



BSI Standards Publication

Industrial automation systems and integration — Parts library

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

National foreword

This British Standard is the UK implementation of [ISO 13584-25:2004](#).

The UK participation in its preparation was entrusted to Technical Committee AMT/4, Industrial data and manufacturing interfaces.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2023
Published by BSI Standards Limited 2023

ISBN 978 0 539 22188 6

ICS 25.040.40

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 March 2023.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

INTERNATIONAL STANDARD

ISO
13584-25

First edition
2004-06-15

Industrial automation systems and integration — Parts library —

Part 25: 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)

AVA Documents
Document Sharing Hub
© ISO 2004



COPYRIGHT PROTECTED DOCUMENT

© ISO 2004, Published in Switzerland

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

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

Contents

Page

Foreword	v
Introduction	vi
1 Scope	1
2 Normative references	1
3 Terms, definitions and abbreviations	2
4 Structure of ISO 13584-25	6
4.1 Generic resources	6
4.1.1 ISO13584_IEC61360_dictionary_aggregate_extension_schema	7
4.1.2 ISO13584_aggregate_value_schema	7
4.2 Library integrated model	7
4.2.1 Conformance class 1: minimal dictionaries	8
4.2.2 Conformance class 2: dictionaries of items classes	8
4.2.3 Conformance class 3: complete dictionaries	8
4.2.4 Conformance class 4: complete dictionaries with limited nested aggregate values	8
4.2.5 Conformance class 5: libraries of item classes	8
4.2.6 Conformance class 6: complete libraries	9
4.2.7 Conformance class 7: complete libraries with limited nested aggregate values	9
4.2.8 Conformance class 10: library instances	9
4.2.9 Conformance class 11: library instances with associated dictionary definitions	9
5 Fundamental concepts and assumptions	9
5.1 Aggregate-structured value of properties	9
5.2 Explicit description of a class extension	10
6 ISO13584_IEC61360_dictionary_aggregate_extension_schema	10
6.1 Introduction to the ISO13584_IEC61360_dictionary_aggregate_extension_schema	10
6.2 ISO13584_IEC61360_dictionary_aggregate_extension_schema entity definitions	10
6.2.1 Aggregate_entity_instance_type	10
6.2.2 Aggregate_type	11
6.2.3 List_type	11
6.2.4 Set_type	12
6.2.5 Bag_type	12
6.2.6 Array_type	13
6.2.7 Set_with_subset_constraint_type	13
7 ISO13584_aggregate_value_schema	14
7.1 Introduction to the ISO13584_aggregate_value_schema	15
7.2 ISO13584_aggregate_value_schema entity definitions	15
7.2.1 Aggregate_entity_instance_value	15
7.2.2 Aggregate_value	15
7.2.3 List_value	16
7.2.4 Set_value	16
7.2.5 Bag_value	16
7.2.6 Array_value	17
7.2.7 Set_with_subset_constraint_value	17
7.3 ISO13584_aggregate_value_schema rule definition	18
7.3.1 Allowed_aggregate_values rule	18
7.4 ISO13584_aggregatevalue_schema function definitions	18
7.4.1 Compatible_complete_types_and_value function	18
7.4.2 Compatible_aggregate_domain_and_aggregate_value function	19
7.4.3 Data_type_final_type function	20
7.4.4 Compatible_aggregate_type_and_value function	20

7.4.5	Compatible_final_type_and_value function	24
8	Library integrated information model 25.....	26
8.1	ISO13584_25_IEC61360_5_liim_schema short listing	28
8.2	Conformance class requirements	32
8.2.1	Conformance class 1: minimal dictionaries.....	32
8.2.2	Conformance class 2: dictionaries of items classes	34
8.2.3	Conformance class 3: complete dictionaries	36
8.2.4	Conformance class 4: complete dictionaries with limited nested aggregate values.....	36
8.2.5	Conformance class 5: libraries of item classes.....	36
8.2.6	Conformance class 6: complete libraries.....	38
8.2.7	Conformance class 7: complete libraries with limited nested aggregate values.....	39
8.2.8	Conformance class 10: library instances	39
8.2.9	Conformance class 11: library instances with asociated dictionary definitions.....	41
Annex A	(normative) Short names of entities defined in this part	43
Annex B	(normative) Information object registration	44
Annex C	(normative) ISO13584_25_IEC61360_5_library_implicit_schema expanded listing	45
Annex D	(normative) Standard data requirements for library integrated information model 25	47
Annex E	(normative) Implementation method specific requirements for the library integrated information model 25	57
Annex F	(informative) EXPRESS-G diagrams	58
Annex G	(informative)Commented example of library integrated information model 25 physical files. Exchange of explicit general models.....	60
Annex H	(informative)Commented example of library integrated information model 25 physical files. Exchange of explicit functional models compliant with ISO 13584-101.....	67

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;
- 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.

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.

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.

[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.

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

[SOURCE: 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)

[SOURCE: ISO 13584-42:1998]

3.3
class extension

set of all the different possible instances conforming to the specification defined by a class

[SOURCE: 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

[SOURCE: ISO 13584-42:1998]

Note 1 to entry: 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

[SOURCE: ISO 13584-24:2003]

3.6
conformance requirement

a precise, text definition of a characteristic required to be present in a conforming implementation

[SOURCE: 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)

[SOURCE: 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

[SOURCE: ISO 13584-42:1998]

3.9
data type

a domain of values

[SOURCE: ISO 10303-11:1994]

3.10
family of parts

a simple or generic family of parts

[SOURCE: 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

[SOURCE: ISO 13584-1:2001]

3.12

functional view of a part

the data that represent one representation category of a part in product data

[SOURCE: ISO 13584-1:2001]

Note 1 to entry: 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

[SOURCE: 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

[SOURCE: 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](#)

[SOURCE: ISO 13584-24:2003]

Note 1 to entry: 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

[SOURCE: ISO 13584-1:2001]

3.17

library part data

the data that represent a part in a library

[SOURCE: 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

[SOURCE: 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

[SOURCE: ISO 13584-24:2003]

Note 1 to entry: 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

[SOURCE: ISO 13584-24:2003]

3.21

library specification of a class

explicit representation of a class extension in a supplier library

[SOURCE: ISO 13584-24:2003]

Note 1 to entry: 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 to entry: 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

[SOURCE: ISO 13584-1:2001]

3.23

property

an information that may be represented by a data element type

[SOURCE: ISO 13584-42:1998]

3.24

representation category

an abstraction used to distinguish between different possible user requirements regarding a part representation

[SOURCE: ISO 13584-1:2001]

Note 1 to entry: 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

[SOURCE: 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

[SOURCE: 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

[SOURCE: 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

[SOURCE: 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

[SOURCE: 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

[SOURCE: 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.

Note 1 to entry: 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;
- 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](#).

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 Clause 8.2.8 of this part of [ISO 13584](#).

4.2.9 Conformance class 11: library instances with associated 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:
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 EXPRESSdefined 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'.

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).

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.

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 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;
(*
```

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 or 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);  
  
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_values**.

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;
(*
```

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_values**.

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_values**.

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_values**.

EXPRESS specification:

```
*)  
ENTITY set_value  
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_values**.

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_values**.

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;
(*)
```

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_values** 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 or 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.

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
    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 aggregate_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 availability 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)))
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;

```