierarchies of classes of items, without classes of item representations and of representation categories of items, and without aggregate-structured properties, corresponds to conformance class 5;

— Libraries that define the set of instances that belong to some hierarchies of classes of items and of item representations, with aggregate-structured properties corresponds to conformance class 6;

— Libraries with the same scope as conformance class 6 but with no more than two levels nesting for aggregate-structured properties corresponds to conformance class 7;

— Set of item instances and of item representation instances without dictionary definitions and without library structure corresponds to conformance class 10;

— Set of item instances and of item representation instances without library structure but possibly with dictionary definitions corresponds to conformance class 11.

Each of the above kind of exchange context corresponds to one conformance class of LIIM 25. Each conformance class specifies the conformance requirements for implementations that claim conformance to this conformance class. Conformance classes 6 and 7 supports all the entities, types and associated constructs that are part of LIIM 25. Other conformance classes shall only support a subset of this set of resource constructs. In this part of ISO 13584, each subset that defines a conformance class is defined by means of a list of entities. An implementation that claims conformance to any conformance class shall support all the entities listed for this conformance class and related constructs.

4.2.1 Conformance class 1: minimal dictionaries

Conformance class 1 supports the information requirements for exchanging definitions of hierarchies of item classes, where items may be parts or materials. It allows the exchange of all the dictionary elements from the ISO/IEC dictionary schema and item classes whose properties may have aggregate-structured values. Conformance class 1 is associated with implementation methods for the library delivery file. Conformance requirements to conformance class 1 of LIIM 25 are defined in Clause 8.2.1 of this part of ISO 13584.

4.2.2 Conformance class 2: dictionaries of items classes

Conformance class 2 supports the information requirements for exchanging definitions of hierarchies of item classes, where items may be parts, materials or features, whose properties may not have aggregate-structured values. Conformance class 2 is associated with a set of standard data that defines the formats of library external files that may be referenced by a library delivery file conforming to conformance class 2 of LIIM 25, and with implementation methods for the library delivery file. Conformance requirements for conformance class 2 of LIIM 25 are defined in Clause 8.2.2 of this part of ISO 13584.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

4.2.3 Conformance class 3: complete dictionaries

Conformance class 3 supports the information requirements for exchanging definitions of hierarchies of item classes, where items may be parts, materials or features , together with definitions of representations of such item classes, and with definitions of representation categories of such item classes. Properties of all these classes may have aggregate-structured values. Conformance class 3 is associated with a set of standard data that defines the formats of library external files that may be referenced by a library delivery file conforming to conformance class 3 of LIIM 25, and with implementation methods for the library delivery file. Conformance requirements for conformance class 3 of LIIM 25 are defined in Clause 8.2.3 of this part of ISO 13584.

4.2.4 Conformance class 4: complete dictionaries with limited nested aggregate values

Conformance class 4 supports the information requirements corresponding to conformance class 3 with a restriction. The aggregate values involved in conformance class 4 shall not be nested more than twice. Conformance requirements for conformance class 4 of LIIM 25 are defined in Clause 8.2.4 of this part of ISO 13584.

4.2.5 Conformance class 5: libraries of item classes

Conformance class 5 supports the information requirements for exchanging definitions and instances of hierarchies of item classes, where items may be parts, materials or features, whose properties may not have aggregate-structured values. Class extensions may only be defined by sets of instances, without any constraint or expression. Conformance class 5 is associated with a set of standard data that defines the formats of library external files that may be referenced by a library delivery file conforming to conformance class 5 of LIIM 25, and with implementation methods for the library delivery file. Conformance requirements for conformance class 5 of LIIM 25 are defined in Clause 8.2.5 of this part of ISO 13584.

4.2.6 Conformance class 6: complete libraries

Conformance class 6 supports the information requirements for exchanging definitions and instances of hierarchies of item classes, where items may be parts, materials or features together with definitions and instances of representations of such item classes, and with definitions of representation categories of such item classes. Properties of all these classes may have aggregate-structured values. Class extensions may only be defined by sets of instances, without any constraint or expression. Conformance class 6 is associated with a set of standard data that defines the formats of library external files that may be referenced by a library delivery file conforming to conformance class 6 of LIIM 25, and with implementation methods for the library delivery file. Conformance requirements for conformance class 6 of LIIM 25 are defined in Clause 8.2.6 of this part of ISO 13584.

4.2.7 Conformance class 7: complete libraries with limited nested aggregate values

Conformance class 7 supports the information requirements corresponding to conformance class 6 with a restriction. The aggregate values involved in this conformance class shall not be nested more than twice. Conformance requirements for conformance class 7 of LIIM 25 are defined in Clause 8.2.7 of this part of ISO 13584.

4.2.8 Conformance class 10: library instances

Conformance class 10 supports the information requirements for exchanging one or several instances of items, where items may be parts, materials or features, or of item representation but without dictionary definitions. This conformance class is intended to provide resources for delivering a set of instance selected by a user in a library, possibly with item representations. Conformance class 10 is associated with a set of standard data that defines the formats of library external files that may be referenced by a library delivery file conforming to conformance class 10 of LIIM 25, and with implementation methods for the library delivery file. Conformance requirements for conformance class 10 of LIIM 25 are defined in

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Clause 8.2.8 of this part of ISO 13584.

4.2.9 Conformance class 11: library instances with asociated dictionary definitions

Conformance class 11 supports the information requirements for exchanging one or several instances of items, where items may be parts, materials or features, together with one or several instances of representations of such item without library structure but possibly with dictionary definitions. This conformance class is intended to provide resources for delivering a set of instance selected by a user in a library, possibly with instance representations, when the user does not possess the dictionaries used in the library. It may also be used for delivering updates of libraries. Conformance class 11 is associated with a set of standard data that defines the formats of library external files that may be referenced by a library delivery file conforming to conformance class 11 of LIIM 25, and with implementation methods for the library delivery file. Conformance requirements for conformance class 11 of LIIM 25 are defined in Clause 8.2.9 of this part of ISO 13584.

5 Fundamental concepts and assumptions

The following concepts and assumptions apply to this part of ISO 13584.

5.1 Aggregate-structured value of properties

The **ISO13584_IEC61360_dictionary_aggregate_extension_schema** and **ISO13584_aggregate_value_schema** schemas define the generic resource constructs for representing aggregate data types and generic resource constructs for representing aggregate values. Aggregate data types and values include those defined in the EXPRESS language (ISO 10303-11:1994). These resources allow to assign aggregate type and values to any property of any instance of any class described in a library.

5.2 Explicit description of a class extension

The library integrated information model defined in this part of ISO 13584 provides for modeling and exchanging supplier libraries that contain properties whose values may be aggregate-structured, and whose possible class extensions are explicitly described as sets of instances. This library integrated information model is provided for the explicit representation of libraries as a set of instances associated with properties values.

6 ISO13584_IEC61360_dictionary_aggregate_extension_schema

This clause defines the requirements for the **ISO13584_IEC61360_dictionary_aggregate_extension_schema**. The following EXPRESS declaration introduces the **ISO13584_IEC61360_dictionary_aggregate_extension_schema** and identifies the necessary external references.

EXPRESS specification:

```
*)
SCHEMA ISO13584_IEC61360_dictionary_aggregate_extension_schema;

REFERENCE FROM ISO13584_IEC61360_dictionary_schema(
      data_type,
      entity_instance_type);

(*
```

NOTE The schema referenced above can be found in the following document:

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

ISO13584_IEC61360_dictionary_schema IEC 61360-2:1998
(which is duplicated for convenience in Informative Annex D of ISO 13584-42:1998)

6.1 Introduction to the ISO13584_IEC61360_dictionary_aggregate_extension_schema

The **ISO13584_IEC61360_dictionary_aggregate_extension_schema** provides the information model for the extension to the ISO/IEC common dictionary schema which allows the use of lists, sets, bags, arrays and sets of subsets of simple or complex data types.

This extension is achieved in two steps.

— the **entity_instance_type_for_aggregate** entity provides the means to reference EXPRESS-defined entities that specify aggregate data types. The **entity_instance_type_for_aggregate** is a subtype of the **entity_instance_type** entity;

NOTE The **entity_instance_type** entity is defined in the IEC 61630–2:1998 and duplicated in ISO 13584-42:1988.

— then, entities that specify aggregate data types are modeled by the **aggregate_type** entity and its specializations.

6.2 ISO13584_IEC61360_dictionary_aggregate_extension_schema entity definitions

The following entity type definitions describe the necessary resources needed to encode aggregate types.

6.2.1 Aggregate_entity_instance_type

The **entity_instance_type_for_aggregate** entity provides for referencing definitions of data types that may be expressed as lists, sets, bags or arrays of simple or complex values. It is defined by referencing an **aggregate_type** defined in this schema.

EXPRESS specification:

```
*)
ENTITY entity_instance_type_for_aggregate
SUBTYPE OF(entity_instance_type);
      type_structure: aggregate_type;
WHERE
      WR1: SELF\entity_instance_type.type_name =
           ['ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA'
           + '.AGGREGATE_TYPE'];
END_ENTITY;
(*
```

Attribute definition:

type_structure: the **aggregate_type** referenced and carried by the **entity_instance_type**.

Formal propositions:

WR1: the **type_name** attribute of the **entity_instance_type** shall contain the string ‘ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.AGGREGATE_TYPE’.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.2.2 Aggregate_type

The **aggregate_type** entity provides for the definition of data types that may be expressed as lists, sets, bags or arrays of simple or complex values.

EXPRESS specification:

```
*)
ENTITY aggregate_type
ABSTRACT SUPERTYPE OF(ONEOF(
          list_type,
          set_type,
          bag_type,
          array_type,   set_with_subset_constraint_type ));
     bound_1: OPTIONAL INTEGER;
     bound_2: OPTIONAL INTEGER;
     value_type: data_type;
WHERE
     WR1: bound_1 <= bound_2;
END_ENTITY;
(*
```

Attribute definition:

value_type: is the type of value (simple or complex) which is used for each element of the aggregate.

bound_1: the optional integer that defines the low bound of the defined aggregate type.

bound_2: the optional integer that defines the upper bound of the defined aggregate type.

Formal propositions:

WR1: **bound_1** cannot be greater than **bound_2**.

6.2.3 List_type

The **list_type** entity provides for the definition of data types that may be expressed as ordered lists of values in which duplication may or may not be allowed.

EXPRESS specification:

```
*)
ENTITY list_type
SUBTYPE OF(aggregate_type);
     uniqueness: BOOLEAN;
WHERE
     WR1: EXISTS(bound_1) OR NOT(EXISTS(bound_2));
     WR2: NOT(EXISTS(bound_1)) OR (bound_1 >= 0);
END_ENTITY;
(*
```

Attribute definition:

uniqueness: a flag to indicate whether all elements of the list must be unique (true) or whether duplicates are allowed (false).

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Formal propositions:

WR1: if the upper bound **bound_2** of the defined list optional attribute exists, it implies that the lower bound **bound_1** optional attribute of the defined list exists as well.

WR2: if the lower bound **bound_1** of the defined list optional attribute exists then it is greater or equal to 0.

6.2.4 Set_type

The **set_type** entity provides for the definition of data types that may be expressed as unordered collections of values in which no duplication can occur.

EXPRESS specification:

```
*)
ENTITY set_type
SUBTYPE OF(aggregate_type);
WHERE
     WR1: EXISTS(bound_1) OR NOT(EXISTS(bound_2));
     WR2: NOT(EXISTS(bound_1)) OR (bound_1 >= 0);
END_ENTITY;
(*
```

Formal propositions:

WR1: if the upper bound **bound_2** of the defined list optional attribute exists, it implies that the lower bound **bound_1** optional attribute of the defined list exists as well.

WR2: if the lower bound **bound_1** of the defined list optional attribute exists then it is greater or equal to 0.

6.2.5 Bag_type

The **bag_type** entity provides for the definition of data types that may be expressed as unordered collections of values in which duplication may occur.

EXPRESS specification:

```
*)
ENTITY bag_type
SUBTYPE OF(aggregate_type);
WHERE
     WR1: EXISTS(bound_1) OR NOT(EXISTS(bound_2));
     WR2: NOT(EXISTS(bound_1)) OR (bound_1 >= 0);
END_ENTITY;
(*
```

Formal propositions:

WR1: if the upper bound **bound_2** of the defined list optional attribute exists, it implies that the lower bound **bound_1** optional attribute of the defined list exists as well.

WR2: if the lower bound **bound_1** of the defined list optional attribute exists then it is greater or equal to 0.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

6.2.6 Array_type

The **array_type** entity provides for the definition of data types that may be expressed as an array of values. An array data type has as its domain indexed, fixed-size collection of like elements. The lower and upper bounds, which are integer values, define the range of index values and thus the size of each array collection. An array data type definition may optionally specify that an array value cannot contain duplicate elements.

EXPRESS specification:

```
*)
ENTITY array_type
SUBTYPE OF (aggregate_type);
      SELF\aggregate_type.bound_1: INTEGER;
      SELF\aggregate_type.bound_2: INTEGER;
      uniqueness: BOOLEAN;
      are_optional: BOOLEAN;
END_ENTITY;
(*
```

Attribute definition:

bound_1: the integer that defines the low index of the defined aggregate type.

bound_2: the integer that defines the upper index of the defined aggregate type.

uniqueness: indicates whether all elements of the array must must be present (false) or whether some elements of the array may be missing (true).

are_optional: indicates whether all elements of the array must be present (false) or whether some elements of the array may be missing (true).

6.2.7 Set_with_subset_constraint_type

The **set_with_subset_constraint_type** entity provides for the definition of data types that may be expressed as a set of values of which subsets may be extracted. The sizes of allowed subsets are defined by their minimal and maximal values. If these sizes do not exist, any subset is allowed.

NOTE The context in which subsets may be extracted is outside the scope of the part of ISO 13584.

EXPRESS specification:

```
*)
ENTITY set_with_subset_constraint_type
SUBTYPE OF (aggregate_type);
      cardinal_min: OPTIONAL INTEGER;
      cardinal_max: OPTIONAL INTEGER;
WHERE
      WR1: cardinal_min <= cardinal_max ;
      WR2: NOT EXISTS (bound_2) OR NOT EXISTS (cardinal_max)
           OR (cardinal_max <= bound_2);
      WR3: NOT EXISTS (bound_1) OR NOT EXISTS (cardinal_min)
           OR (cardinal_min <= bound_1);
END_ENTITY;
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
(*
```

Attribute definition:

cardinal_min: the minimal size of the subsets that may be extracted.

cardinal_max: the maximal size of the subsets that may be extracted.

Formal propositions:

WR1: the minimal size of the subsets that may be extracted **cardinal_min** shall be less of equal the maximal size of the subsets that may be extracted **cardinal_max**.

WR2: the maximal size of the subsets that may be extracted from the set shall not be greater than the maximal size of the set itself.

WR3: the minimal size of the subsets that may be extracted from the set shall not be greater than the minimal size of the set itself.

```
*)
END_SCHEMA;
-- ISO13584_IEC61360_dictionary_aggregate_extension_schema

(*
```

7 ISO13584_aggregate_value_schema

This clause defines the requirements for the **ISO13584_aggregate_value_schema**. The following EXPRESS declaration introduces **ISO13584_aggregate_value_schema** block and identifies the necessary external references.

EXPRESS specification:

```
*)
SCHEMA ISO13584_aggregate_value_schema;

REFERENCE FROM ISO13584_IEC61360_dictionary_schema(
      class_instance_type,
      data_type,
      data_type_element,
      level_type,
      named_type,
      property_BSU,
      property_DET
      );

REFERENCE FROM ISO13584_IEC61360_dictionary_aggregate_extension_schema(
      entity_instance_type_for_aggregate,
      list_type,
      set_type,
      bag_type,
      array_type,
      set_with_subset_constraint_type);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    REFERENCE FROM ISO13584_extended_dictionary_schema(
          data_type_type_name,
          data_type_typeof
          );

    REFERENCE FROM ISO13584_instance_resource_schema(
          compatible_class_and_class,
          compatible_level_type_and_instance,
          compatible_type_and_value,
          dic_class_instance,
          null_or_primitive_value,
          primitive_value,
          property_or_data_type_BSU,
          property_value,
          uncontrolled_entity_instance_value
          );

    (*
```

NOTE The schemas referenced above can be found in the following documents:

ISO13584_IEC61360_dictionary_schema	IEC 61360-2:1998
(which is duplicated for convenience in Informative Annex D of ISO 13584-42:1998)	
ISO13584_IEC61360_dictionary_aggregate_extension_schema	This part of ISO 13584
ISO13584_extended_dictionary_schema	ISO 13584-24:2003
ISO13584_instance_resource_schema	ISO 13584-24:2003

7.1 Introduction to the ISO13584_aggregate_value_schema

The **ISO13584_aggregate_value_schema** provides the information model for the extension to the **ISO_13584_instance_resource_schema,** which extends the scope of instance values to instances whose data types are aggregate data types that are lists, sets, bags, arrays or sets of subsets of simple or complex data types, as allowed by the extension of the ISO/IEC common dictionary schema defined in the **ISO13584_IEC61360_dictionary_aggregate_extension_schema** schema.

7.2 ISO13584_aggregate_value_schema entity definitions

The following entity type definitions describe the necessary resources needed to encode aggregate values according to the EXPRESS language.

7.2.1 Aggregate_entity_instance_value

The **aggregate_entity_instance_value** entity provides for the referencing of data values that may be expressed as aggregates of **primitive_value**s.

It is obtained by subtyping the **uncontrolled_entity_instance_value** provided by the **ISO13584_instance_resource_schema** defined in ISO 13584-24:2003.

EXPRESS specification:

```
    *)
    ENTITY aggregate_entity_instance_value
    SUBTYPE OF (uncontrolled_entity_instance_value);
          the_value: aggregate_value;
    END_ENTITY;
    (*
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Attribute definition:

the_value: the aggregate carrying all the values of the described entity.

7.2.2 Aggregate_value

The **aggregate_value** entity provides for the definition of data values that may be expressed as aggregates of **primitive_value**s.

EXPRESS specification:

```
*)
ENTITY aggregate_value
ABSTRACT SUPERTYPE OF(ONEOF(
          list_value,
          set_value,
          bag_value,
          array_value,
          set_with_subset_constraint_value ));
          values: LIST OF null_or_primitive_value;
END_ENTITY;
(*
```

Attribute definition:

values: is the list carrying all the aggregate values of the described entity.

7.2.3 List_value

The **list_value** entity provides for the definition of data values that may be expressed as lists of **primitive_value**s.

EXPRESS specification:

```
*)
ENTITY list_value
SUBTYPE OF (aggregate_value);
     SELF\aggregate_value.values: LIST OF primitive_value;
END_ENTITY;
(*
```

Attribute definition:

values: the list carrying all the values of the list described by the entity.

7.2.4 Set_value

The **set_value** entity provides for the definition of data values that may be expressed as sets of **primitive_value**s.

EXPRESS specification:

```
*)
ENTITY set_value
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    SUBTYPE OF (aggregate_value);
          SELF\aggregate_value.values: LIST OF primitive_value;
    WHERE
          WR1: VALUE_UNIQUE(values);
    END_ENTITY;
    (*
```

Attribute definition:

values: the list carrying all the values of the set described by the entity.

Formal propositions:

WR1: all the values of the **values** list shall be unique, duplicates are not allowed.

7.2.5 Bag_value

The **bag_value** entity provides for the definition of data values that may be expressed as bags of **primitive_value**s.

EXPRESS specification:

```
    *)
    ENTITY bag_value
    SUBTYPE OF (aggregate_value);
          SELF\aggregate_value.values: LIST OF primitive_value;
    END_ENTITY;
    (*
```

Attribute definition:

values: the list carrying all the values of the bag described by the entity.

7.2.6 Array_value

The **array_value** entity provides for the definition of data values that may be expressed as arrays of **null_or_primitive_value**s.

EXPRESS specification:

```
    *)
    ENTITY array_value
    SUBTYPE OF (aggregate_value);
          bound_1: INTEGER;
          bound_2: INTEGER;
    WHERE
          WR1: SIZEOF(SELF\aggregate_value.values) =
               SELF.bound_2 - SELF.bound_1 + 1;
    END_ENTITY;
    (*
```

--`,``,,-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Attribute definition:

bound_1: the low bound value of the described array.

bound_2: the high bound value of the described array.

Formal propositions:

WR1: the size of the **values** list attribute shall be equal to the size of defined by the **bound_1** and **bound_2** attributes.

7.2.7 Set_with_subset_constraint_value

The **set_with_subset_constraint_value** entity provides for the definition of data values that may be expressed as set of **primitive_value**s of which subsets may be extracted. The sizes of allowed subsets are defined either by their minimal and maximal values specified in **set_with_subset_constraint_type** data type, or in the **min** and **max** attributes . If none of the bounds exist, any subset is allowed.

NOTE 1 The context in which subsets may be extracted is outside the scope of the part of ISO 13584.

NOTE 2 If both **min** and **max**, and **cardinal_min** and **cardinal_max** exist, the rule **allowed_aggregate_value** rule ensures that the following holds: **cardinal_min** <= **min <= max <= cardinal_max**.

EXPRESS specification:

```
*)
ENTITY set_with_subset_constraint_value
SUBTYPE OF (aggregate_value);
     SELF\aggregate_value.values: LIST OF primitive_value;
     min: OPTIONAL INTEGER;
     max: OPTIONAL INTEGER;
WHERE
     WR1: NOT EXISTS (min) OR NOT EXISTS (max) OR (min <= max) ;
     WR2: VALUE_UNIQUE(values);
END_ENTITY;
(*
```

Attribute definition:

values: the list carrying all the values of the set of which subsets may be extracted.

min: the minimal size of the subsets that may be extracted.

max: the maximal size of the subsets that may be extracted.

Formal propositions:

WR1: the minimal size of the subsets that may be extracted **min** shall be less of equal the maximal size of the subsets that may be extracted **max**.

WR2: all the values of the **values** list shall be unique, duplicates are not allowed.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

7.3 ISO13584 _ aggregate_value_schema rule definition

7.3.1 Allowed_aggregate_values rule

The **allowed_aggregate_values** rule ensures that any **property_value** has its value compatible with the data type of the property even when this data type is an aggregate data type.

EXPRESS specification:

```
*)
RULE allowed_aggregate_values FOR (
     property_value,
     entity_instance_type_for_aggregate);

WHERE
     WR1: QUERY (prop <* property_value |
          NOT(compatible_complete_types_and_value(
          prop.prop_def, prop.its_value))) = [];
END_RULE; -- allowed_aggregate_values
(*
```

Formal propositions:

WR1: all the values associated to a **property_value** through the **its_value** attribute are type compatible with the data type defined by the **prop_def** attribute associated to the **property_BSU** carried by a **property_value**.

7.4 ISO13584_ aggregatevalue_schema function definitions

7.4.1 Compatible_complete_types_and_value function

The **compatible_complete_types_and_value** function completes the function **compatible_type_and_value** defined in ISO 13584-24:2003 by checking all the data types defined by this extension. It adds the checking of the aggregate values with respect to their defined data type.

When the value associated to a property is not an aggregate, then the **compatible_type_and_value** is called, otherwise, the **compatible_aggregate_domain_and_aggregate_value** function is called.

EXPRESS specification:

```
*)
FUNCTION compatible_complete_types_and_value(
     dom: property_or_data_type_BSU;
     val: primitive_value): LOGICAL;

IF (data_type_typeof(dom) = [])
THEN
     RETURN(UNKNOWN);
END_IF;

-- checking that values are primitive values but are not aggregate
-- values.
IF ('ISO13584_INSTANCE_RESOURCE_SCHEMA.PRIMITIVE_VALUE' IN TYPEOF(val))
     AND (NOT('ISO13584_AGGREGATE_VALUE_SCHEMA.' +
     'AGGREGATE_ENTITY_INSTANCE_VALUE' IN TYPEOF(val)))
THEN
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          RETURN(compatible_type_and_value(dom, val));
     END_IF;

     IF 'ISO13584_AGGREGATE_VALUE_SCHEMA.' +
          'AGGREGATE_ENTITY_INSTANCE_VALUE' IN TYPEOF(val)
     THEN
          RETURN(compatible_aggregate_domain_and_aggregate_value(
               dom, val));
     END_IF;

     -- neither primitive value nor aggregate value
     RETURN(UNKNOWN);

     END_FUNCTION; -- compatible_complete_types_and_value
     (*
```

7.4.2 Compatible_aggregate_domain_and_aggregate_value function

The **compatible_aggregate_domain_and_aggregate_value** function checks whether a given **dom property_or_data_type_BSU** is the correct data type for a given **val agregate_entity_instance_value**.

It returns unknown if the final data type associated to **dom** is unknown.

If the final type associated to the **dom property_or_data_type_BSU** does not reference an **aggregate_type** entity, then this function returns false.

When the final type of the **dom property_or_data_type_BSU** references an **aggregate_type** entity, the function calls the **compatible_aggregate_type_and_value** function with the **the_data_type** computed data type and the **val aggregate_entity_instance_value** as parameters.

EXPRESS specification:

```
     *)
     FUNCTION compatible_aggregate_domain_and_aggregate_value(
          dom: property_or_data_type_BSU;
          val: aggregate_entity_instance_value) : LOGICAL;

     LOCAL
          the_data_type: data_type;
     END_LOCAL;

     -- Check the avalability of the final type of a property or a
     -- data type BSU.
     IF data_type_typeof(dom) = []
     THEN (* the final domain of the type is not available *)
          RETURN(UNKNOWN);
     END_IF;

     -- Check that the final type of the property or data type BSU
     -- is an entity_instance_type whose type_name_attribute
     -- references the aggregate_type entity

     IF (NOT('ISO13584_IEC61360_DICTIONARY_SCHEMA.ENTITY_INSTANCE_TYPE' IN
     data_type_typeof(dom)))
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    THEN
         RETURN(FALSE);
    END_IF;

    IF (NOT('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA' +
         '.AGGREGATE_TYPE' IN data_type_type_name(dom)))
    THEN
         RETURN(FALSE);
    END_IF;

    -- Compute the final type of the dom property or data_type_BSU

    the_data_type := data_type_final_type(dom)[1];

    RETURN(compatible_aggregate_type_and_value(the_data_type, val));

    END_FUNCTION; -- compatible_aggregate_domain_and_value
    (*
```

7.4.3 Data_type_final_type function

The **data_type_final_type** function computes the **data_type** that defines the final domain of a **property_BSU** or a **data_type_BSU**.

If the **data_type** is associated with **named_type**s, the function recursively traverses their **referred_type**s attributes, until arriving either to a **data_type** that is either a **simple_type** or a **complex_type**. Then the function returns a set that contains this **data_type** entity.

If a BSU definition is not available, with the result that the function cannot be resolved to a **simple_type** or to a **complex_type**, the function returns an empty set of **data_type**s.

<u>EXPRESS specification:</u>

```
    *)
    FUNCTION data_type_final_type(
         type_spec: property_or_data_type_BSU): SET [0:1] OF data_type;

    LOCAL
         res: BOOLEAN := FALSE;
         x: data_type;
    END_LOCAL;

    IF NOT EXISTS(type_spec)
    THEN
         RETURN([]); -- type_spec is indeterminate
    END_IF;

    IF ('ISO13584_IEC61360_DICTIONARY_SCHEMA.PROPERTY_BSU' IN
         TYPEOF(type_spec))
    THEN
         IF NOT(SIZEOF(type_spec.definition) = 0)
         THEN
              x := type_spec.definition[1]\property_DET.domain;
              res := TRUE;
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
        END_IF;
    ELSE
        IF NOT(SIZEOF(type_spec.definition) = 0)
        THEN
            x := type_spec.definition[1]\data_type_element.
                type_definition;
            res := TRUE;
        END_IF;
    END_IF;

    IF NOT(res)
    THEN
        RETURN([]);
    END_IF;

    IF ('ISO13584_IEC61360_DICTIONARY_SCHEMA.NAMED_TYPE' IN TYPEOF(x))
    THEN
        IF NOT(SIZEOF(x\named_type.referred_type.definition) = 0)
        THEN
            RETURN(data_type_final_type(x\named_type.referred_type));
        ELSE
            RETURN([]);
        END_IF;
    ELSE
        RETURN([x]);
    END_IF;

    END_FUNCTION; -- data_type_final_type
    (*
```

7.4.4 Compatible_aggregate_type_and_value function

The **compatible_aggregate_type_and_value** function checks that an **entity_instance_type_for_aggregate** and **aggregate_entity_instance_value** pair are type compatible.

This function first computes the type of the elements contained in the aggregate **val**. It returns unknown if the final data type of the values in the aggregate is not available. If the type of the values of the aggregate is a **named_type**, the **data_type_final_type** function is called in order to retrieve the final type of the elements.

When the final type is computed, each possible aggregate type is checked (list, set, bag, array and sets of subsets).

For each aggregate, the function checks that lower and upper bounds of the values are compatible with lower and upper bounds of the values of the type declaration.

A particular processing is associated to the array aggregate data type.

For array data type and values, the function checks that the size of the **the_value** storage list of values is size compatible with the type declaration.

Then, in case of a list or an array of unique values, the function checks that no value in the **the_value** list is duplicated. For an array where optional values are not allowed, the function checks that no **null_value** is present in the **the_value** list.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Finally, this function checks the compatibility of all the values of the **the_value** list of the aggregate with their corresponding data type **type_of_elements**. It calls through an indirect recursion the **compatible_final_type_and_value** function to check type and values compatibility.

EXPRESS specification:

```
*)
FUNCTION compatible_aggregate_type_and_value(
      the_data_type: entity_instance_type_for_aggregate;
      val: aggregate_entity_instance_value): LOGICAL;

LOCAL
      elements: LIST OF null_or_primitive_value;
      type_of_elements: data_type;
      result: LOGICAL;
      tmp: LIST OF primitive_value := [];
END_LOCAL;

elements := val.the_value.values;

-- Compute type of elements contained in the aggregate
IF ('ISO13584_IEC61630_DICTIONARY_SCHEMA.NAMED_TYPE' IN
      TYPEOF(the_data_type.type_structure.value_type))
THEN
      IF (data_type_typeof(the_data_type.type_structure.
           value_type\named_type.referred_type) = [])
      THEN
      (* the final domain of the type of elements is not available *)
           RETURN(UNKNOWN);
      END_IF;
      type_of_elements := data_type_final_type(the_data_type.
           type_structure.value_type\named_type.referred_type)[1];
ELSE
      type_of_elements := the_data_type.type_structure.value_type;
END_IF;

-- check that a value aggregate is compatible with its type
-- aggregate declaration
IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.LIST_VALUE'
      IN TYPEOF (val.the_value))
      AND NOT
      ('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA'+
      '.LIST_TYPE' IN TYPEOF(the_data_type.type_structure))
THEN
      RETURN(FALSE);
END_IF;

IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.BAG_VALUE' IN TYPEOF(
      val.the_value)) AND NOT
      ('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.'+
      'BAG_TYPE' IN TYPEOF(the_data_type.type_structure))
THEN
      RETURN(FALSE);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
END_IF;

IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.SET_VALUE' IN TYPEOF(
     val.the_value)) AND NOT
     ('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.'+
     'SET_TYPE' IN TYPEOF(the_data_type.type_structure))
THEN
     RETURN(FALSE);
END_IF;

IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.ARRAY_VALUE' IN TYPEOF(
     val.the_value)) AND NOT
     ('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.'+
     'ARRAY_TYPE' IN TYPEOF(the_data_type.type_structure))
THEN
     RETURN(FALSE);
END_IF;

IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.SET_WITH_SUBSET_CONSTRAINT_VALUE'
          IN TYPEOF(val.the_value)) AND NOT
     ('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.'+
               'SET_WITH_SUBSET_CONSTRAINT_TYPE'
          IN TYPEOF(the_data_type.type_structure))
THEN
     RETURN(FALSE);
END_IF;

-- check that that low and high bounds of the values are compatible
-- with the type declaration.
IF   (('ISO13584_AGGREGATE_VALUE_SCHEMA.LIST_VALUE'
          IN TYPEOF (val.the_value))
     OR ('ISO13584_AGGREGATE_VALUE_SCHEMA.BAG_VALUE'
          IN TYPEOF (val.the_value))
     OR ('ISO13584_AGGREGATE_VALUE_SCHEMA.SET_VALUE'
          IN TYPEOF (val.the_value))
     OR
('ISO13584_AGGREGATE_VALUE_SCHEMA.SET_WITH_SUBSET_CONSTRAINT_VALUE'
          IN TYPEOF (val.the_value)))
     AND (
     (EXISTS(the_data_type.type_structure.bound_1))
     AND NOT
     (SIZEOF(elements) >= the_data_type.type_structure.bound_1)
     )
THEN
     RETURN(FALSE);
END_IF;

IF (('ISO13584_AGGREGATE_VALUE_SCHEMA.LIST_VALUE' IN TYPEOF(
          val.the_value))
     OR ('ISO13584_AGGREGATE_VALUE_SCHEMA.BAG_VALUE' IN TYPEOF(
          val.the_value))
     OR ('ISO13584_AGGREGATE_VALUE_SCHEMA.SET_VALUE' IN TYPEOF(
          val.the_value))
     OR
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
    ('ISO13584_AGGREGATE_VALUE_SCHEMA.SET_WITH_SUBSET_CONSTRAINT_VALUE'
              IN TYPEOF (val.the_value)))
          AND (
          (EXISTS(the_data_type.type_structure.bound_2))
          AND NOT
          (SIZEOF(elements) <= the_data_type.type_structure.bound_2))
    THEN
          RETURN(FALSE);
    END_IF;


    -- For array data type and values, check that size of the storage
    -- list of values is size compatible with the type declaration.
    IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.ARRAY_VALUE' IN TYPEOF(
          val.the_value))
    THEN
          IF (NOT(val.the_value\array_value.bound_1 =
               the_data_type.type_structure.bound_1) OR
               NOT(val.the_value\array_value.bound_2 =
               the_data_type.type_structure.bound_2))
          THEN
               RETURN(FALSE);
          END_IF;
    END_IF;

    -- For set_with_subset_constraint_type, check that the possible
    -— constraints on subset size defined at the value level are consistent
    -- with those defined at the type level.
    IF
    ('ISO13584_AGGREGATE_VALUE_SCHEMA.SET_WITH_SUBSET_CONSTRAINT_VALUE'
              IN TYPEOF (val.the_value))
          AND NOT
          ((the_data_type.type_structure.cardinal_min
          <= val.the_value\set_with_subset_constraint_value.min)
          AND
          (val.the_value\set_with_subset_constraint_value.min
          <= val.the_value\set_with_subset_constraint_value.max)
          AND
          (val.the_value\set_with_subset_constraint_value.max
          <= the_data_type.type_structure.cardinal_max))
    THEN
          RETURN(FALSE);
    END_IF;

    -- In case of a list or array of unique values, check that no
    -- value is duplicated
    IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.LIST_VALUE' IN TYPEOF(
          val.the_value)) OR
          ('ISO13584_AGGREGATE_VALUE_SCHEMA.ARRAY_VALUE'
          IN TYPEOF (val.the_value))
    THEN
          IF (the_data_type.type_structure.uniqueness)
          THEN
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          REPEAT i := 1 TO SIZEOF(val.the_value.values);
               IF NOT('ISO13584_INSTANCE_RESOURCE_SCHEMA.NULL_VALUE'
                    IN TYPEOF(val.the_value.values[i]))
               THEN
                    tmp := tmp + val.the_value.values[i];
               END_IF;
          END_REPEAT;

          IF NOT(VALUE_UNIQUE(tmp))
          THEN
               RETURN(FALSE);
          END_IF;
     END_IF;
END_IF;

-- For an array where optional values are not allowed, check that
-- no null_value is provided
IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.ARRAY_VALUE'
     IN TYPEOF(val.the_value))
THEN
     IF NOT(the_data_type.type_structure.are_optional)
     THEN
          REPEAT i := 1 TO SIZEOF(val.the_value.values);
               IF ('ISO13584_INSTANCE_RESOURCE_SCHEMA.NULL_VALUE'
                    IN TYPEOF(val.the_value.values[i]))
               THEN
                    RETURN(FALSE);
               END_IF;
          END_REPEAT;
     END_IF;
END_IF;

-- Check for value/type compatibility for all the elements contained
-- in the aggregate

result := TRUE;

REPEAT i := 1 TO SIZEOF(elements);
     IF NOT('ISO13584_AGGREGATE_VALUE_SCHEMA.NULL_VALUE'
           IN TYPEOF(elements[i]))
     THEN
          IF('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.'
               + 'ENTITY_INSTANCE_TYPE_FOR_AGGREGATE'
               IN TYPEOF(type_of_elements))
          THEN
               IF ('ISO13584_AGGREGATE_VALUE_SCHEMA.'+
                    'AGGREGATE_ENTITY_INSTANCE_VALUE' IN
                    TYPEOF(elements[i]))
               THEN
                    result := result AND
                         compatible_aggregate_type_and_value(
                         type_of_elements, elements[i]);
               ELSE
                    RETURN(FALSE);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
                    END_IF;
               ELSE
                    result := result AND compatible_final_type_and_value(
                         type_of_elements, elements[i]);
               END_IF;

               IF NOT(result)
               THEN
                    RETURN(FALSE);
               END_IF;
          END_IF;
    END_REPEAT;

    RETURN(result);

    END_FUNCTION; -- compatible_aggregate_type_and_value
    (*
```

7.4.5 Compatible_final_type_and_value function

The **compatible_final_type_and_value** function checks that a **data_type** and a **primitive_value** are type compatible.

This function checks the corresponding data type associated to **typ**.

For the **integer_value, real_value, number_value, boolean_value, translatable_string_value** and **entity_instance_value** values, the control is achieved locally in this function.

For **dic_class_instance** and **level_spec_value** values, the **compatible_class_and_class** and **compatible_level_type_and_instance** functions are respectively called.

This function returns false if none of these types is a possible type of the **val** parameter.

EXPRESS specification:

```
    *)
    FUNCTION compatible_final_type_and_value(
          typ: data_type; val: primitive_value): LOGICAL;

    LOCAL
          set_string: SET OF STRING := [];
          set_integer: SET OF INTEGER := [];
          code_type: non_quantitative_code_type;
          int_type: non_quantitative_int_type;
    END_LOCAL;

    IF ('ISO13584_INSTANCE_RESOURCE_SCHEMA.INTEGER_VALUE' IN TYPEOF(val))
    THEN
          IF (('ISO13584_IEC61360_DICTIONARY_SCHEMA.' +
               'NON_QUANTITATIVE_INT_TYPE' IN TYPEOF(typ)))
          THEN
               set_integer := [];
               code_type := typ;
```

--````,,-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          REPEAT j := 1 TO SIZEOF(code_type.domain.its_values);
               set_integer := set_integer +
                    code_type.domain.its_values[j].value_code;
          END_REPEAT;

          RETURN(val IN set_integer);
     ELSE
          RETURN(('ISO13584_IEC61360_DICTIONARY_SCHEMA.INT_TYPE'
               IN TYPEOF(typ)) OR
               (('ISO13584_IEC61360_DICTIONARY_SCHEMA.NUMBER_TYPE'
               IN TYPEOF(typ))
               AND NOT('ISO13584_IEC61360_DICTIONARY_SCHEMA.REAL_TYPE'
               IN TYPEOF(typ))));
     END_IF;
END_IF;

IF ('ISO13584_INSTANCE_RESOURCE_SCHEMA.REAL_VALUE' IN TYPEOF(val))
THEN
     RETURN(('ISO13584_IEC61360_DICTIONARY_SCHEMA.REAL_TYPE'
          IN TYPEOF(typ)) OR
          (('ISO13584_IEC61360_DICTIONARY_SCHEMA.NUMBER_TYPE'
          IN TYPEOF(typ))
          AND NOT('ISO13584_IEC61360_DICTIONARY_SCHEMA.INT_TYPE'
          IN TYPEOF(typ))));
END_IF;

IF ('ISO13584_INSTANCE_RESOURCE_SCHEMA.NUMBER_VALUE' IN TYPEOF(val))
THEN
     RETURN('ISO13584_IEC61360_DICTIONARY_SCHEMA.NUMBER_TYPE'
          IN TYPEOF(typ));
END_IF;

IF ('ISO13584_INSTANCE_RESOURCE_SCHEMA.BOOLEAN_VALUE' IN TYPEOF(val))
THEN
     RETURN('ISO13584_IEC61360_DICTIONARY_SCHEMA.BOOLEAN_TYPE'
          IN TYPEOF(typ));
END_IF;

IF ('ISO13584_INSTANCE_RESOURCE_SCHEMA.TRANSLATABLE_STRING_VALUE' IN
TYPEOF(val))
THEN
     IF (('ISO13584_IEC61360_DICTIONARY_SCHEMA' +
          '.NON_QUANTITATIVE_CODE_TYPE') IN TYPEOF(typ))
     THEN
               set_string := [];
               code_type := typ;
               REPEAT j := 1 TO SIZEOF(code_type.domain.its_values);
                    set_string := set_string +
                         code_type.domain.its_values[j].value_code;
               END_REPEAT;

               RETURN(('ISO13584_INSTANCE_RESOURCE_SCHEMA.STRING_VALUE'
                    IN TYPEOF(val)) AND (val IN set_string));
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
      ELSE
           RETURN('ISO13584_IEC61360_DICTIONARY_SCHEMA.STRING_TYPE'
                IN TYPEOF(typ));
      END_IF;
END_IF;

IF 'ISO13584_INSTANCE_RESOURCE_SCHEMA.ENTITY_INSTANCE_VALUE'
      IN TYPEOF(val)
THEN
      IF 'ISO13584_INSTANCE_RESOURCE_SCHEMA' +
           '.UNCONTROLLED_ENTITY_INSTANCE_VALUE' IN TYPEOF(val)
      THEN
           RETURN(UNKNOWN);
      END_IF;
      IF ('ISO13584_IEC61360_DICTIONARY_SCHEMA.ENTITY_INSTANCE_TYPE'
           IN TYPEOF(typ)) AND
           (typ\entity_instance_type.type_name <= TYPEOF(val))
      THEN
           RETURN(TRUE);
      ELSE
           RETURN(FALSE);
      END_IF;
END_IF;

IF 'ISO13584_INSTANCE_RESOURCE_SCHEMA.DIC_CLASS_INSTANCE'
      IN TYPEOF(val)
THEN
      IF ('ISO13584_IEC61360_DICTIONARY_SCHEMA.CLASS_INSTANCE_TYPE'
           IN TYPEOF(typ))
      THEN
           RETURN(compatible_class_and_class(typ\class_instance_type.
                domain,val\dic_class_instance.class_def));
      ELSE
           RETURN(FALSE);
      END_IF;
END_IF;

IF 'ISO13584_INSTANCE_RESOURCE_SCHEMA.LEVEL_SPEC_VALUE' IN TYPEOF(val)
THEN
      IF ('ISO13584_IEC61360_DICTIONARY_SCHEMA.LEVEL_TYPE'
           IN TYPEOF(typ))
      THEN
           RETURN(compatible_level_type_and_instance(
                typ\level_type.levels,
                TYPEOF(typ\level_type.value_type), val));
      ELSE
           RETURN(FALSE);
      END_IF;
END_IF;

RETURN(FALSE);

END_FUNCTION; -- compatible_final_type_and_value
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
(*

*)
END_SCHEMA; -- ISO13584_aggregate_value_schema;
(*
```

8 Library integrated information model 25

Conformance to the library integrated information model LIIM 25 includes satisfying the information requirements stated in the **ISO13584_25_IEC61360_5_liim_schema** schema presented in 8.1, the requirements to support standard data stated in the **ISO13584_25_IEC61360_5_conformance_schema** schema presented in Annex D, the requirements of the implementation method(s) supported and the relevant requirements of the normative references.

An implementation shall support at least the following implementation method: ISO 10303-21. Requirements with respect to implementation methods are specified in Annex E.

The **ISO13584_25_IEC61360_5_liim_schema** schema provides for a number of options that may be supported by an implementation. These options have been grouped into conformance classes. Nine conformance classes are defined. Options are defined by each class and may be selected by an implementation. Conformance to a particular conformance class requires that all the **ISO13584_25_IEC61360_5_liim_schema** entities, types and associated constraints defined as part of the class, be supported, together with the standard data associated with the class.

NOTE 1 Support of standard data associated with a class is insured by the global rule specified in the **ISO13584_25_IEC61360_5_conformance_schema.**

The numbering schema of the conformance classes is as follows:

— class 1: minimal **dictionary_element**s from the ISO/IEC common dictionary schema more aggregate types;

NOTE 2 The ISO/IEC common dictionary schema is defined by the **ISO13584_IEC61360_dictionary_schema** documented in ISO 13584-42:1998.

— class 2: **dictionary_element**s from the extended dictionary schema without functional models and functional views and without aggregate types;

NOTE 3 The extended dictionary schema is defined by the **ISO13584_extended_dictionary_schema** documented in ISO 13584-24.

— class 3: **dictionary_element**s from the extended dictionary schema with functional models and functional views and aggregate types;

— class 4: identical to class 3 but with limited nested aggregate types;

— class 5: **dictionary_element**s from the extended dictionary schema without functional models and functional view classes and without aggregate types and values, but with explicit description of **class_extension**s for the classes in the library;

— class 6: **dictionary_element**s from the extended dictionary schema with functional models and functional views, aggregate types and values and with explicit description of **class_extension**s for the classes in the library;

— class 7: identical to class 6 but with limited nested aggregate types and values;

— class 10: item instances and item representation instances without dictionary definitions and without library structure;

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

— class 11: item instances and item representation instances with dictionary definitions but without library structure.

NOTE 4 The attribute values for the **external_file_protocol** entities that do not belong to the standard data defined in Annex D of this part of ISO 13584 or to the standard data defined in one part of the view exchange protocol series of part of ISO 13584 are subject to prior agreement between the sender and the receiver. They are outside the scope of this International Standard.

NOTE 5 The only files that may be referenced as **http_file**s in conformance classes 2 to 8 and 10 to 11 of library integrated information model 25 are files whose MIME type and subtype:

— either corresponding to specifications that are publicly available, or

— that are associated with public domain Internet-available readers.

Table 1 shows the supported capabilities of the different conformance classes of library integrated information model 25.

Table 1 — Conformance options of library integrated information model 25

Capabilities	**Dictionary elements**			**Library specification (class extension)**	**Instance representation**
Conformance class	Dictionary definitions of item classes	Dictionary definitions of item class representations and representation categories	Aggregate structured properties		
1	x		x		
2	x				
3	x	x	x		
4	x	x	x		
5	x			x	x
6	x	x	x	x	x
7	x	x	x	x	x
10			x		x
11	x	x	x		x

8.1 ISO13584_25_IEC61360_5_liim_schema short listing

This clause specifies the EXPRESS schema that uses elements from the integrated resource series of ISO 10303 and from the logical resource and description methodology series of parts of ISO 13584 to define the requirements of the library integrated information model LIIM25 specified in this part of ISO 13584.

NOTE 1 The integrated resource series of ISO 10303 are part 10303-4x and 10303-1xx. The logical resource series of parts of ISO 13584 are ISO 13584-2x and the description methodology series of parts of ISO 13584 are ISO 13584-4x.

The expanded EXPRESS listing of the **ISO13584_25_IEC61360_5_liim_schema**, with the additional constraints defined in **ISO13584_25_IEC61360_5_conformance_schema**, is presented in Annex A. The resulting schema, called **ISO13584_25_IEC61360_5_library_implicit_schema**, is the information

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

model of supplier libraries that reference the library integrated information model LIIM 25 specified in this part of ISO 13584, whether or not they also reference some view exchange protocols.

NOTE 2 The information model of integrated libraries is outside the scope of this International Standard.

EXPRESS specification:

```
*)
SCHEMA ISO13584_25_IEC61360_5_liim_schema;

USE FROM ISO13584_IEC61360_dictionary_schema
     (axis1_placement_type,
     axis2_placement_2d_type,
     axis2_placement_3d_type,
     boolean_type,
     class_BSU,
     class_instance_type,
     class_value_assignment,
     complex_type,
     component_class,
     condition_DET,
     data_type_BSU,
     data_type_element,
     dates,
     dependent_P_DET,
     dic_unit,
     dic_value,
     entity_instance_type,
     identified_document,
     int_currency_type,
     int_measure_type,
     int_type,
     integer_type,
     item_class,
     item_names,
     label_with_language,
     level_type,
     material_class,
     mathematical_string,
     named_type,
     non_dependent_P_DET,
     non_quantitative_code_type,
     non_quantitative_int_type,
     non_si_unit,
     number_type,
     placement_type,
     property_BSU,
     property_DET,
     real_currency_type,
     real_measure_type,
     real_type,
     string_type,
     supplier_BSU,
     supplier_element,
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
      value_domain);

USE FROM ISO13584_IEC61360_language_resource_schema
      (global_language_assignment,
      present_translations,
      translated_label,
      translated_text);

USE FROM ISO13584_instance_resource_schema
      (null_value,
      primitive_value,
      null_or_primitive_value,
      simple_value,
      null_or_simple_value,
      number_value,
      null_or_number_value,
      integer_value,
      null_or_integer_value,
      real_value,
      null_or_real_value,
      boolean_value,
      null_or_boolean_value,
      translatable_string_value,
      translated_string_value,
      string_value,
      null_or_translatable_string_value,
      complex_value,
      null_or_complex_value,
      entity_instance_value,
      null_or_entity_instance_value,
      defined_entity_instance_value,
      controlled_entity_instance_value,
      STEP_entity_instance_value,
      PLIB_entity_instance_value,
      property_or_data_type_BSU,
      level_spec_value,
      null_or_level_spec_value,
      int_level_spec_value,
      null_or_int_level_spec_value,
      real_level_spec_value,
      null_or_real_level_spec_value,
      property_value,
      context_dependent_property_value,
      dic_class_instance,
      null_or_dic_class_instance,
      dic_component_instance,
      dic_feature_instance,
      dic_material_instance,
      lib_component_instance,
      lib_feature_instance,
      lib_material_instance,
      dic_f_model_instance,
      lib_f_model_instance);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
USE FROM ISO13584_IEC61360_dictionary_aggregate_extension_schema
      (entity_instance_type_for_aggregate,
      list_type,
      set_type,
      bag_type,
      array_type,
      set_with_subset_constraint_type);

USE FROM ISO13584_extended_dictionary_schema
      (dictionary,
      dictionary_in_standard_format,
      library_iim_identification,
      view_exchange_protocol_identification,
      representation_type,
      geometric_representation_context_type,
      representation_reference_type,
      program_reference_type,
      program_library_BSU,
      document_BSU,
      supplier_program_library_relationship,
      class_document_relationship,
      representation_P_DET,
      class_related_dictionary_element,
      program_library_element,
      document_element,
      document_element_with_http_access,
      document_element_with_translated_http_access,
      referenced_document,
      referenced_graphics,
      feature_class,
      functional_model_class,
      fm_class_view_of,
      functional_view_class,
      non_instantiable_functional_view_class,
      view_control_variable_range,
      item_class_case_of,
      component_class_case_of,
      material_class_case_of,
      feature_class_case_of,
      a_posteriori_case_of,
      a_posteriori_view_of);

USE FROM ISO13584_external_file_schema
      (standard_simple_program_protocol,
      non_standard_simple_program_protocol,
      linked_interface_program_protocol,
      standard_data_protocol,
      non_standard_data_protocol,
      http_protocol,
      program_library_content,
      document_content,
      representation_reference,
      program_reference,
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
           property_value_external_item,
           message,
           illustration,
           A6_illustration,
           A9_illustration,
           translated_external_content,
           not_translated_external_content,
           not_translatable_external_content,
           language_specific_content,
           external_file_unit,
           http_file,
           http_class_directory,
           simple_program_protocol);


     USE FROM ISO13584_aggregate_value_schema
           (aggregate_entity_instance_value,
           list_value,
           set_value,
           bag_value,
           array_value,
           set_with_subset_constraint_value);

     USE FROM ISO13584_library_content_schema
           (library,
           library_in_standard_format,
           explicit_item_class_extension,
           explicit_functional_model_class_extension,
           property_classification,
           property_value_recommended_presentation);

     USE FROM measure_schema
           (amount_of_substance_measure,
           area_measure,
           context_dependent_measure,
           context_dependent_unit,
           conversion_based_unit,
           count_measure,
           derived_unit,
           derived_unit_element,
           dimensional_exponents,
           electric_current_measure,
           global_unit_assigned_context,
           length_measure,
           length_measure_with_unit,
           length_unit,
           luminous_intensity_measure,
           mass_measure,
           measure_value,
           measure_with_unit,
           named_unit,
           numeric_measure,
           parameter_value,
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
      plane_angle_measure,
      positive_length_measure,
      positive_plane_angle_measure,
      ratio_measure,
      si_unit,
      solid_angle_measure,
      thermodynamic_temperature_measure,
      time_measure,
      volume_measure);

USE FROM person_organization_schema
      (address,
      organization,
      person);

USE FROM date_time_schema
      (date,
      date_and_time,
      local_time,
      calendar_date,
      ordinal_date,
      week_of_year_and_day_date);

USE FROM geometry_schema
      (axis1_placement,
      axis2_placement_2D,
      axis2_placement_3D,
      geometric_representation_context,
      placement);

USE FROM representation_schema
      (representation,
      representation_context,
      representation_item);

USE FROM application_context_schema
      (application_context,
      application_context_element,
      application_protocol_definition);

END_SCHEMA; -- ISO13584_25_IEC61360_5_liim_schema
(*
```

NOTE The schemas referenced above can be found in the following documents:

ISO13584_IEC61360_dictionary_schema	IEC 61360-2:1998
(which is duplicated for convenience in informative Annex D of ISO 13584-42:1998),	
ISO13584_IEC61360_language_resource_schema	IEC 61360-2:1998
(which is duplicated for convenience in informative Annex D of ISO 13584-42:1998),	
ISO13584_instance_resource_schema	ISO 13584-24:2003,
ISO13584_IEC61360_dictionary_aggregate_extension_schema	This part of ISO 13584,
ISO13584_extended_dictionary_schema	ISO 13584-24:2003,
ISO13584_external_file_schema	ISO 13584-24:2003,
ISO13584_aggregate_value_schema	This part of ISO 13584,
ISO13584_library_content_schema	ISO 13584-24:2003,
measure_schema	ISO 10303-41:2000,
person_organization_schema	ISO 10303-41:2000,
date_time_schema	ISO 10303-41:2000,
geometry_schema	ISO 10303-42:2000,

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

representation_schema	ISO 10303-43:2000,
application_context_schema	ISO 10303-41:2000.

8.2 Conformance class requirements

8.2.1 Conformance class 1: minimal dictionaries

Conformance class 1 addresses those implementations that are intended to support the common requirements stated in the ISO/IEC dictionary schema and its extension which handles aggregate data types and values. An implementation of conformance class 1 of library integrated information model 25 shall support the following entities and related constructs.

```
FROM ISO13584_IEC61360_dictionary_schema
        supplier_BSU
        supplier_element
        class_BSU
        item_class
        component_class
        material_class
        property_BSU
        property_DET
        condition_DET
        dependent_P_DET
        non_dependent_P_DET
        class_value_assignment
        data_type_BSU
        data_type_element
        number_type
        int_type
        int_measure_type
        int_currency_type
        integer_type
        non_quantitative_int_type
        real_type
        real_measure_type
        real_currency_type
        boolean_type
        string_type
        non_quantitative_code_type
        complex_type
        level_type
        class_instance_type
        entity_instance_type
        placement_type
        axis1_placement_type
        axis2_placement_2d_type
        axis2_placement_3d_type
        named_type
        value_domain
        dic_value
        non_si_unit
        dic_unit
        dates
        identified_document
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          item_names
          label_with_language
          mathematical_string

FROM ISO13584_IEC61360_language_resource_schema
          global_language_assignment
          present_translations
          translated_label
          translated_text

FROM ISO13584_IEC61360_dictionary_aggregate_extension_schema
          aggregate_entity_instance_type
          list_type
          set_type
          bag_type
          array_type
          set_with_subset_constraint_type

FROM measure_schema
          amount_of_substance_measure
          area_measure
          context_dependent_measure
          context_dependent_unit
          conversion_based_unit
          count_measure
          derived_unit
          derived_unit_element
          dimensional_exponents
          electric_current_measure
          global_unit_assigned_context
          length_measure
          length_measure_with_unit
          length_unit
          luminous_intensity_measure
          mass_measure
          measure_value
          measure_with_unit
          named_unit
          numeric_measure
          parameter_value
          plane_angle_measure
          positive_length_measure
          positive_plane_angle_measure
          ratio_measure
          si_unit
          solid_angle_measure
          thermodynamic_temperature_measure
          time_measure
          volume_measure

FROM person_organization_schema
          address
          organization
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

8.2.2 Conformance class 2: dictionaries of items classes

Conformance class 2 addresses those implementations that support **dictionary_element**s from the extended dictionary schema without functional model and functional view classes and without aggregate types,. An implementation of conformance class 2 of library integrated information model 25 shall support the following entities and related constructs.

FROM ISO13584_IEC61360_dictionary_schema

- supplier_BSU
- supplier_element
- class_BSU
- item_class
- component_class
- material_class
- property_BSU
- property_DET
- condition_DET
- dependent_P_DET
- non_dependent_P_DET
- class_value_assignment
- data_type_BSU
- data_type_element
- number_type
- int_type
- int_measure_type
- int_currency_type
- integer_type
- non_quantitative_int_type
- real_type
- real_measure_type
- real_currency_type
- boolean_type
- string_type
- non_quantitative_code_type
- complex_type
- level_type
- class_instance_type
- entity_instance_type
- placement_type
- axis1_placement_type
- axis2_placement_2d_type
- axis2_placement_3d_type
- named_type
- value_domain
- dic_value
- non_si_unit
- dic_unit
- dates
- identified_document
- item_names
- label_with_language
- mathematical_string

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
FROM ISO13584_IEC61360_language_resource_schema
        global_language_assignment
        present_translations
        translated_label
        translated_text

FROM ISO13584_extended_dictionary_schema
        dictionary
        dictionary_in_standard_format
        library_iim_identification
        view_exchange_protocol_identification
        document_BSU
        class_document_relationship
        representation_P_DET
        class_related_dictionary_element
        document_element
        document_element_with_http_access
        documented_element_with_translated_http_access
        referenced_document
        referenced_graphics
        feature_class
        item_class_case_of
        component_class_case_of
        material_class_case_of
        feature_class_case_of
        a_posteriori_case_of
        a_posteriori_view_of

FROM ISO13584_external_file_schema
        standard_data_protocol
        non_standard_data_protocol
        http_protocol
        document_content
        translated_external_content
        not_translated_external_content
        not_translatable_external_content
        language_specific_content
        external_file_unit
        http_file
        http_class_directory
        simple_program_protocol

FROM measure_schema
        amount_of_substance_measure
        area_measure
        context_dependent_measure
        context_dependent_unit
        conversion_based_unit
        count_measure
        derived_unit
        derived_unit_element
        dimensional_exponents
        electric_current_measure
        global_unit_assigned_context
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

length_measure
length_measure_with_unit
length_unit
luminous_intensity_measure
mass_measure
measure_value
measure_with_unit
named_unit
numeric_measure
parameter_value
plane_angle_measure
positive_length_measure
ratio_measure
si_unit
solid_angle_measure
thermodynamic_temperature_measure
time_measure
volume_measure

FROM person_organization_schema
address
organization

8.2.3 Conformance class 3: complete dictionaries

Conformance class 3 addresses those implementations that support conformance class 2 and that support both functional model and functional view classes and aggregate data types. An implementation of conformance class 3 of library integrated information model 25 shall support all the entities supported by conformance class 2 more the following entities and related constructs.

FROM ISO13584_extended_dictionary_schema
representation_type
geometric_representation_context_type
representation_reference_type
supplier_program_library_relationship
functional_model_class
fm_class_view_of
functional_view_class
non_instantiable_functional_view_class
view_control_variable_range

FROM ISO13584_external_file_schema
standard_simple_program_protocol
non_standard_simple_program_protocol
linked_interface_program_protocol
program_library_content
representation_reference
program_reference

FROM ISO13584_IEC61360_dictionary_aggregate_extension_schema
aggregate_entity_instance_type

--`,``,,-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

list_type
set_type
bag_type
array_type
set_with_subset_constraint_type

8.2.4 Conformance class 4: complete dictionaries with limited nested aggregate values

Conformance class 4 addresses those implementations that support all the entities and associated constructs defined for conformance class 3 but with the restriction that the level of nesting of aggregates is limited to 2 by the **nesting_level_aggregate_limit_rule** rule defined in Annex D.

8.2.5 Conformance class 5: libraries of item classes

Conformance class 5 addresses those implementations that support conformance class 2 and explicit description of item class extensions by means of definition of their set of instances. Conformance class 5 does not support functional model instances, nor an aggregate-structured values. An implementation of conformance class 5 of library integrated information model 25 shall support the entities defined for conformance class 2 more following entities and related constructs.

FROM ISO13584_extended_dictionary_schema

FROM ISO13584_external_file_schema
property_value_external_item,
message,
illustration,
A6_illustration,
A9_illustration,

FROM ISO13584_instance_resource_schema
null_value
primitive_value
null_or_primitive_value
simple_value
null_or_simple_value
number_value
null_or_number_value
integer_value
null_or_integer_value
real_value
null_or_real_value
boolean_value
null_or_boolean_value
translatable_string_value
translated_string_value
string_value
null_or_translatable_string_value
complex_value
null_or_complex_value
entity_instance_value
null_or_entity_instance_value
defined_entity_instance_value

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          controlled_entity_instance_value
          STEP_entity_instance_value
          PLIB_entity_instance_value
          property_or_data_type_BSU
          level_spec_value
          null_or_level_spec_value
          Int_level_spec_value
          null_or_int_level_spec_value
          real_level_spec_value
          null_or_real_level_spec_value
          property_value
          context_dependent_property_value
          dic_class_instance
          null_or_dic_class_instance
          dic_component_instance
          dic_feature_instance
          dic_material_instance
          lib_component_instance
          lib_feature_instance
          lib_material_instance

    FROM ISO13584_library_content_schema
          library
          library_in_standard_format
          explicit_item_class_extension
          property_classification
          property_value_recommended_presentation


    FROM person_organization_schema
          person

    FROM date_time_schema
          date
          date_and_time
          local_time
          calendar_date
          ordinal_date
          week_of_year_and_day_date

    FROM geometry_schema
          axis1_placement
          axis2_placement_2D
          axis2_placement_3D
          geometric_representation_context
          placement

    FROM representation_schema
          representation
          representation_context
          representation_item

    FROM application_context_schema
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

application_context
application_context_element
application_protocol_definition

8.2.6 Conformance class 6: complete libraries

Conformance class 6 addresses those implementations that support conformance class 5 more explicit description of functional model class extensions and aggregate-structured prepares values. An implementation of conformance class 6 of library integrated information model 25 shall support all the entities supported by conformance class 5 more the following entities and related constructs.

FROM ISO13584_extended_dictionary_schema
representation_type
geometric_representation_context_type
representation_reference_type
supplier_program_library_relationship
program_reference_type
program_library_BSU
program_library_element
functional_model_class
fm_class_view_of
functional_view_class
non_instantiable_functional_view_class
view_control_variable_range
a_posteriori_view_of

FROM ISO13584_external_file_schema
standard_simple_program_protocol
non_standard_simple_program_protocol
linked_interface_program_protocol
program_library_content
representation_reference
program_reference

FROM ISO13584_instance_resource_schema
dic_f_model_instance
lib_f_model_instance

FROM ISO13584_library_content_schema
explicit_functional_model_class_extension

FROM ISO13584_IEC61360_dictionary_aggregate_extension_schema
entity_instance_type_for_aggregate
list_type
set_type
bag_type
array_type
set_with_subset_constraint_type

FROM ISO13584_aggregate_value_schema
(aggregate_entity_instance_value
list_value
set_value
bag_value
array_value

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          set_with_subset_constraint_value
  FROM person_organization_schema
          person

  FROM date_time_schema
          date
          date_and_time
          local_time
          calendar_date
          ordinal_date
          week_of_year_and_day_date
```

8.2.7 Conformance class 7: complete libraries with limited nested aggregate values

Conformance class 7 addresses those implementations that support all the entities and associated constructs defined for conformance class 6 but with the restriction that the level of nesting of aggregates is limited to 2 by the **nesting_level_aggregate_limit_rule** rule defined in Annex D.

8.2.8 Conformance class 10: library instances

Conformance class 10 addresses those implementations that support description of item class instances or of item representation instances without dictionary definition and without library structure. An implementation of conformance class 10 of library integrated information model 25 shall support all the following entities and related constructs.

NOTE Conformance class 10 does not need the use of any **dictionary** or **library** entity If no view exchange protocol is used for instance representation

```
  FROM ISO13584_IEC61360_dictionary_schema
          supplier_BSU
          supplier_element
          class_BSU
          property_BSU
          data_type_BSU
          dic_value
          dates
          identified_document
          item_names
          label_with_language
          mathematical_string

  FROM ISO13584_IEC61360_language_resource_schema
          global_language_assignment
          present_translations
          translated_label
          translated_text


  FROM ISO13584_instance_resource_schema
          null_value
          primitive_value
          null_or_primitive_value
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          simple_value
          null_or_simple_value
          number_value
          null_or_number_value
          integer_value
          null_or_integer_value
          real_value
          null_or_real_value
          boolean_value
          null_or_boolean_value
          translatable_string_value
          translated_string_value
          string_value
          null_or_translatable_string_value
          complex_value
          null_or_complex_value
          entity_instance_value
          null_or_entity_instance_value
          defined_entity_instance_value
          controlled_entity_instance_value
          STEP_entity_instance_value
          PLIB_entity_instance_value
          property_or_data_type_BSU
          level_spec_value
          null_or_level_spec_value
          Int_level_spec_value
          null_or_int_level_spec_value
          real_level_spec_value
          null_or_real_level_spec_value
          property_value
          context_dependent_property_value
          dic_class_instance
          null_or_dic_class_instance
          dic_component_instance
          dic_feature_instance
          dic_material_instance
          lib_component_instance
          lib_feature_instance
          lib_material_instance
          dic_f_model_instance
          lib_f_model_instance

    FROM ISO13584_extended_dictionary_schema
          dictionary
          dictionary_in_standard_format
          library_iim_identification
          view_exchange_protocol_identification
          program_library_BSU
          document_element
          document_element_with_http_access
          documented_element_with_translated_http_access
          referenced_document
          referenced_graphics
          document_BSU
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
        class_document_relationship

FROM ISO13584_external_file_schema
        http_protocol
        document_content
        translated_external_content
        not_translated_external_content
        not_translatable_external_content
        language_specific_content
        external_file_unit
        http_file
        property_value_external_item

FROM ISO13584_aggregate_value_schema
        aggregate_entity_instance_value
        list_value
        set_value
        bag_value
        array_value
        set_with_subset_constraint_value

 FROM ISO13584_library_content_schema
        library
        library_in_standard_format


FROM person_organization_schema
        address
        organization
        person

FROM date_time_schema
        date
        date_and_time
        local_time
        calendar_date
        ordinal_date
        week_of_year_and_day_date
```

8.2.9 Conformance class 11: library instances with asociated dictionary definitions

Conformance class 11 addresses those implementations that support description of item class instances or of item representation instances with dictionary definition but without library structure. An implementation of conformance class 11 of library integrated information model 25 shall support all the entities defined for conformance class 10 more the following entities and related constructs.

NOTE Conformance class 11 does not need the use of any **dictionary** or **library** entity if no view exchange protocol is used for instance representation.

```
FROM ISO13584_IEC61360_dictionary_schema
        item_class
        component_class
        material_class
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          property_DET
          condition_DET
          dependent_P_DET
          non_dependent_P_DET
          class_value_assignment
          data_type_element
          number_type
          int_type
          int_measure_type
          int_currency_type
          integer_type
          non_quantitative_int_type
          real_type
          real_measure_type
          real_currency_type
          boolean_type
          string_type
          non_quantitative_code_type
          complex_type
          level_type
          class_instance_type
          entity_instance_type
          placement_type
          axis1_placement_type
          axis2_placement_2d_type
          axis2_placement_3d_type
          named_type
          value_domain
          non_si_unit
          dic_unit

     FROM ISO13584_extended_dictionary_schema
          representation_P_DET
          class_related_dictionary_element
          feature_class
          item_class_case_of
          component_class_case_of
          material_class_case_of
          feature_class_case_of
          a_posteriori_case_of
          a_posteriori_view_of
          representation_type
          geometric_representation_context_type
          representation_reference_type
          supplier_program_library_relationship
          functional_model_class
          fm_class_view_of
          functional_view_class
          non_instantiable_functional_view_class
          view_control_variable_range


     FROM ISO13584_external_file_schema
          standard_data_protocol
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
                    non_standard_data_protocol
                    http_class_directory
                    simple_program_protocol
                    standard_simple_program_protocol,
                    non_standard_simple_program_protocol,
                    linked_interface_program_protocol
                    representation_reference
                    program_reference

          FROM measure_schema
                    amount_of_substance_measure
                    area_measure
                    context_dependent_measure
                    context_dependent_unit
                    conversion_based_unit
                    count_measure
                    derived_unit
                    derived_unit_element
                    dimensional_exponents
                    electric_current_measure
                    length_measure
                    length_measure_with_unit
                    length_unit
                    luminous_intensity_measure
                    mass_measure
                    measure_value
                    measure_with_unit
                    named_unit
                    numeric_measure
                    parameter_value
                    plane_angle_measure
                    positive_length_measure
                    positive_plane_angle_measure
                    ratio_measure
                    si_unit
                    solid_angle_measure
                    thermodynamic_temperature_measure
                    time_measure
                    volume_measure

          FROM ISO13584_IEC61360_dictionary_aggregate_extension_schema
                    aggregate_entity_instance_type
                    list_type
                    set_type
                    bag_type
                    array_type
                    set_with_subset_constraint_type

          FROM geometry_schema
                    axis1_placement
                    axis2_placement_2D
                    axis2_placement_3D
                    geometric_representation_context
```

© ISO 2004 All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

```
          placement

FROM representation_schema
          representation
          representation_context
          representation_item

FROM application_context_schema
          application_context
          application_context_element
          application_protocol_definition
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Annex A (normative) Short names of entities defined in this part

Table A.1 provides the short names of entities specified in this part of ISO 13584. Requirements on the use of short names are found in the implementation methods included in ISO 10303.

A listing of the EXP RESS entity names and corresponding short names for the EXPRESS schema specified in this part of ISO 13584 is available in computer-interpretable form and can be found at the following URL:

http://www.tc184-sc4.org/Short_Names/

NOTE The information provi ded in computer-interpretable form at the above URLs is informative. The information that is contained in the body of this part of ISO 10303 is normative.

Table A.1 — Short names of entities

Long Name	Short Name
AGGREGATE_ENTITY_INSTANCE_VALUE	AEIV
AGGREGATE_TYPE	AGGTYP
AGGREGATE_VALUE	AGGVL
ARRAY_TYPE	ARRTYP
ARRAY_VALUE	ARRVL
BAG_TYPE	BGTYP
BAG_VALUE	BGVL
ENTITY_INSTANCE_TYPE_FOR_AGGREGATE	EITFA
LIST_TYPE	LSTTYP
LIST_VALUE	LSTVL
SET_TYPE	STTYP
SET_WITH_SUBSET_CONSTRAINT_TYPE	SWSCT
SET_VALUE	STVL
SET_WITH_SUBSET_CONSTRAINT_VALUE	SWSCV

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Annex B (normative)
Information object registration

B.1 Document identification

In order to provide for unambiguous identification of an information object in an open system, the object identifier:

{ iso standard 13584 part (25) version(1) }

is assigned to this part of ISO 13584. The meaning of this value is defined in ISO 8824-1, and is described in ISO 13584-1.

B.2 Schema identification

B.2.1 ISO13584_IEC61360_dictionary_aggregate_extension_schema

The **ISO13584_IEC61360_dictionary_aggregate_extension_schema** (see Clause 6) is assigned the object identifier:

{ iso standard 13584 part (25) version(1) object(1) ISO13584-IEC61360-dictionary-aggregate-extension-schema (1) }

B.2.2 ISO13584_aggregate_value_schema

The **ISO13584_aggregate_value_schema** (see Clause 7) is assigned the object identifier:

{ iso standard 13584 part (25) version(1) object(1) ISO13584-aggregate-value-schema (2) }

B.3.2 ISO13584_25_IEC61360_5_liim_schema

The **ISO13584_25_IEC61360_5_liim_schema** (see Clause 8.1) is assigned the object identifier:

{ iso standard 13584 part (25) version(1) object(1) ISO13584-25-liim-schema (3) }

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Annex C (normative) ISO13584_25_IEC61360_5_library_implicit_schema expanded listing

This annex references a listing of the complete EXPRESS schemas specified in Clause 8 of this part of ISO 13584 without comments or other explanatory text but with the additional constraints defined in **ISO13584_25_IEC61360_5_conformance_schema** defined in Annex D. The name of this schema is **ISO13584_25_IEC61360_5_library_implicit_schema**. This listing incorporates all the elements that the corresponding short form schema in Clause 8 uses from other schemas into a single schema without any external references

This schema may be used:

— to exchange libraries that reference the **ISO13584_25_IEC61360_5_liim_schema** and its associated **ISO13584_25_IEC61360_5_conformance_schema**, but that do not reference any view exchange protocol, and

— to exchange libraries that reference the **ISO13584_25_IEC61360_5_liim_schema** and its associated **ISO13584_25_IEC61360_5_conformance_schema**, and that do reference some view exchange protocols; in this case, the constraints defined in these view exchange protocols are not checked.

This schema may also be completed to check the constraints defined in all the referenced view exchange protocols using the following process for each referenced view exchange protocol.

Assume that V1 is a referenced view exchange protocol and that it specifies two constraint schemas of which schema names are S1_V1, S2_V1.

a) Check that all the entities referenced in the S1_V1 schema and in the S2_V1 schema are already existing in the **ISO13584_25_IEC61360_5_library_implicit_schema**, else reference to the library integrated information model 25 and to the view exchange protocol S1 by a same library delivery file is not allowed.

NOTE 1 The information model of a library delivery file and the entities it may contain are specified by a library integrated information model. A view exchange protocol may only add constraints.

b) Build the long form of the S1_V1 schema and give to the resulting schema the same name: "S1_V1";

c) Build the long form of the S2_V1 schema and give to the resulting schema the same name: "S2_V1";

d) Replace everywhere in the long form of the S1_V1 schema, the string "S1_V1" by '**ISO13584_25_IEC61360_5_library_implicit_schema**' with the same case;

e) Replace everywhere in the long form of the S2_V1 schema, the string "S2_V1" by '**ISO13584_25_IEC61360_5_library_implicit_schema**' with the same case;

f) Add the content of the long form of the S1_V1 schema to the content of the **ISO13584_25_IEC61360_5_library_implicit_schema**, removing possible duplicates;

g) Add the content of the long form of the S2_V1 schema to the content of the **ISO13584_25_IEC61360_5_library_implicit_schema**, removing possible duplicates.

When the above process is performed for view exchange protocols V1, V2,...Vn, the resulting **ISO13584_25_IEC61360_5_library_implicit_schema** may be used for exchanging any library that references the **ISO13584_25_IEC61360_5_liim_schema** and its associated

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

ISO13584_25_IEC61360_5_conformance_schema as its library integrated information model, and that references whole or part of the V1, V2,...Vn view exchange protocols set. This schema also includes the constraints of all the referenced view exchange protocols.

The listing of the **ISO13584_25_IEC61360_5_library_implicit_schema** schema is available in computer-interpretable form and can be found at the following URL:

http://www.tc184-sc4.org/EXPRESS/

If there is difficulty accessing these sites contact ISO Central Secretariat or contact the ISO TC 184/SC4 Secretariat directly at: sc4sec@tc184-sc4.org

NOTE 2 The information provided in computer-interpretable form at the above URLs is normative.

NOTE 3 If some errors are identified in the EXPRESS code during the ballot process, the description of these errors, together with the corrections recommended for PLIB implementations by the part editors can be found at the following URL:

http://www.lisi.ensma.fr/ftp/pub/PLIB_release_notes/Part25/Part25-IS/

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Annex D (normative) Standard data requirements for library integrated information model 25

Standard data are the entity instances that shall be recognized by any implementation compliant with ISO 13584 that claims conformance to some conformance class of some library integrated information model or view exchange protocol of ISO 13584.

Standard data must be specified by each library integrated information model and by each view exchange protocol. For each conformance class, each expanded listing in this annex incorporates all the elements that the corresponding short form schema, specified in Clause 8, uses from other schemas into a single schema without any external references to each of these schemas.

Standard data may include:

— instances of **basic_semantic_unit**s, associated with the corresponding **dictionary_element** and possibly with a **content_item**;

— instances of **external_file_protocol**s, and

— instances of other entities required to define the previous entity instances.

Recognition of a received **basic_semantic_unit** means that the corresponding **basic_semantic_unit** shall be already stored in the user library, together with a corresponding **dictionary_element** and possibly a **content_item** as specified in the view exchange protocol or library integrated information model standard data. This implies that a reference to a value-equal **basic_semantic_unit** in a supplier library is interpreted as a reference to the pre-existing **basic_semantic_unit**.

NOTE 1 Examples of **basic_semantic_unit**s that may be defined as standard data in a view exchange protocol include the **class_BSU** that identifies the functional view class which may be defined by the view exchange protocol and the **property_BSU** that identifies the view control variable of this functional view class.

Recognition of an external file protocol means that external files that reference an **external_file_protocol** that is value equal, shall be processed by an implementation that recognize this **external_file_protocol**.

NOTE 2 Example of an external file protocol that may be defined as standard data by a view exchange protocol or a library integrated information model is the ISO standard ISO 8859-1 that specifies a 8-bit single byte coded graphics character set for Latin alphabet N°1.

Standard data are specified by means of a set of constraints that shall be fulfilled by any library that claims conformance to some conformance class of LIIM 25. The following standard data are specified by library integrated information model 25.

D.1 Constraints on a library delivery file for referencing library integrated information model 25

This subclause defines **library_iim_identification** instance values that are allowed for use in a library delivery file to reference library integrated information model 25 defined in this part of ISO 13584.

The set of allowed values is defined by means of Table D.1 that specifies for each conformance class the allowed values of **library_iim_identification.name** and **library_iim_identification.application**, and by means of one EXPRESS schema that contains a global rule. This rule shall be fulfilled by any library delivery file that references library integrated information model 25, defined in this part of ISO 13584 in any of its conformance class. The goal of this rule is to specify the allowed values for the

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

other attributes of **library_iim_identification** that shall be used to reference library integrated information model 25, by means of relationships with **view_exchange_protocol_identification.name** and **view_exchange_protocol_identification.application**.

This rule is included in the **ISO13584_25_IEC61360_5_library_implicit_schema** specified in clause D.4.1.

D.2 Conformance class specification table

Table D.1 specifies the values of **library_iim_identification.name** and **library_iim_identification.application** that are allowed for use in a **library_iim_identification** to reference library integrated information model 25 in any of its conformance classes.

Table D. 1 — ISO 13584 LIIM 25 conformance class specification

Conformance Class	library_iim_identification.name mandatory value	library_iim_identification.application mandatory value
2	'ISO13584_25_IEC61360_5'	'2'
3	'ISO13584_25_IEC61360_5'	'3'
4	'ISO13584_25_IEC61360_5'	'4'
5	'ISO13584_25_IEC61360_5'	'5'
6	'ISO13584_25_IEC61360_5'	'6'
7	'ISO13584_25_IEC61360_5'	'7'
10	'ISO13584_25_IEC61360_5'	'10'
11	'ISO13584_25_IEC61360_5'	'11'

NOTE 1 Conformance class 1 is not explicitly specified because it does not reference the **library_iim_identification** entity data type. Consequently, no standard data will be specified for conformance class 1. Conformance classes 10 and 11 are explicitly specified because they may (optionally) reference the **library_iim_identification** entity data type

NOTE 2 The **allowed_reference_to_LIIM_25_rule**, **allowed_entity_instance_type_in_LIIM_25_rule** and **allowed_language_assignement_rule** rules apply to all the elements involved in the definitions of conformance classes 1 to 7 and 10 to 11. However, the **allowed_reference_to_LIIM_25_rule** has no effect on conformance classes 1 and may have no effect on conformance classes 10 and 11, since these latter may not involve **library_iim_identification** in their exchange context.

D.3 Standard data for conformance class 2 to 7 and 10 to 11 (all the conformance classes but conformance class 1)

This clause specifies the constraints on a library delivery file conform to the library integrated model LIIM 25.

The **library_iim_identification** instance values allowed for use in a library delivery file conform to the library integrated model LIIM 25 defined in this part of ISO 13584 in any conformance class 2 to 7 and 10 to 11 shall obey the constraints defined in the following EXPRESS schema.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

EXPRESS specification:

```
*)
SCHEMA ISO13584_25_IEC61360_5_conformance_schema;

USE FROM ISO13584_IEC61360_language_resource_schema(
      translated_label,
      present_translations,
      global_language_assignment);

USE FROM ISO13584_IEC61360_dictionary_aggregate_extension_schema(
      aggregate_type,
      entity_instance_type);

USE FROM ISO13584_extended_dictionary_schema(
      data_exchange_specification_identification,
      library_iim_identification);

USE FROM ISO13584_external_file_schema(
      external_file_protocol);
(*
```

NOTE The schema used above can be found in the following document:

ISO13584_IEC61360_language_resource_schema	IEC 61360-2:1998
(which is duplicated for convenience in Informative Annex D of ISO 13584-42:1998)	
ISO13584_IEC61360_dictionary_schema	IEC 61360-2:1998
(which is duplicated for convenience in Informative Annex D of ISO 13584-42:1998)	
ISO13584_extended_dictionary_schema	ISO 13584-24: :2003.
ISO13584_external_file_schema	ISO 13584-24: :2003.

D.3.1 Allowed_reference_to_LIIM_25_rule rule

The **allowed_reference_to_LIIM_25_rule** rule defines a formal constraint and an informal constraint on **library_iim_identification**s to be allowed for use to reference conformance classes 1 to 7 and 10 to 11 of library integrated model LIIM 25 defined in this part of ISO 13584. A **library_iim_identification** is allowed for use to reference conformance classes 2 to 7 and 10 to 11 of library integrated model LIIM 25 if the following conditions hold:

— the **name** attribute of the **library_iim_identification** that references library integrated model LIIM 25 shall be equal to 'ISO13584_25_IEC61360_5', and

— the **status** attribute of the **library_iim_identification** shall be equal to either, ‘WD’ or 'CD' or 'DIS' or 'FDIS' or 'IS' or 'TS' or 'PAS' or 'ITA', and

— the **application** attribute of the **library_iim_identification** shall have '2', '3', ‘4’, ‘5’, ‘6’ , '7', ‘10’ or '11'and as value the **external_file_protocol**s referenced by the **external_file_protocols** attribute of the **library_iim_identification** shall fulfill the constraints required by the **compliant_external_file_protocol_25** function.

Moreover, a **library_iim_identification** is allowed for use to reference conformance classes 2 to 7 and 10 to 11 of library integrated model LIIM 25 if one of the two following conditions hold concerning the **http_file**s that may be referenced directly or indirectly from the **library_iim_identification**:

— either each referenced **http_file** it is associated with a **mime** attribute and an **exchange_format** attribute corresponding to MIME type and subtype that correspond to a specification that is publicly available, or

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

— it is associated with a **mime** attribute and an **exchange_format** attribute corresponding to MIME type and subtype that correspond to a specification that is associated with public domain Internet-available readers.

Reference to **http_file**s corresponding to other MIME types and subtypes may only be done by private agreement between the sender and the receiver and are outside the scope of this International Standard. This is documented as an informal proposition **IP1** in **allowed_reference_to_LIIM_25_rule** rule.

EXPRESS specification:

```
*)
RULE allowed_reference_to_LIIM_25_rule FOR (
      library_iim_identification);
WHERE
      WR1: QUERY( liim_id <* library_iim_identification |
           ((liim_id\data_exchange_specification_identification.status
                = 'WD') OR
           (liim_id\data_exchange_specification_identification.status
                = 'CD') OR
           (liim_id\data_exchange_specification_identification.status
                = 'DIS') OR
           (liim_id\data_exchange_specification_identification.status
                = 'FDIS') OR
           (liim_id\data_exchange_specification_identification.status
                = 'IS') OR
           (liim_id\data_exchange_specification_identification.status
                = 'TS') OR
           (liim_id\data_exchange_specification_identification.status
                = 'PAS') OR
           (liim_id\data_exchange_specification_identification.status
                = 'ITA'))
           AND
           (liim_id\data_exchange_specification_identification.name
                = 'ISO13584_25_IEC61360_5')
           AND
           is_correct_liim_25_application_value(liim_id)
           AND
           (QUERY( efp <*
                liim_id\data_exchange_specification_identification
                .external_file_protocols
                | NOT(compliant_external_file_protocol_25([efp]))
                ) = []))
           = QUERY( liim_id <* library_iim_identification |
                (liim_id\data_exchange_specification_identification.name
                = 'ISO13584_25_IEC61360_5'));
END_RULE; -- allowed_reference_to_LIIM_25_rule
(*
```

Formal proposition:

WR1: when referencing library integrated model LIIM 25 defined in this part of ISO 13584, the **library_iim_identification.name** shall have ‘ISO13584_25_IEC61360_5’ as its value, **library_iim_identification.status** shall be equal to either 'WD', 'CD' or 'DIS' or 'FDIS' or 'IS' or ‘TS’ or

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

‘PAS’ or ‘ITA’, the **library_iim_identification.application** shall have ‘2’, '3', ‘4’, ‘5’, ‘6’, '7', ‘10’ or ‘11’ as its value, and the **library_iim_identification.external_file_protocols** shall fulfill the constraint specifications required by the **compliant_external_file_protocol_25** function defined below.

<u>Informal proposition:</u>

IP1: when it references library integrated model LIIM 25 defined in this part of ISO 13584 in one of the conformance class 2, 3, 4, 5, 6 or 7, a **library_iim_identification** may only reference, directly or indirectly, **http_file**s characterized by MIME types and subtypes that either correspond to specifications that are publicly available, or to specifications that are associated with public domain Internet-available readers.

D.3.2 Allowed_entity_instance_type_in_LIIM_25_rule rule

The **allowed_entity_instance_type_in_LIIM_25_rule** rule defines a formal constraint for the consistent **entity_instance_type** allowed data types.

For the purpose of the LIIM 25, solely the following **entity_instance_data_type** data types issued from the STEP resources are allowed

— the **entity_instance_type** data type which refer to an entity **representation**;

— the **entity_instance_type** data type which refer to an entity **representation_context**;

— the **entity_instance_type** data type which refer to an entity **geometric_representation_context**;

— the **entity_instance_type** data type which refer to an entity **representation_item**;

— the **entity_instance_type** data type which refer to an entity **date**;

— the **entity_instance_type** data type which refer to an entity **ordinal_date**;

— the **entity_instance_type** data type which refer to an entity **calendar_date**;

— the **entity_instance_type** data type which refer to an entity **local_time**;

— the **entity_instance_type** data type which refer to an entity **week_of_year_and_day_date**;

— the **entity_instance_type** data type which refer to an entity **date_and_time**;

— the **entity_instance_type** data type which refer to an entity **person**;

— the **entity_instance_type** data type which refer to an entity **organization**;

— the **entity_instance_type** data type which refer to an entity **address.**

For the purpose of the LIIM 25, solely the following entity_instance_data_type data types issued from the PLIB resources are allowed

— the **entity_instance_type** data type which refer to an entity **representation_reference**;

— the **entity_instance_type** data type which refer to an entity **program_reference**;

— the **entity_instance_type** data type which refer to an entity **property_value_external_item**;

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

EXPRESS specification:

```
*)
RULE allowed_entity_instance_type_in_LIIM_25_rule FOR (
     entity_instance_type);
WHERE
     WR1: QUERY( x<*entity_instance_type |
          NOT(
          ('REPRESENTATION_SCHEMA.REPRESENTATION'
               IN X.type_name)
               OR
          ('REPRESENTATION_SCHEMA.REPRESENTATION_CONTEXT'
               IN X.type_name)
               OR
          ('GEOMETRY_SCHEMA.GEOMETRIC_REPRESENTATION_CONTEXT'
               IN X.type_name)
               OR
          ('REPRESENTATION_SCHEMA.REPRESENTATION_ITEM'
               IN X.type_name)
               OR
          ('DATE_TIME_SCHEMA.DATE' IN X.type_name)
               OR
          ('DATE_TIME_SCHEMA.DATE_AND_TIME' IN X.type_name)
               OR
          ('DATE_TIME_SCHEMA.LOCAL_TIME' IN X.type_name)
               OR
          ('DATE_TIME_SCHEMA.CALENDAR_TIME' IN X.type_name)
               OR
          ('DATE_TIME_SCHEMA.ORDINAL_TIME' IN X.type_name)
               OR
          ('DATE_TIME_SCHEMA.WEEK_OF_YEAR_AND_DAY_TIME'
               IN X.type_name)
               OR
          (' PERSON_ORGANIZATION_SCHEMA.PERSON'
               IN X.type_name)
               OR
          (' PERSON_ORGANIZATION _SCHEMA.ORGANIZATION'
               IN X.type_name)
               OR
          (' PERSON_ORGANIZATION_SCHEMA.ADDRESS'
               IN X.type_name)
               OR
          ('ISO13584_EXTERNAL_FILE_SCHEMA.PROGRAM_REFERENCE'
               IN X.type_name)
               OR
          ('ISO13584_EXTERNAL_FILE_SCHEMA.REPRESENTATION_REFERENCE'
               IN X.type_name)
               OR
          ('ISO13584_EXTERNAL_FILE_SCHEMA.PROPERTY_VALUE_EXTERNAL_ITEM'
                IN X.type_name)
          )) = [];
END_RULE; -- allowed_entity_instance_type_in_LIIM_25_rule
(*
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Formal proposition:

WR1: when referencing library integrated model LIIM 25 defined in this part of ISO 13584, the **entity_instance_type.type_name** shall refer to **entity_instance_type** data types allowed in the exchange context defined by this part of ISO 13584.

D.3.3 Allowed_language_assignement_rule rule

The **allowed_language_assignement_rule** rule ensures that either an instance of **global_language_assignment** is available in the library delivery file or there exist one or several instances of **present_translation**s, but not both.

EXPRESS specification:

```
*)
RULE allowed_language_assignment_rule FOR (present_translations,
                                           global_language_assignment);
     WHERE
     WR1: (QUERY (x <*  global_language_assignment | TRUE)= [])
     XOR
          (QUERY(x<* present_translations | TRUE)=[]
          ) ;

END_RULE; -- Allowed_language_assignment
(*
```

Formal proposition:

WR1: no instance of **global_language_assignment** is available in the exchange file when there exist one or several instances of **present_translation**s

D.3.4 Compliant_http_protocol_25 function

The **compliant_http_protocol_25** function checks whether an **external_file_protocol** may be referenced as the HTTP protocol by a **library_iim_identification** that references library integrated model LIIM 25 in any of its conformance classes, or not. It returns TRUE if the given **external_file_protocol** is allowed for reference, otherwise, it returns FALSE. An **external_file_protocol** may be referenced as the HTTP protocol by a **library_iim_identification** that reference library integrated model LIIM 25 in any of its conformance classes if the following conditions hold:

— the **external_file_protocol** shall be a **http_protocol**, and

— the **organisation** attribute of the **external_file_protocol** shall reference an organization of which the **id** attribute equals to 'IAB' and the **name** attribute equals to 'Internet Architecture Board', and

— the **protocol_name** attribute of the **external_file_protocol** shall equal to 'HTTP' or to ‘HTTPS’, and

— the **designation** attribute of the **external_file_protocol** shall reference an **item_names** for which the **preferred_name** attribute equals to 'Hypertext Transfer Protocol' and the **short_name** attribute equals to 'RFC' followed by four digits and possibly some other characters.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

EXPRESS specification:

```
*)
FUNCTION compliant_http_protocol_25(ef : external_file_protocol):
BOOLEAN;

LOCAL
     ok: BOOLEAN := TRUE;
END_LOCAL;

IF (('ISO13584_EXTERNAL_FILE_SCHEMA'
     + '.HTTP_PROTOCOL' IN TYPEOF(ef)) AND
     (ef.organisation.id = 'IAB') AND
     (ef.organisation.name = 'Internet Architecture Board') AND
     ((ef.protocol_name = 'HTTP')
          OR (ef.protocol_name = 'HTTPS'))AND
     (ef.designation.preferred_name
     = 'Hypertext Transfer Protocol'))
THEN
     IF 'ISO13584_IEC61360_LANGUAGE_RESOURCE_SCHEMA.TRANSLATED_LABEL'
          IN TYPEOF(ef.designation.short_name)
     THEN
          REPEAT i:= 1 TO SIZEOF(ef.designation.short_name
                    \translated_label.labels);
               IF (ef.designation.short_name\translated_label.
                    labels[i] LIKE 'RFC####&')
               THEN
                    ok := ok AND TRUE;
               ELSE
                    ok := OK AND FALSE;
               END_IF;
          END_REPEAT;
          RETURN(OK);
     ELSE
          IF (ef.designation.short_name LIKE 'RFC####&')
          THEN
               RETURN(TRUE);
          ELSE
               RETURN(FALSE);
          END_IF;
     END_IF;
ELSE
     RETURN(FALSE);
END_IF;

END_FUNCTION; -- compliant_http_protocol_25
(*
```

D.3.5 Compliant_8859_1_protocol_25 function

The **compliant_8859_1_protocol_25** function checks whether an **external_file_protocol** may be referenced as the ISO 8859-1 protocol by a **library_iim_identification** that references library integrated model LIIM 25 in any of its conformance classes, or not. It returns TRUE if the given **external_file_protocol** is allowed for reference, otherwise, it returns FALSE. An **external_file_protocol** may be referenced as the ISO 8859-1 protocol by a **library_iim_identification**

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

that represents reference library integrated model LIIM 25 in any of its conformance classes, if the following conditions hold:

— the **external_file_protocol** shall be a **standard_data_protocol**, and

— the organisation attribute of the **external_file_protocol** shall reference an organization of which the **id** attribute equals to 'ISO' and the **name** attribute equals to 'International Organisation for Standardization', and

— the **protocol_name** attribute of the **external_file_protocol** shall equal to 'ISO_8859_1', and

— the **designation** attribute of the **external_file_protocol** shall reference an **item_names** for which the **preferred_name** attribute equals to 'Latin alphabet No 1' and the **short_name** attribute equals to 'ISO 8859-1'.

EXPRESS specification:

```
*)
FUNCTION compliant_8859_1_protocol_25(ef: external_file_protocol)
      : BOOLEAN;

IF (('ISO13584_EXTERNAL_FILE_SCHEMA'
      + '.STANDARD_DATA_PROTOCOL' IN TYPEOF(ef)) AND
      (ef.organisation.id = 'ISO') AND
      (ef.organisation.name
      = 'International Organisation for Standardization') AND
      (ef.protocol_name = 'ISO_8859_1') AND
      (ef.designation.preferred_name
      = 'Latin alphabet No 1') AND
      (ef.designation.short_name = 'ISO 8859-1'))
THEN
      RETURN(TRUE);
ELSE
      RETURN(FALSE);
END_IF;
END_FUNCTION; -- compliant_8859_1_protocol_25
(*
```

D.3.6 Compliant_external_file_protocol_25 function

The **compliant_external_file_protocol_25** function checks whether all the **external_file_protocol**s of a set of **external_file_protocol**s may be referenced as an library integrated model LIIM 25 by a **library_iim_identification** that references library integrated model LIIM 25 in one of its conformance class 1 to 4, or not. It returns TRUE if all the **external_file_protocol**s of a set of **external_file_protocol**s are allowed for reference, otherwise, it returns FALSE.

An **external_file_protocol** may be referenced by a **library_iim_identification** that represents conformance class 1 to 4 of library integrated model LIIM 25 if it may be referenced:

— either as the HTTP protocol, or

— as the ISO 8859-1 protocol.

NOTE In extended conformance classes of library integrated model LIIM 25, any other **external_file_protocol** may be referenced, subject to private agreement between the sender and the receiver.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

EXPRESS specification:

```
*)
FUNCTION compliant_external_file_protocol_25(
      s: SET [0:?] OF external_file_protocol): BOOLEAN;

REPEAT i := 1 TO SIZEOF(s);
      IF NOT (compliant_8859_1_protocol_25(s[i])
               OR compliant_http_protocol_25(s[i]))
      THEN
           RETURN(FALSE);
      END_IF;
END_REPEAT;

RETURN(TRUE);

END_FUNCTION; -- compliant_external_file_protocol_25
(*
```

D.3.7 Is_correct_liim_25_application_value function

The **is_correct_liim_25_application_value** function checks that the **liim_id library_iim_identification** is compatible with the conformance classes associated to the LIMM 25.

EXPRESS specification:

```
*)
FUNCTION is_correct_liim_25_application_value(
      liim_id: library_iim_identification): BOOLEAN;

IF EXISTS(liim_id\data_exchange_specification_identification.
      application)
      AND
      (((liim_id\data_exchange_specification_identification.
           application[1]='2')
           OR
      (liim_id\data_exchange_specification_identification.
           application[1]='3')
           OR
      (liim_id\data_exchange_specification_identification.
           application[1]='4')
           OR
      (liim_id\data_exchange_specification_identification.
           application[1]='5')
           OR
      (liim_id\data_exchange_specification_identification.
           application[1]='6')
           OR
      (liim_id\data_exchange_specification_identification.
           application[1]='7'))
      AND
      (liim_id\data_exchange_specification_identification.
           Application LIKE '#'))
      OR
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
          ((liim_id\data_exchange_specification_identification.
               application[1]='1')
               AND
          ((liim_id\data_exchange_specification_identification.
               application[2]='0')
          OR
          (liim_id\data_exchange_specification_identification.
               application[2]='1')))

     THEN
          RETURN(TRUE);
     ELSE
          RETURN(FALSE);
     END_IF;
     END_FUNCTION; -- is_correct_liim_25_application_value
     (*
```

D.4 Additional constraint for conformance classes 4 and 7

This clause specifies the additional constraint on a library delivery file conform to conformance classes 4 and 7 associated to the library integrated model LIIM 25.

D.4.1. nesting_level_aggregate_limit_rule rule

The **nesting_level_aggregate_limit_rule** rule checks that the level of nested elements in the aggregate values is limited to 2.

This rule is applied for each instance of the entity **aggregate_type**. The Boolean function **no_more_than_two_nested_levels** is used as a filter for each instance of the **aggregate_type** entity.

EXPRESS specification:

```
     *)
     RULE nesting_level_aggregate_limit_rule FOR
                    (library_iim_identification,
                        aggregate_type);
          WHERE
          WR1: NOT (QUERY( liim_id <* library_iim_identification |
                    (liim_id\data_exchange_specification_identification.name
                    = 'ISO13584_25_IEC61360_5')
                    AND
                    ((liim_id\data_exchange_specification_identification.
                    application[1]='4')
                    OR
                    (liim_id\data_exchange_specification_identification.
                    application[1]='7'))) <> [])
               OR
                    (QUERY (x <*  aggregate_type | NOT
                    no_more_than_two_nested_levels(x))= []);
     END_RULE; -- nesting_level_aggregate_limit_rule
      (*
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

D.4.2. no_more_than_two_nested_levels function

The **no_more_than_two_nested_levels** function checks that an aggregate value does not contain more than two levels of aggregate values.

EXPRESS specification:

```
*)
FUNCTION no_more_than_two_nested_levels(typ : aggregate_type):BOOLEAN;


IF NOT ('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.'+
          'ENTITY_INSTANCE_TYPE_FOR_AGGREGATE' IN
          TYPEOF(typ.value_type))
THEN -- level 1 is not an aggregate
     RETURN (TRUE);
END_IF;

-- level 1 is an aggregate

IF NOT ('ISO13584_IEC61360_DICTIONARY_AGGREGATE_EXTENSION_SCHEMA.'+
          'ENTITY_INSTANCE_TYPE_FOR_AGGREGATE' IN
          TYPEOF(typ.value_type.type_structure.value_type))
THEN -- level 2 is not an aggregate
     RETURN (TRUE);
END_IF;

-- Level 2 is an aggregate
RETURN(FALSE);

END_FUNCTION; -- more_than_two_nested_levels
(*

*)
END_SCHEMA; --ISO13584_25_IEC61360_5_conformance_schema
(*
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Annex E (normative) Implementation method specific requirements for the library integrated information model 25

Conformance to the library integrated information model 25 shall be realised in one or more implementation methods. The implementation methods define what types of exchange behaviour is required with respect to exchange protocols.

One implementation method is defined for the library delivery file: ISO 10303-21.

The implementation methods for the possible external files referenced from the library delivery file and whose **external_file_protocol** belong to the standard data of the library integrated information model 25 are defined by the standard referenced in this **external_file_protocol**, possibly further specified as part of the description of the library integrated information model standard data (see Annex B).

For the exchange structure, the file format of the library delivery file shall be encoded according to the syntax and EXPRESS language mapping defined in ISO 10303-21 for the schema defined in Annex A of this part of ISO 13584. The header of the exchange structure shall identify use of this part of ISO 13584 by the schema names 'ISO13584_25_IEC61360_5_library_implicit_schema'.

NOTE Identification of the library delivery file is done by separate agreement between the sender and the receiver and is outside the scope of this part of ISO 13584.

© ISO 2004 All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

Annex F (informative)
EXPRESS-G diagrams

Figure F.1 through F.2 correspond to the EXPRESS schemas given in chapters 6 and 7. The diagrams use the EXPRESS-G graphical notation for the EXPRESS language. EXPRESS-G is defined in Annex A of ISO 10303-11.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

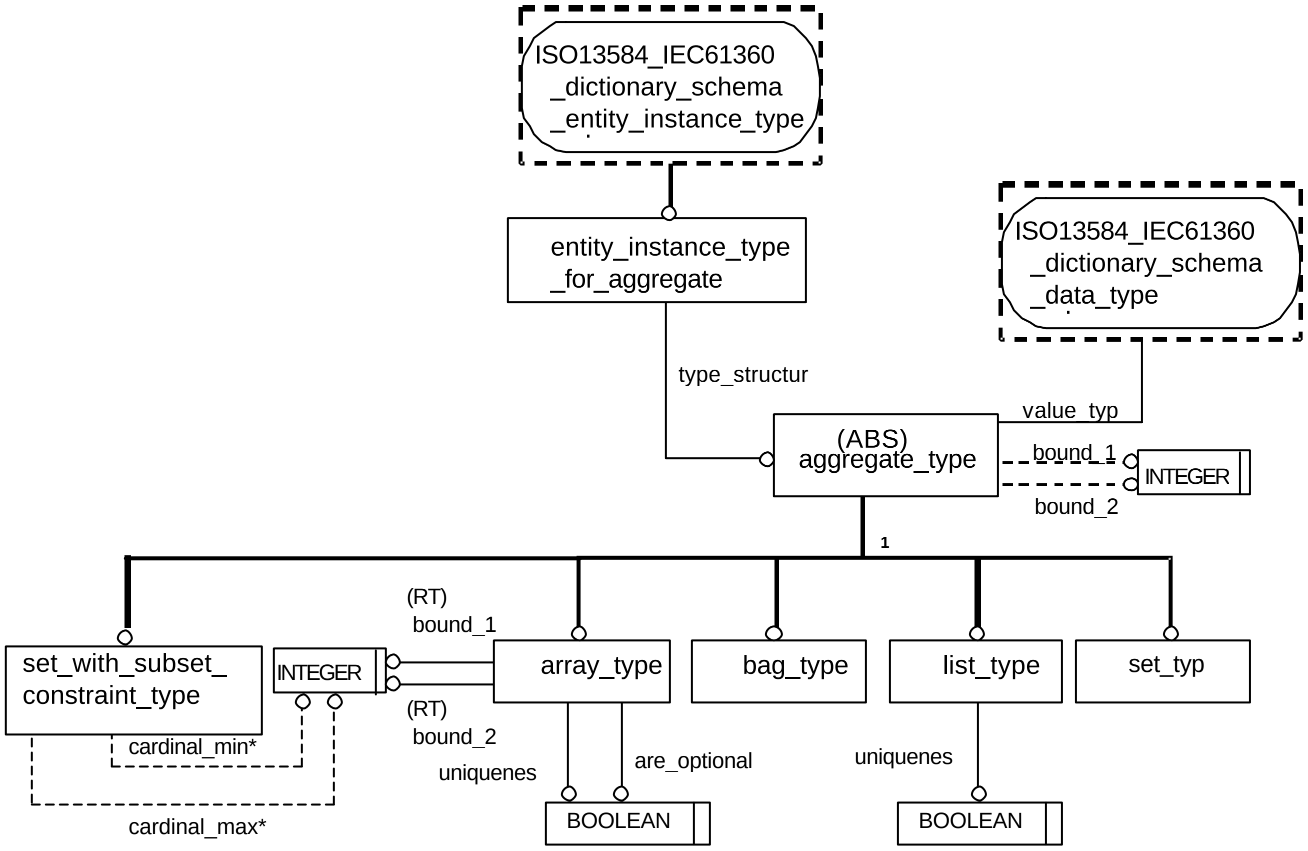

Figure F.1 — ISO13584_IEC61630_dictionary_aggregate_extension_schema diagram 1 of 1

© ISO 2004 All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

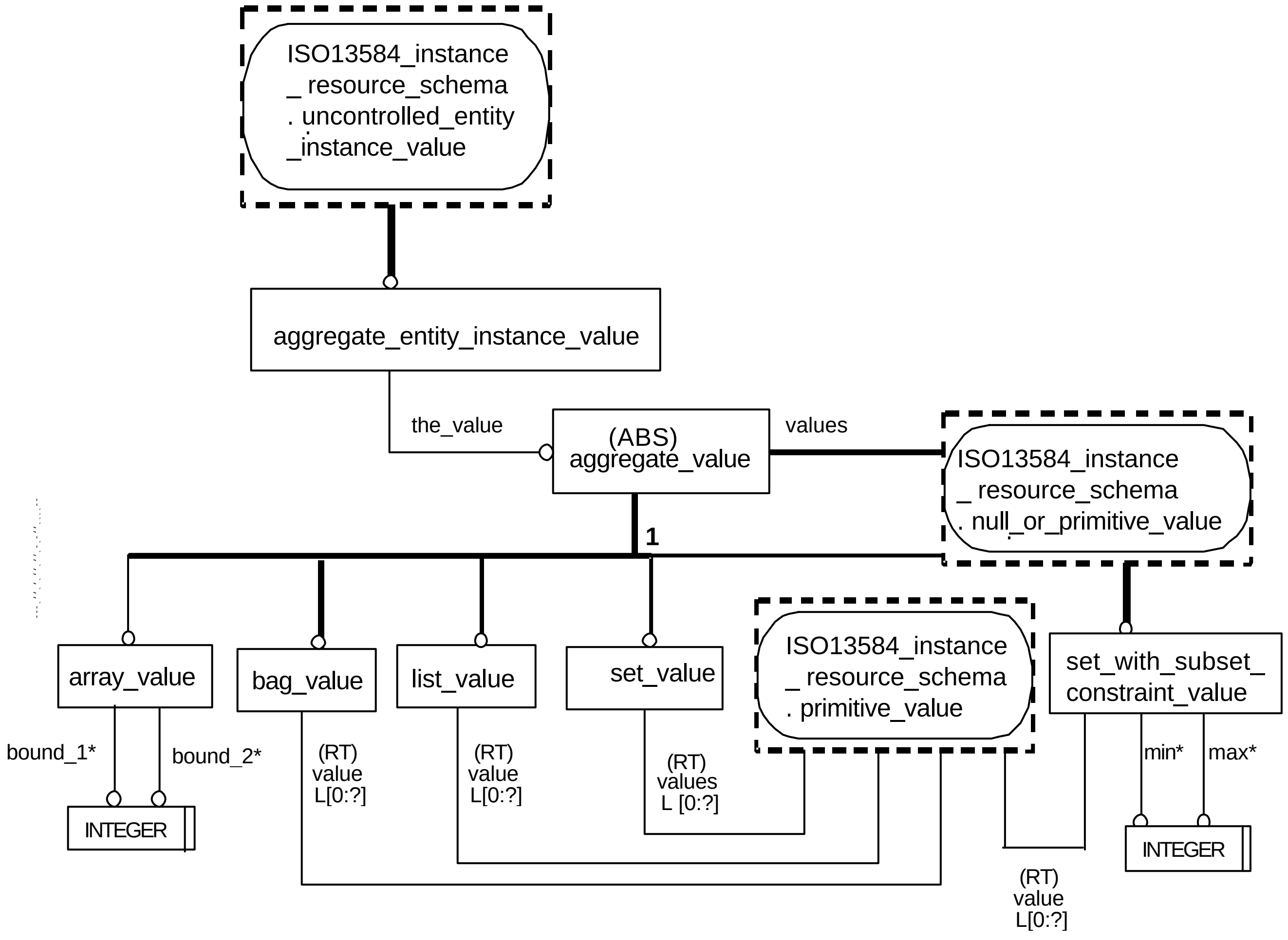
ISO13584_instance _ resource_schema . uncontrolled_entity _instance_value
aggregate_entity_instance_value
the_value
(ABS) aggregate_value
values
ISO13584_instance _ resource_schema . null_or_primitive_value
1
array_value
bag_value
list_value
set_value
ISO13584_instance _ resource_schema . primitive_value
set_with_subset_ constraint_value
bound_1*
bound_2*
INTEGER
(RT) value L[0:?]
(RT) value L[0:?]
(RT) values L [0:?]
(RT) value L[0:?]
min*
max*
INTEGER

Figure F.2 — ISO13584_aggregate_value_schema diagram 1 of 1

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Annex G (informative) Commented example of library integrated information model 25 physical files. Exchange of explicit general models

This annex presents, on an example, an overview of the different resources involved in the description of a parts library using an explicit modeling according to this part of ISO 13584.

G.1 Capturing a parts family in ISO 13584

Figure G.1 illustrates the part family intended to be described. It is a family of washers, denoted PAW, that is sold by a bearing supplier and that is used as a bearing in some mechanical contexts.

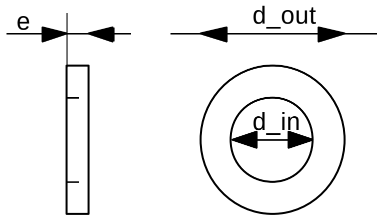

Figure G.1 — PAW family description

Such a part family may be described at two levels of abstraction:

a) The dictionary level (**ISO13584_IEC61360_dictionary_schema** and **ISO13584_extended_dictionary_schema**) permits the description of the concepts of a part family i.e., the supplier(s), the class(es), the property(ies,) … It defines what are the meaning and the value types of the properties, in which classes the properties are visible, by which supplier the classes are specified... Such a data **dictionary_element** may be done both for the general model description (what is this family of parts) and for the functional model description (what kind of representations may be defined for this family of parts).

b) The library level (**ISO13584_library_content_schema**) permits the definition of the values of the properties to explicitly define the allowed instances of the parts family. This level is for instance useful when describing an extension of a library.

These two levels of abstraction have to be represented in two different ways.

Figure G.2 presents an example of instance of a family that is only defined at the dictionary level. The permitted instances are those where the values of the properties describing the part belong to the type defined in the data dictionary.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

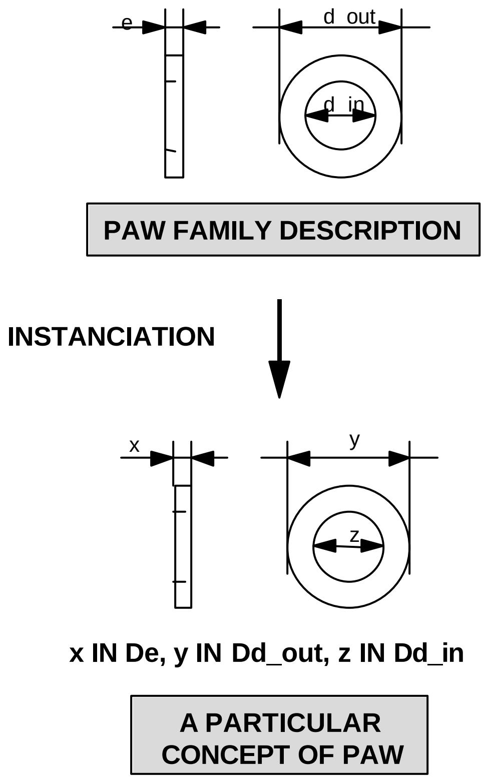

Figure G.2 — Instance of a dictionary description

Figure G.3 describes and presents an example of a family that is associated with a two-fold description: **dictionary_element** and library specification. The allowed instances are those where the values of the properties describing the part belong to the (explicit) list of authorized tuples of values defined in the library specification of a part.

d_in	e	d_out
10	1	15
11	1	16.5
13	2	19.5
17	3	25.5
19	4	28.5

Figure G.3 — Explicit description of a dictionary description

The example presented in this annex involves this kind of two-fold description because this part family is defined originally in a paper catalogue. Therefore only a well-defined set of instances are really provided for the bearing.

G.2 Description of the PAW parts family

This description is two-fold. First the concept of PAW and of its properties are defined. Second, the allowed instances are precisely defined.

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

G.2.1 Dictionary description: the BSU mechanism

The description of a data dictionary according to the **ISO13584_IEC61360_dictionary_schema** schema requires the specification of what are the identifiers of the different concepts (called basic semantic unit: BSU) involved in the parts family definition. These identifiers define unambiguously and universally each concept within an ISO 13584-compliant data dictionary.

The following example (Figure G.4) outlines the resources used for the specification of these identifiers:

```
/* BSU for supplier */
/* The code of the supplier must be defined according to ISO13584-26:
Supplier identification. Here, the code doesn't follow the ISO 13584-26
requirements, because the supplier code is not known at the moment */
#20 = SUPPLIER_BSU('INA', *);

/* BSU for component_class */
/* The class BSUs defines the identification of the various classes,
and who is the supplier that is responsible of the class definitions */
#50 = CLASS_BSU('BEARING', '001', #20);
#60 = CLASS_BSU('PAW', '001', #20);

/* BSU for properties */
/* The property BSU defines the identifications of the properties and
the class where these properties are visible */
#90 = PROPERTY_BSU('d_in', '001', #50);
#100 = PROPERTY_BSU('d_out', '001', #50);
#110 = PROPERTY_BSU('e', '001', #60);
```

Figure G.4 — Identifiers of the concepts involved in the PAW family

G.2.2 Dictionary description: the dictionary element definition

A BSU only identifies a concept. A **dictionary_element** provides a computer-sensible and human-readable definition of the concept. This relationship between these two levels is presented in Figure G.5.

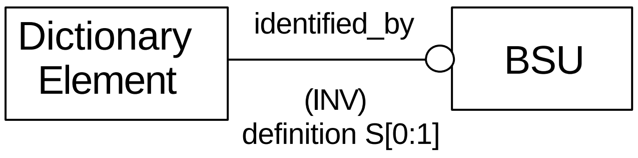

Figure G.5 — The BSU / Dictionary element relationship

The following figure (Figure G.6) outlines the main structure of the **dictionary_element**s corresponding to the previous basic semantic units identifiers.

```
/* Dictionary properties description */
/* supplier description */
#21 = SUPPLIER_ELEMENT(#20, $, '001', #22, #23);
#22 = ORGANIZATION($, 'INA', '');
#23 = ADDRESS($, $, $, $, $, $, $, 'GERMANY', $, $, $, $);

/* d_in */
#91 = NON_DEPENDENT_P_DET(#90, $, '001',
      #92,     /* Item names associated to the non dependent P_DET */
```

© ISO 2004 All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
      TEXT('inner diameter'), $, $, $, $, (), $, 'TO3',
      #93, /* the specific data type of the property (not represented:measure
      in mm) */
      $);
#92 = ITEM_NAMES(LABEL('inner diameter'), (), LABEL(''), $, $);
#93 = REAL_MEASURE_TYPE('NR2..3.3', #94);
#94 = DIC_UNIT(#95, $);
#95 = SI_UNIT(*, .MILLI., .METRE.);
/* d_out */
#101 = NON_DEPENDENT_P_DET(#100, $, '001', #102, TEXT('outer
      diameter'), $, $, $, $, (), $, 'TO3', #93, $);

/* e */
#111 = NON_DEPENDENT_P_DET (#110, $, '001', #112, TEXT('thickness'),
      $, $, $, $, (), $, 'TO3', #93, $);

/* Dictionary class description */
/* Part class */
#71 = COMPONENT_CLASS (#50,  /* reference to its BSU */
       $, '001',
      #72, /* item_names */
      TEXT('Class associated to the generic bearing family'),
           /* Definition */
      $, $, $, $,
      (#90, #100),  /* the list of the properties that may be used
                    to describe an instance of this class
                    (applicability of the properties) */
      (), $, (),(), $);
#72 = ITEM_NAMES(LABEL('Generic bearing family'), (),
      LABEL('Bearing family'), $, $);

/* PAW class */
#81 = COMPONENT_CLASS(#60, $, '001', #82,
      TEXT('Class associated to the PAW part family'),
      $, $, $, #50, (#110), (), $, (),(), $);
```

Figure G.6 — Dictionary_element of the concepts involved in the PAW family

G.2.3 Library specification: description of the class extension

The extension of a class is given by the set of instances of a class. Each instance of a class contains a set of property values that correspond to the values of the properties belonging to the part family described by this instance. Figure G.7 describes the extension of a class by an explicit modeling.

```
/* Dictionary extension */
/* Extension of a class */
#8000= EXPLICIT_ITEM_CLASS_EXTENSION(#60,   /*Reference to the BSU */
      (),(),(),'001','001',(),(),(#90),
      (#8100,#8200,#8300,#8400,#8500),/* the extension of class given
                                       by a list of class instances.
      .T.,$,$,(),$,(),());
```

Figure G.7 — Dictionary_element of the concepts involved in the PAW family

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

The class instance is described by the list of its property values. Figure G.8 gives the descrition, by extension of one instance corresponding to the part family for which **d_in** equals 10, **e** equals 1 and **d_out** equals 15.

```
/* Extension of a library component */
#8100=LIB_COMPONENT_INSTANCE(#60, /* the BSU associated to the class
      which the current instance is an instance of */
      (#8101, #8102, #8103), /* the property values */
      (), $, $, $, $, .T., $);

/* Property values of the extension of a class */
#8101=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#8102=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#8103=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
```

Figure G.8 — Dictionary_element of the concepts involved in the PAW family

G.3 A complete physical file for explicit general models.

In this clause, the complete example of a physical file is provided. The following physical file, compliant with the library integrated information 25 defines the PAW family.

```
ISO-10303-21;
HEADER;

FILE_DESCRIPTION (('THIS IS AN EXAMPLE OF AN EXPLICIT GENERAL MODEL'),
'2');
FILE_NAME('P25_gm_explicit.p21',
          '2000-11-28T17:38:14',
          (''),
          ('LISI/ENSMA'),
          'ECCO RUNTIME SYSTEM BUILT-IN PREPROCESSOR V2.3beta1',
          'ECCO RUNTIME SYSTEM V2.3beta1',
          '');
FILE_SCHEMA (('ISO13584_25_IEC61360_5_LIBRARY_IMPLICIT_SCHEMA'));
ENDSEC;

DATA;

/* Global library description */
#2 = LIBRARY_IN_STANDARD_FORMAT ($, $, $, $, (), #20, #11, (), (),
(#20), (#50, #60), (), #3, $, $, ());
#3 = ITEM_NAMES (LABEL('Explicit general model example'), (),
LABEL(''), $, $);
#10 = GLOBAL_LANGUAGE_ASSIGNMENT ('en');
#11 = LIBRARY_IIM_IDENTIFICATION ($, 'IS', 'ISO13584_25_IEC61360_5',
2003, '5', $, ());

/* DICTIONARY DESCRIPTION */
/*BSU for supplier */
#20 = SUPPLIER_BSU ('INA', *);

/* BSU for component_class */
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#50 = CLASS_BSU ('BEARING', '001', #20);
#60 = CLASS_BSU ('PAW', '001', #20);

/* BSU for properties */
#90 = PROPERTY_BSU ('d_in', '001', #50);
#100 = PROPERTY_BSU ('d_out', '001', #50);
#110 = PROPERTY_BSU ('e', '001', #60);

/* Dictionary properties description */
/* supplier description */
#21 = SUPPLIER_ELEMENT (#20, $, '001', #22, #23);
#22 = ORGANIZATION ($, 'INA', '');
#23 = ADDRESS ($, $, $, $, $, $, $, 'GERMANY', $, $, $, $);

/* d_in */
#91 = NON_DEPENDENT_P_DET (#90, $, '001', #92, TEXT('inner diameter'),
$, $,
$, $, (), $, 'TO3', #93, $);
#92 = ITEM_NAMES (LABEL('inner diameter'), (), LABEL(''), $, $);
#93 = REAL_MEASURE_TYPE ('NR2..3.3', #94);
#94 = DIC_UNIT (#95, $);
#95 = SI_UNIT (*, .MILLI., .METRE.);
/* d_out */
#101 = NON_DEPENDENT_P_DET (#100, $, '001', #102, TEXT('outer
diameter'), $,
$, $, $, (), $, 'TO3', #93, $);
#102 = ITEM_NAMES (LABEL('outer diameter'), (), LABEL(''), $, $);
#103 = REAL_MEASURE_TYPE ('NR2..3.3', #104);
#104 = DIC_UNIT (#105, $);
#105 = SI_UNIT (*, .MILLI., .METRE.);

/* e */
#111 = NON_DEPENDENT_P_DET (#110, $, '001', #112, TEXT('thickness'), $,
$, $,
$, (), $, 'TO3', #93, $);
#112 = ITEM_NAMES (LABEL('thickness'), (), LABEL(''), $, $);
#113 = REAL_MEASURE_TYPE ('NR2..3.3', #114);
#114 = DIC_UNIT (#115, $);
#115 = SI_UNIT (*, .MILLI., .METRE.);

/* Dictionary class description */
/* Part class */
#71 = COMPONENT_CLASS (#50, $, '001', #72, TEXT('Class associated to
the
generic bearing family'), $, $, $, $, (#90, #100), (), $, (),(), $);
#72 = ITEM_NAMES (LABEL('Generic bearing family'), (), LABEL('Bearing
family'), $,
$);

/* PAW class */
#81 = COMPONENT_CLASS (#60, $, '001', #82, TEXT('Class associated to
the PAW
part family'), $, $, $, #50, (#110), (), $, (),(), $);
#82 = ITEM_NAMES (LABEL('PAW family'), (), LABEL('PAW'), $, $);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
/* Library extension */
/* Extension of a class */
#8000=
EXPLICIT_ITEM_CLASS_EXTENSION(#60,(),(),(),'001','001',(),(),(#90),
(#8100,#8200,#8300,#8400,#8500),.T.,$,$,(),$,(),());

/* Extension of a library component */
#8100=LIB_COMPONENT_INSTANCE(#60, (#8101, #8102, #8103), (), $, $, $,
$, .T., $);
/* Property values of the extension of a class */
#8101=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#8102=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#8103=PROPERTY_VALUE(REAL_VALUE(15.0), #110);

/* Extension of a library component */
#8200=LIB_COMPONENT_INSTANCE(#60, (#8201, #8202, #8203), (), $, $, $,
$, .T., $);
/* Property values of the extension of a class */
#8201=PROPERTY_VALUE(REAL_VALUE(11.0), #90);
#8202=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#8203=PROPERTY_VALUE(REAL_VALUE(16.5), #110);

/* Extension of a library component */
#8300=LIB_COMPONENT_INSTANCE(#60, (#8301, #8302, #8303), (), $, $, $,
$, .T., $);

/* Property values of the extension of a class */
#8301=PROPERTY_VALUE(REAL_VALUE(13.0), #90);
#8302=PROPERTY_VALUE(REAL_VALUE(2.0), #100);
#8303=PROPERTY_VALUE(REAL_VALUE(19.5), #110);

/* Extension of a library component */
#8400=LIB_COMPONENT_INSTANCE(#60, (#8401, #8402, #8403), (), $, $, $,
$, .T., $);

/* Property values of the extension of a class */
#8401=PROPERTY_VALUE(REAL_VALUE(17.0), #90);
#8402=PROPERTY_VALUE(REAL_VALUE(3.0), #100);
#8403=PROPERTY_VALUE(REAL_VALUE(25.5), #110);

/* Extension of a library component */
#8500=LIB_COMPONENT_INSTANCE(#60, (#8501, #8502, #8503), (), $, $, $,
$, .T., $);

/* Property values of the extension of a class */
#8501=PROPERTY_VALUE(REAL_VALUE(19.0), #90);
#8502=PROPERTY_VALUE(REAL_VALUE(4.0), #100);
#8503=PROPERTY_VALUE(REAL_VALUE(28.5), #110);

ENDSEC;
END-ISO-10303-21;
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

(Blank page)

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

Annex H (informative)
Commented example of library integrated information model 25 physical files.
Exchange of explicit functional models compliant with ISO 13584-101

This annex presents, on an example, an overview of the different resources involved in the description of a parts library using an explicit modeling according to this part of ISO 13584. It describes a physical file example allowing the exchange of explicit functional models compliant with ISO 13584:101 view exchange protocol.

H.1 Description of the PAW parts family and its geometry

The description of the Paw family is identical to the one described and commented in the previous Annex G except that supplier elements have been introduced in order to describe the view and the geometry suppliers.

H.1.1 Dictionary description

The two **supplier_BSU**s LISI/ENSMA and ISO identify respectively the suppliers of the functional model and of the functional view

```
/*BSU for supplier */
#20=SUPPLIER_BSU('94/1124946367', *);  /* parts manufacturer code
following ISO 13584-26*/
#30=SUPPLIER_BSU('9/19860073600021', *);
/* LISI/ENSMA code in the coding scheme ICD=0009 : SIRET number */
#40=SUPPLIER_BSU('0112/1///13584_101_1', *);
/* Identification of ISO 13584-101 according to ISO 13584-26 */
/* supplier of the functional view */

/* BSU for component_class */
/*The two class_BSUs PAW_Gemetry and basic_geometry identify
respectively the classes corresponding to the functional model and of
the functional view. */
#50=CLASS_BSU('Bearing', '001', #20);
#60=CLASS_BSU('PAW', '001', #20);
#130=CLASS_BSU('PAW_Geometry', '001', #30);
#140=CLASS_BSU('basic_geometry', '001', #40);

/* BSU for properties */
/* The follwing property_BSU describe the properties of the part family
PAW as defined in Annex G.*/
#90=PROPERTY_BSU('d_in', '001', #50);
#100=PROPERTY_BSU('d_out', '001', #50);
#110=PROPERTY_BSU('e', '001', #50);

/* The definition of the geometry for a given part, and particularly
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
for the part family PAW requires the definition of representation
properties. These properties are defined as for part family properties
through the BSU mechanism. However, this BSU identifies a
representation_P_DET element.*/
#150=PROPERTY_BSU('geometry_level', '001', #140);
#160=PROPERTY_BSU('detail_level', '001', #140);
#170=PROPERTY_BSU('side', '001', #140);
#180=PROPERTY_BSU('prg', '001', #130);
#200=PROPERTY_BSU('variant', '001', #140);
#210=PROPERTY_BSU('unreg_variant', '001', #140);

/* supplier description */
/* The following supplier_element describes LISI/ENSMA as supplier. It
will be used as supplier of a functional model class. */
#31=SUPPLIER_ELEMENT(#30, $, '001', #32, #33);
#32=ORGANIZATION('LISI/ENSMA', 'LISI/ENSMA', '');
#33=ADDRESS($, $, $, $, $, $, $, 'FRANCE', $, $, $, $);

/* Dictionary properties description */
/* prg */
/* Data type elements associated to represetation properties are
representation_P_DETs. As example; the following data element describes
the variable allowing to refer programs (prg). Its values will be
described below. */
#91=REPRESENTATION_P_DET (#180, $, '001', #92, TEXT('variable used to
reference geometry programs'), $, $, $, $, (), $, 'A58', #93, $);
#92=ITEM_NAMES (LABEL('related program'), (), LABEL(''), $, $);
#93=PROGRAM_REFERENCE_TYPE
(('ISO13584_25_IEC61360_5_LIBRARY_IMPLICIT_SCHEMA.PROGRAM_REFERENCE'));
```

Figure H.1 — The Identifiers of the concepts involved in the Paw family and its geometry representation

H.2. Description of geometric representations for the PAW parts family

We assume now that some library data supplier wants to provide a geometric representation for all the instances of the PAW family that are described in an explicit manner (by extension). This requires the description of a functional model class.

A functional model class is intended to represent different perspectives of the different parts described in the general model class. A functional model class has to be described like a general model class, i.e., through a class definition and through a dictionary extension (library specification).

A **functional_model_class** describes a particular view (is-view-of relationship) of a given parts family (described as a general model class), according to the point of view specified by a **functional_view_class**.

In the example, it will be defined a functional model class representing some kind of geometry (specified by) a **functional_view_class** for the PAW parts family.

A **fm_class_view_of** is a functional model class that refers to a well defined general model class (here the class that models the PAW family) and that provides a particular kind of representation (specified by a **functional_view_class**) for this general model class.

A functional model class is not required to provide representation for all the values of the functional view

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

class view control variables. The range of supported values is specified by **view_control_variable_range** as shown in **Fehler! Verweisquelle konnte nicht gefunden werden.**.

```
/* v_c_v range */
#155=VIEW_CONTROL_VARIABLE_RANGE(#150, 1, 1);
#165=VIEW_CONTROL_VARIABLE_RANGE(#160, 2, 2);
#175=VIEW_CONTROL_VARIABLE_RANGE(#170, 1, 6);
#205=VIEW_CONTROL_VARIABLE_RANGE (#200, 1, 1);
#215=VIEW_CONTROL_VARIABLE_RANGE (#210, 0, 0);
```

Figure H.2 — View control variables range definition

In our example, the functional model class only provides 2D views (range 1..1, for #150 that is ‘geometry_level’), with standard representation (range 2..2, for #160 that is ‘detail-level’) for all the sides from ‘front’ to ‘bottom’ (range 1..6, for # 170 that is ‘side’).

Moreover, to be able to create the geometry, the **fm_class_view_of** needs to import some properties of the PAW family. The **dictionary_element** of the **fm_class_view_of** is presented in the **Fehler! Verweisquelle konnte nicht gefunden werden.**.

```
/* Dictionary class description */
/* Functional model class view_of definition*/
/* The following instance describes the is_view_of relationship through
a fm_class_view_of class supplied by LISI/ENSMA.
#71=FM_CLASS_VIEW_OF(#130,   /* reference to BSU */
      $, '001', #72, /* item names */
      TEXT('Explicit functional model class describing the 2d standard
      geometry of PAW'), $, $, $, $,
      (#180), /* BSU of the ‘prg’and of the required_side
                properties */
      (), *, *, *, *, *,
      #140, /* the created view (reference to the BSU of the
                functional_view_class */
      (#155, #165, #175, #205, #215), /* the vcv ranges */
      (#150, #160, #170, #200, #210), /* vcv’s imported from the
                functional view class */
      (), (), (), (), (), (), (), (),
      #60, /* is_view_of relationship. Reference to the
                functional model */
      (#90, #100, #110), ),/* imported properties from the
                general model */
      (),(),());

#72=ITEM_NAMES(LABEL('Functional model class of PAW'), (),
      LABEL('fm class of PAW'), $, $);
```

Figure H.3 — Specification of the view created by a functional model class

H.3. Library specification of the functional model class

An **explicit_functional_model_class_extension** is the **content_item** that constitutes the library specification a **functional_model_class** and **fm_class_view_of** subtype. It gives the set of the

© ISO 2004 All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

properties that need to be valued in the context of an instance of such a class

In the case of an explicit representation, the instances of a functional model are explity enumerated. In the following **explict_functional_model_extension** instance, the instances of a functional model are described by **lib_f_model_instance**s in the range 3000 to 3450.

```
/* LIBRARY DESCRIPTION */
/* Description of the extension of a functional model class */
/* It references all the library functional model instances described
themselves by extesnion.
In the case of an explicit representation, the instances of a
functional model are explity enumerated. In the following
explict_functional_model_extension instance, the instances of a
functional model are described by lib_f_model_instances in the range
3000 to 3450. */
#1300=EXPLICIT_FUNCTIONAL_MODEL_CLASS_EXTENSION(#130,
      (#2501, #2502, #2503, #2504, #2505, #2506), /* the set of
           program references. The programs which allow to display
           geometry.*/
      (#7), (#12), '001', '001', (), (),
      (#90, #100, #110, #170), /* properties needed to display the
           geometry */
      (#3000, #3010, #3020, #3030, #3040, #3050,
      #3100, #3110, #3120, #3130, #3140, #3150,
      #3200, #3210, #3220, #3230, #3240, #3250,
      #3300, #3310, #3320, #3330, #3340, #3350,
      #3400, #3410, #3420, #3430, #3440, #3450), /* The extension of
           all the instances of a functional model. They are given by
           the lib_f_model_instances */
      .T., $, (#90, #100, #110), #180, $, $ ,(), $);
```

Figure H.4 — Description by extension of the instances of a functional model

```
/* Reference to programs which display geometry */
/* According to the previous instance, references to programs that
describe geometry representation are performed. These
program_references refer themselves to external files PAW_p1.for ..
PAW_p6.for containing geometry programs written in the FORTRAN language
according to view exchange protocol ISO 13584-101. */

#2501=PROGRAM_REFERENCE(#7, #2601, 'Add1_PAW', 'PAW_p1',
      (#90, #100, #110), (), ());
#2502=PROGRAM_REFERENCE(#7, #2602, 'Add2_PAW', 'PAW_p2',
      (#90, #100, #110), (), ());
#2503=PROGRAM_REFERENCE(#7, #2603, 'Add3_PAW', 'PAW_p3',
      (#90, #100, #110), (), ());
#2504=PROGRAM_REFERENCE(#7, #2604, 'Add4_PAW', 'PAW_p4',
      (#90, #100, #110), (), ());
#2505=PROGRAM_REFERENCE(#7, #2605, 'Add5_PAW', 'PAW_p5',
      (#90, #100, #110), (), ());
#2506=PROGRAM_REFERENCE(#7, #2606, 'Add6_PAW', 'PAW_p6',
      (#90, #100, #110), (), ());

#2601=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2701));
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#2602=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2702));
#2603=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2703));
#2604=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2704));
#2605=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2705));
#2606=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2706));
#2701=LANGUAGE_SPECIFIC_CONTENT((#2801), #2801, $);
#2702=LANGUAGE_SPECIFIC_CONTENT((#2802), #2802, $);
#2703=LANGUAGE_SPECIFIC_CONTENT((#2803), #2803, $);
#2704=LANGUAGE_SPECIFIC_CONTENT((#2804), #2804, $);
#2705=LANGUAGE_SPECIFIC_CONTENT((#2805), #2805, $);
#2706=LANGUAGE_SPECIFIC_CONTENT((#2806), #2806, $);

/* Descriprion of the source files for geometry */
#2801=EXTERNAL_FILE_UNIT('PAW_p1.for', '7bit');
#2802=EXTERNAL_FILE_UNIT('PAW_p2.for', '7bit');
#2803=EXTERNAL_FILE_UNIT('PAW_p3.for', '7bit');
#2804=EXTERNAL_FILE_UNIT('PAW_p4.for', '7bit');
#2805=EXTERNAL_FILE_UNIT('PAW_p5.for', '7bit');
#2806=EXTERNAL_FILE_UNIT('PAW_p6.for', '7bit');
```

Figure H.5 — References to FORTRAN programs that display geometry.

Instances of functional models described by extension, describe themselves their property values by extension.

Fehler! Verweisquelle konnte nicht gefunden werden. shows one instance of functional model which describe by extension the values of its properties.

```
/* Descriprion of library functional model instance by extension */

#3000=LIB_F_MODEL_INSTANCE(#130, (#3001, #3008, #3009, #3002, #3003,
      #3004, #3005, #3006, #3007), ());

/* property values that describe the previous library functional
model extension */

#3001=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#3008=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3009=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
#3002=PROPERTY_VALUE(1PROPERTY_VALUE(INTEGER_VALUE(1), #170);
#3003=PROPERTY_VALUE(1PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3004=PROPERTY_VALUE(2PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3005=PROPERTY_VALUE(1PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3006=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3007=PROPERTY_VALUE(#2501, #180);
```

Figure H.6 — The BSU / Dictionary element relationship

H.4 A complete physical file for explicit functional models compliant with ISO 13584-101.

In this clause, the complete example of a physical file is provided. The following physical file, compliant

© ISO 2004 All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

with LIBRARY INTEGRATED INFORMATION MODEL 25 defines the PAW family and its geometry according to the view exchange protocol defined in ISO 13584 Part 101 which references the **programs** that allows to display geometry.

```
*/
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('PLIB EXPLICIT FUNCTIONAL MODEL EXAMPLE'), '1');
FILE_NAME('P25_fm_explicit_p101.p21',
          '2001-09-21T02:38:14',
          (''),
          ('LISI/ENSMA'),
          'ECCO RUNTIME SYSTEM BUILT-IN PREPROCESSOR V2.3.4',
          'ECCO RUNTIME SYSTEM V2.3.4S',
          '');
FILE_SCHEMA(('ISO13584_25_IEC61360_5_LIBRARY_IMPLICIT_SCHEMA'));
ENDSEC;

DATA;

/* Global library description */
#2=LIBRARY_IN_STANDARD_FORMAT($, $, $, $, (), #30, #11, (#7), (#12),
(#20, #30, #40), (#50, #60, #140, #130), (), #3, $, $, ());
#3=ITEM_NAMES(LABEL('Explicit functional model: Geometry'), (),
LABEL('Geometry'), $, $);
#6=ORGANIZATION('LISI/ENSMA', 'LISI/ENSMA', '');
#7=STANDARD_SIMPLE_PROGRAM_PROTOCOL(#6, $, 'ISO_IS_13584_31', '001', $,
#8, $, 'FORTRAN', .SOURCE., $, $, $);
#8=ITEM_NAMES(LABEL('Geometric prog. interface'), (),
LABEL('ISO_IS_13584_31'), $, $);
#11=LIBRARY_IIM_IDENTIFICATION($, 'IS', 'ISO13584_25_IEC61360_5', 2003,
'6', $, ());
#12=VIEW_EXCHANGE_PROTOCOL_IDENTIFICATION($, 'IS', 'ISO13584_101',
2003, '2D', $, (#7), $);
#10=GLOBAL_LANGUAGE_ASSIGNMENT('en');

/* DICTIONARY DESCRIPTION */
/*BSU for suppliers */
#20=SUPPLIER_BSU('94/1124946367', *);  /* parts manufacturer code
following ISO 13584-26*/
#30=SUPPLIER_BSU('9/19860073600021', *);
/* LISI/ENSMA code in the coding scheme ICD=0009 : SIRET number */
#40=SUPPLIER_BSU('0112/1///13584_101_1', *);
/* Identification of ISO 13584-101 according to ISO 13584-26 */

/* BSU for classes */
#50=CLASS_BSU('Bearing', '001', #20);
#60=CLASS_BSU('PAW', '001', #20);
#130=CLASS_BSU('PAW_Geometry', '001', #30);
#140=CLASS_BSU('basic_geometry', '001', #40);

/* BSU for properties */
#90=PROPERTY_BSU('d_in', '001', #50);
#100=PROPERTY_BSU('d_out', '001', #50);
#110=PROPERTY_BSU('e', '001', #50);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#150=PROPERTY_BSU('geometry_level', '001', #140);
#160=PROPERTY_BSU('detail_level', '001', #140);
#170=PROPERTY_BSU('side', '001', #140);
#180=PROPERTY_BSU('prg', '001', #130);
#200=PROPERTY_BSU('variant', '001', #140);
#210=PROPERTY_BSU('unreg_variant', '001', #140);

/* view control variable ranges */
#155=VIEW_CONTROL_VARIABLE_RANGE(#150, 1, 1);
#165=VIEW_CONTROL_VARIABLE_RANGE(#160, 2, 2);
#175=VIEW_CONTROL_VARIABLE_RANGE(#170, 1, 6);
#205=VIEW_CONTROL_VARIABLE_RANGE (#200, 1, 1);
#215=VIEW_CONTROL_VARIABLE_RANGE (#210, 0, 0);

/* supplier description */
#31=SUPPLIER_ELEMENT(#30, $, '001', #32, #33);
#32=ORGANIZATION('LISI/ENSMA', 'LISI/ENSMA', '');
#33=ADDRESS($, $, $, $, $, $, $, 'FRANCE', $, $, $, $);

/* Dictionary properties description */
/* prg */
#91=REPRESENTATION_P_DET(#180, $, '001', #92, TEXT('variable used to
reference geometry programs'), $, $, $, $, (), $, 'A58', #93, $);
#92=ITEM_NAMES(LABEL('related program'), (), LABEL(''), $, $);
#93=PROGRAM_REFERENCE_TYPE
(('ISO13584_25_IEC61360_5_LIBRARY_IMPLICIT_SCHEMA.PROGRAM_REFERENCE'));

/* Dictionary class description */
/* Functional model class view_of definition*/
#71=FM_CLASS_VIEW_OF(#130, $, '001', #72, TEXT('Explicit functional
model class describing the 2d standard geometry of PAW'), $, $, $, $,
(#180), (), *, *, *, *, *, #140, (#155, #165, #175, #205, #215), (#150,
#160, #170, #200, #210), (), (), (), (), (), (), (), (), #60, (#90,
#100, #110), (),(),());
#72=ITEM_NAMES(LABEL('Functional model class of PAW'), (), LABEL('fm
class of PAW'), $, $);

/* LIBRARY DESCRIPTION */

#1300=EXPLICIT_FUNCTIONAL_MODEL_CLASS_EXTENSION(#130, (#2501, #2502,
#2503, #2504, #2505, #2506), (#7), (#12), '001', '001', (), (), (#90,
#100, #110, #170), (#3000, #3010, #3020, #3030, #3040, #3050, #3100,
#3110, #3120, #3130, #3140, #3150, #3200, #3210, #3220, #3230, #3240,
#3250, #3300, #3310, #3320, #3330, #3340, #3350, #3400, #3410, #3420,
#3430, #3440, #3450), .T., $, (#90, #100, #110), #180, $, $ ,(), $);

#2501=PROGRAM_REFERENCE(#7, #2601, 'Add1_PAW', 'PAW_p1', (#90, #100,
#110), (), ());
#2502=PROGRAM_REFERENCE(#7, #2602, 'Add2_PAW', 'PAW_p2', (#90, #100,
#110), (), ());
#2503=PROGRAM_REFERENCE(#7, #2603, 'Add3_PAW', 'PAW_p3', (#90, #100,
#110), (), ());
#2504=PROGRAM_REFERENCE(#7, #2604, 'Add4_PAW', 'PAW_p4', (#90, #100,
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#110), (), ());
#2505=PROGRAM_REFERENCE(#7, #2605, 'Add5_PAW', 'PAW_p5', (#90, #100,
#110), (), ());
#2506=PROGRAM_REFERENCE(#7, #2606, 'Add6_PAW', 'PAW_p6', (#90, #100,
#110), (), ());

#2601=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2701));
#2602=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2702));
#2603=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2703));
#2604=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2704));
#2605=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2705));
#2606=NOT_TRANSLATABLE_EXTERNAL_CONTENT((#2706));
#2701=LANGUAGE_SPECIFIC_CONTENT((#2801), #2801, $);
#2702=LANGUAGE_SPECIFIC_CONTENT((#2802), #2802, $);
#2703=LANGUAGE_SPECIFIC_CONTENT((#2803), #2803, $);
#2704=LANGUAGE_SPECIFIC_CONTENT((#2804), #2804, $);
#2705=LANGUAGE_SPECIFIC_CONTENT((#2805), #2805, $);
#2706=LANGUAGE_SPECIFIC_CONTENT((#2806), #2806, $);
#2801=EXTERNAL_FILE_UNIT('PAW_p1.for', '7bit');
#2802=EXTERNAL_FILE_UNIT('PAW_p2.for', '7bit');
#2803=EXTERNAL_FILE_UNIT('PAW_p3.for', '7bit');
#2804=EXTERNAL_FILE_UNIT('PAW_p4.for', '7bit');
#2805=EXTERNAL_FILE_UNIT('PAW_p5.for', '7bit');
#2806=EXTERNAL_FILE_UNIT('PAW_p6.for', '7bit');

/*
     GM      |                        FV                          | FM
-------------------------------------------------------------------------
d_in d_out e | side | geom_level | det_level | var | unreg_var    | prg
*/

#3000=LIB_F_MODEL_INSTANCE(#130, (#3001, #3008, #3009, #3002, #3003,
#3004, #3005, #3006, #3007), ());
#3001=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#3008=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3009=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
#3002=PROPERTY_VALUE(INTEGER_VALUE(1), #170);
#3003=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3004=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3005=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3006=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3007=PROPERTY_VALUE(#2501, #180);

#3010=LIB_F_MODEL_INSTANCE(#130, (#3011, #3018, #3019, #3012, #3013,
#3014, #3015, #3016, #3017), ());
#3011=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#3018=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3019=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
#3012=PROPERTY_VALUE(INTEGER_VALUE(2), #170);
#3013=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3014=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3015=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3016=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3017=PROPERTY_VALUE(#2502, #180);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#3020=LIB_F_MODEL_INSTANCE(#130, (#3021, #3028, #3029, #3022, #3023,
#3024, #3025, #3026, #3027), ());
#3021=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#3028=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3029=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
#3022=PROPERTY_VALUE(INTEGER_VALUE(3), #170);
#3023=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3024=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3025=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3026=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3027=PROPERTY_VALUE(#2503, #180);

#3030=LIB_F_MODEL_INSTANCE(#130, (#3031, #3038, #3039, #3032, #3033,
#3034, #3035, #3036, #3037), ());
#3031=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#3038=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3039=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
#3032=PROPERTY_VALUE(INTEGER_VALUE(4), #170);
#3033=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3034=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3035=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3036=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3037=PROPERTY_VALUE(#2504, #180);

#3040=LIB_F_MODEL_INSTANCE(#130, (#3041, #3048, #3049, #3042, #3043,
#3044, #3045, #3046, #3047), ());
#3041=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#3048=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3049=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
#3042=PROPERTY_VALUE(INTEGER_VALUE(5), #170);
#3043=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3044=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3045=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3046=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3047=PROPERTY_VALUE(#2505, #180);

#3050=LIB_F_MODEL_INSTANCE(#130, (#3051, #3058, #3059, #3052, #3053,
#3054, #3055, #3056, #3057), ());
#3051=PROPERTY_VALUE(REAL_VALUE(10.0), #90);
#3058=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3059=PROPERTY_VALUE(REAL_VALUE(15.0), #110);
#3052=PROPERTY_VALUE(INTEGER_VALUE(6), #170);
#3053=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3054=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3055=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3056=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3057=PROPERTY_VALUE(#2506, #180);

#3100=LIB_F_MODEL_INSTANCE(#130, (#3101, #3108, #3109, #3102, #3103,
#3104, #3105, #3106, #3107), ());
#3101=PROPERTY_VALUE(REAL_VALUE(11.0), #90);
#3108=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#3109=PROPERTY_VALUE(REAL_VALUE(16.5), #110);
#3102=PROPERTY_VALUE(INTEGER_VALUE(1), #170);
#3103=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3104=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3105=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3106=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3107=PROPERTY_VALUE(#2501, #180);

#3110=LIB_F_MODEL_INSTANCE(#130, (#3111, #3118, #3119, #3112, #3113,
#3114, #3115, #3116, #3117), ());
#3111=PROPERTY_VALUE(REAL_VALUE(11.0), #90);
#3118=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3119=PROPERTY_VALUE(REAL_VALUE(16.5), #110);
#3112=PROPERTY_VALUE(INTEGER_VALUE(2), #170);
#3113=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3114=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3115=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3116=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3117=PROPERTY_VALUE(#2502, #180);

#3120=LIB_F_MODEL_INSTANCE(#130, (#3121, #3128, #3129, #3122, #3123,
#3124, #3125, #3126, #3127), ());
#3121=PROPERTY_VALUE(REAL_VALUE(11.0), #90);
#3128=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3129=PROPERTY_VALUE(REAL_VALUE(16.5), #110);
#3122=PROPERTY_VALUE(INTEGER_VALUE(3), #170);
#3123=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3124=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3125=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3126=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3127=PROPERTY_VALUE(#2503, #180);

#3130=LIB_F_MODEL_INSTANCE(#130, (#3131, #3138, #3139, #3132, #3133,
#3134, #3135, #3136, #3137), ());
#3131=PROPERTY_VALUE(REAL_VALUE(11.0), #90);
#3138=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3139=PROPERTY_VALUE(REAL_VALUE(16.5), #110);
#3132=PROPERTY_VALUE(INTEGER_VALUE(4), #170);
#3133=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3134=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3135=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3136=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3137=PROPERTY_VALUE(#2504, #180);

#3140=LIB_F_MODEL_INSTANCE(#130, (#3141, #3148, #3149, #3142, #3143,
#3144, #3145, #3146, #3147), ());
#3141=PROPERTY_VALUE(REAL_VALUE(11.0), #90);
#3148=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3149=PROPERTY_VALUE(REAL_VALUE(16.5), #110);
#3142=PROPERTY_VALUE(INTEGER_VALUE(5), #170);
#3143=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3144=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3145=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3146=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
```

--`,``,,,`-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#3147=PROPERTY_VALUE(#2505, #180);

#3150=LIB_F_MODEL_INSTANCE(#130, (#3151, #3158, #3159, #3152, #3153,
#3154, #3155, #3156, #3157), ());
#3151=PROPERTY_VALUE(REAL_VALUE(11.0), #90);
#3158=PROPERTY_VALUE(REAL_VALUE(1.0), #100);
#3159=PROPERTY_VALUE(REAL_VALUE(16.5), #110);
#3152=PROPERTY_VALUE(INTEGER_VALUE(6), #170);
#3153=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3154=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3155=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3156=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3157=PROPERTY_VALUE(#2506, #180);

#3200=LIB_F_MODEL_INSTANCE(#130, (#3201, #3208, #3209, #3202, #3203,
#3204, #3205, #3206, #3207), ());
#3201=PROPERTY_VALUE(REAL_VALUE(13.0), #90);
#3208=PROPERTY_VALUE(REAL_VALUE(2.0), #100);
#3209=PROPERTY_VALUE(REAL_VALUE(19.5), #110);
#3202=PROPERTY_VALUE(INTEGER_VALUE(1), #170);
#3203=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3204=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3205=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3206=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3207=PROPERTY_VALUE(#2501, #180);

#3210=LIB_F_MODEL_INSTANCE(#130, (#3211, #3218, #3219, #3212, #3213,
#3214, #3215, #3216, #3217), ());
#3211=PROPERTY_VALUE(REAL_VALUE(13.0), #90);
#3218=PROPERTY_VALUE(REAL_VALUE(2.0), #100);
#3219=PROPERTY_VALUE(REAL_VALUE(19.5), #110);
#3212=PROPERTY_VALUE(INTEGER_VALUE(2), #170);
#3213=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3214=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3215=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3216=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3217=PROPERTY_VALUE(#2502, #180);

#3220=LIB_F_MODEL_INSTANCE(#130, (#3221, #3228, #3229, #3222, #3223,
#3224, #3225, #3226, #3227), ());
#3221=PROPERTY_VALUE(REAL_VALUE(13.0), #90);
#3228=PROPERTY_VALUE(REAL_VALUE(2.0), #100);
#3229=PROPERTY_VALUE(REAL_VALUE(19.5), #110);
#3222=PROPERTY_VALUE(INTEGER_VALUE(3), #170);
#3223=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3224=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3225=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3226=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3227=PROPERTY_VALUE(#2503, #180);

#3230=LIB_F_MODEL_INSTANCE(#130, (#3231, #3238, #3239, #3232, #3233,
#3234, #3235, #3236, #3237), ());
#3231=PROPERTY_VALUE(REAL_VALUE(13.0), #90);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS
Not for Resale

```
#3238=PROPERTY_VALUE(REAL_VALUE(2.0), #100);
#3239=PROPERTY_VALUE(REAL_VALUE(19.5), #110);
#3232=PROPERTY_VALUE(INTEGER_VALUE(4), #170);
#3233=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3234=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3235=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3236=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3237=PROPERTY_VALUE(#2504, #180);

#3240=LIB_F_MODEL_INSTANCE(#130, (#3241, #3248, #3249, #3242, #3243,
#3244, #3245, #3246, #3247), ());
#3241=PROPERTY_VALUE(REAL_VALUE(13.0), #90);
#3248=PROPERTY_VALUE(REAL_VALUE(2.0), #100);
#3249=PROPERTY_VALUE(REAL_VALUE(19.5), #110);
#3242=PROPERTY_VALUE(INTEGER_VALUE(5), #170);
#3243=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3244=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3245=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3246=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3247=PROPERTY_VALUE(#2505, #180);

#3250=LIB_F_MODEL_INSTANCE(#130, (#3251, #3258, #3259, #3252, #3253,
#3254, #3255, #3256, #3257), ());
#3251=PROPERTY_VALUE(REAL_VALUE(13.0), #90);
#3258=PROPERTY_VALUE(REAL_VALUE(2.0), #100);
#3259=PROPERTY_VALUE(REAL_VALUE(19.5), #110);
#3252=PROPERTY_VALUE(INTEGER_VALUE(6), #170);
#3253=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3254=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3255=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3256=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3257=PROPERTY_VALUE(#2506, #180);

#3300=LIB_F_MODEL_INSTANCE(#130, (#3301, #3308, #3309, #3302, #3303,
#3304, #3305, #3306, #3307), ());
#3301=PROPERTY_VALUE(REAL_VALUE(17.0), #90);
#3308=PROPERTY_VALUE(REAL_VALUE(3.0), #100);
#3309=PROPERTY_VALUE(REAL_VALUE(25.5), #110);
#3302=PROPERTY_VALUE(INTEGER_VALUE(1), #170);
#3303=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3304=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3305=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3306=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3307=PROPERTY_VALUE(#2501, #180);

#3310=LIB_F_MODEL_INSTANCE(#130, (#3311, #3318, #3319, #3312, #3313,
#3314, #3315, #3316, #3317), ());
#3311=PROPERTY_VALUE(REAL_VALUE(17.0), #90);
#3318=PROPERTY_VALUE(REAL_VALUE(3.0), #100);
#3319=PROPERTY_VALUE(REAL_VALUE(25.5), #110);
#3312=PROPERTY_VALUE(INTEGER_VALUE(2), #170);
#3313=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3314=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3315=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
```

--`,`,``,,-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#3316=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3317=PROPERTY_VALUE(#2502, #180);

#3320=LIB_F_MODEL_INSTANCE(#130, (#3321, #3328, #3329, #3322, #3323,
#3324, #3325, #3326, #3327), ());
#3321=PROPERTY_VALUE(REAL_VALUE(17.0), #90);
#3328=PROPERTY_VALUE(REAL_VALUE(3.0), #100);
#3329=PROPERTY_VALUE(REAL_VALUE(25.5), #110);
#3322=PROPERTY_VALUE(INTEGER_VALUE(3), #170);
#3323=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3324=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3325=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3326=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3327=PROPERTY_VALUE(#2503, #180);

#3330=LIB_F_MODEL_INSTANCE(#130, (#3331, #3338, #3339, #3332, #3333,
#3334, #3335, #3336, #3337), ());
#3331=PROPERTY_VALUE(REAL_VALUE(17.0), #90);
#3338=PROPERTY_VALUE(REAL_VALUE(3.0), #100);
#3339=PROPERTY_VALUE(REAL_VALUE(25.5), #110);
#3332=PROPERTY_VALUE(INTEGER_VALUE(4), #170);
#3333=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3334=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3335=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3336=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3337=PROPERTY_VALUE(#2504, #180);

#3340=LIB_F_MODEL_INSTANCE(#130, (#3341, #3348, #3349, #3342, #3343,
#3344, #3345, #3346, #3347), ());
#3341=PROPERTY_VALUE(REAL_VALUE(17.0), #90);
#3348=PROPERTY_VALUE(REAL_VALUE(3.0), #100);
#3349=PROPERTY_VALUE(REAL_VALUE(25.5), #110);
#3342=PROPERTY_VALUE(INTEGER_VALUE(5), #170);
#3343=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3344=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3345=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3346=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3347=PROPERTY_VALUE(#2505, #180);

#3350=LIB_F_MODEL_INSTANCE(#130, (#3351, #3358, #3359, #3352, #3353,
#3354, #3355, #3356, #3357), ());
#3351=PROPERTY_VALUE(REAL_VALUE(17.0), #90);
#3358=PROPERTY_VALUE(REAL_VALUE(3.0), #100);
#3359=PROPERTY_VALUE(REAL_VALUE(25.5), #110);
#3352=PROPERTY_VALUE(INTEGER_VALUE(6), #170);
#3353=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3354=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3355=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3356=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3357=PROPERTY_VALUE(#2506, #180);

#3400=LIB_F_MODEL_INSTANCE(#130, (#3401, #3408, #3409, #3402, #3403,
#3404, #3405, #3406, #3407), ());
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#3401=PROPERTY_VALUE(REAL_VALUE(19.0), #90);
#3408=PROPERTY_VALUE(REAL_VALUE(4.0), #100);
#3409=PROPERTY_VALUE(REAL_VALUE(28.5), #110);
#3402=PROPERTY_VALUE(INTEGER_VALUE(1), #170);
#3403=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3404=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3405=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3406=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3407=PROPERTY_VALUE(#2501, #180);

#3410=LIB_F_MODEL_INSTANCE(#130, (#3411, #3418, #3419, #3412, #3413,
#3414, #3415, #3416, #3417), ());
#3411=PROPERTY_VALUE(REAL_VALUE(19.0), #90);
#3418=PROPERTY_VALUE(REAL_VALUE(4.0), #100);
#3419=PROPERTY_VALUE(REAL_VALUE(28.5), #110);
#3412=PROPERTY_VALUE(INTEGER_VALUE(2), #170);
#3413=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3414=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3415=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3416=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3417=PROPERTY_VALUE(#2502, #180);

#3420=LIB_F_MODEL_INSTANCE(#130, (#3421, #3428, #3429, #3422, #3423,
#3424, #3425, #3426, #3427), ());
#3421=PROPERTY_VALUE(REAL_VALUE(19.0), #90);
#3428=PROPERTY_VALUE(REAL_VALUE(4.0), #100);
#3429=PROPERTY_VALUE(REAL_VALUE(28.5), #110);
#3422=PROPERTY_VALUE(INTEGER_VALUE(3), #170);
#3423=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3424=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3425=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3426=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3427=PROPERTY_VALUE(#2503, #180);

#3430=LIB_F_MODEL_INSTANCE(#130, (#3431, #3438, #3439, #3432, #3433,
#3434, #3435, #3436, #3437), ());
#3431=PROPERTY_VALUE(REAL_VALUE(19.0), #90);
#3438=PROPERTY_VALUE(REAL_VALUE(4.0), #100);
#3439=PROPERTY_VALUE(REAL_VALUE(28.5), #110);
#3432=PROPERTY_VALUE(INTEGER_VALUE(4), #170);
#3433=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3434=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3435=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3436=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3437=PROPERTY_VALUE(#2504, #180);

#3440=LIB_F_MODEL_INSTANCE(#130, (#3441, #3448, #3449, #3442, #3443,
#3444, #3445, #3446, #3447), ());
#3441=PROPERTY_VALUE(REAL_VALUE(19.0), #90);
#3448=PROPERTY_VALUE(REAL_VALUE(4.0), #100);
#3449=PROPERTY_VALUE(REAL_VALUE(28.5), #110);
#3442=PROPERTY_VALUE(INTEGER_VALUE(5), #170);
#3443=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3444=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
```

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

```
#3445=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3446=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3447=PROPERTY_VALUE(#2505, #180);

#3450=LIB_F_MODEL_INSTANCE(#130, (#3451, #3458, #3459, #3452, #3453,
#3454, #3455, #3456, #3457), ());
#3451=PROPERTY_VALUE(REAL_VALUE(19.0), #90);
#3458=PROPERTY_VALUE(REAL_VALUE(4.0), #100);
#3459=PROPERTY_VALUE(REAL_VALUE(28.5), #110);
#3452=PROPERTY_VALUE(INTEGER_VALUE(6), #170);
#3453=PROPERTY_VALUE(INTEGER_VALUE(1), #150);
#3454=PROPERTY_VALUE(INTEGER_VALUE(2), #160);
#3455=PROPERTY_VALUE(INTEGER_VALUE(1), #200);
#3456=PROPERTY_VALUE(INTEGER_VALUE(0), #210);
#3457=PROPERTY_VALUE(#2506, #180);

ENDSEC;
END-ISO-10303-21;
```

© ISO 2004 All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

Index

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale

© ISO 2004 All rights reserved

--`,``,,-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

--`,,,`,,-`-`,,`,,`,`,,`---

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS

Not for Resale

ICS 25.040.40

Price based on 101 pages

© ISO 2004 – All rights reserved

Copyright International Organization for Standardization
Reproduced by IHS under license with ISO
No reproduction or networking permitted without license from IHS Not for Resale