



**International
Standard**

ISO 10303-1

**Industrial automation systems
and integration — Product data
representation and exchange —**

**Part 1:
Overview and fundamental
principles**

*Systemes d'automatisation industrielle et integration
Représentation et échange de données de produits*

Partie 1: Aperçu et principes fondamentaux

**Third edition
2024-01**

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-1:2024



COPYRIGHT PROTECTED DOCUMENT

© ISO 2024

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

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

	Page
Foreword	v
Introduction	vi
1 Scope	1
2 Normative references	2
3 Terms, definitions and abbreviated terms	2
3.1 Terms and definitions.....	2
3.2 Abbreviated terms.....	2
4 Overview of the ISO 10303 series	2
4.1 Purpose.....	2
4.2 Scope of the ISO 10303 series.....	3
4.3 Fundamental principles.....	3
4.3.1 General.....	3
4.3.2 Integrated resources (IRs) and Core model.....	3
4.3.3 Support for application protocols (APs).....	4
4.3.4 Implementation methods.....	4
4.3.5 Implementations.....	4
4.3.6 Conformance testing.....	5
5 Architecture of the ISO 10303 series	5
5.1 Overview.....	5
5.2 Types of architecture.....	6
5.2.1 General.....	6
5.2.2 Sharing interpretations in the initial architecture.....	6
5.2.3 Sharing interpretations in the modular architecture.....	6
5.2.4 Sharing interpretations in the extended architecture.....	7
6 Structure of the ISO 10303 series	9
6.1 General.....	9
6.2 Description methods.....	9
6.2.1 Purpose.....	9
6.2.2 The EXPRESS modelling language.....	10
6.2.3 Transformation description methods.....	10
6.3 Implementation methods.....	10
6.3.1 Purpose.....	10
6.3.2 Use of formal language.....	10
6.3.3 Implementation methods for product data described using the EXPRESS language.....	11
6.4 Integrated resources (IRs).....	12
6.4.1 Purpose.....	12
6.4.2 Generic resources.....	12
6.4.3 Application resources.....	12
6.5 Application interpreted construct (AIC).....	12
6.5.1 Purpose.....	12
6.5.2 Characteristics.....	13
6.6 Application modules (AMs).....	13
6.6.1 Purpose.....	13
6.6.2 Characteristics.....	13
6.6.3 Business benefits.....	13
6.7 Application protocols (APs).....	14
6.7.1 Purpose.....	14
6.7.2 Definition of information requirements.....	14
6.7.3 Information representation for the modular architecture.....	14
6.7.4 Information representation for the extended architecture.....	14
6.7.5 Implementation methods.....	14
6.7.6 Conformance requirements.....	15

ISO 10303-1:2024(en)

6.8	Core model.....	15
6.8.1	Purpose.....	15
6.8.2	Information representation in core technical capabilities.....	15
6.9	Application domain models (ADMs).....	15
6.9.1	Purpose.....	15
6.9.2	Information representation.....	15
6.10	Usage guides.....	15
6.10.1	Purpose.....	15
6.10.2	Characteristics.....	15
6.10.3	Document structure.....	16
6.10.4	Content.....	16
6.11	Conformance testing methodology and framework.....	16
6.11.1	Purpose.....	16
6.11.2	Procedures for conformance testing.....	16
6.11.3	Abstract test methods (ATMs).....	16
6.12	Abstract test suites (ATSS).....	17
7	Information object registration scheme	17
	Annex A (normative) Information object registration	20
	Bibliography	21

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-1:2024

Foreword

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

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 4, *Industrial data*.

This third edition cancels and replaces the second edition (ISO 10303-1:2021), which has been technically revised.

The main changes are as follows:

- all terms removed, and included in ISO 10303-2;
- inclusion of STEP extended architecture.

A list of all parts in the ISO 10303 series can be found on the ISO website.

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

Introduction

The ISO 10303 series of International Standards describe the computer-interpretable representation of product information for the exchange of product data. The objective is to provide a neutral mechanism capable of describing products throughout their life cycle. This mechanism is suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases, and as a basis for archiving.

The information generated about a product during its design, manufacture, use, maintenance, and disposal is used for many purposes. The use can involve many information systems, including some that can be located in different organizations. In order to support such uses, organizations need to be able to represent their product information in a common computer-interpretable form that is required to remain complete and consistent when exchanged among different information systems.

This document is an overview of the ISO 10303 series. It specifies the overall scope of the ISO 10303 series and describes the ISO 10303 series architectures and structure. It describes the various parts of the ISO 10303 series and the relationships among them.

The ISO 10303 series is organized as a series of parts, each published separately.

Each part of the ISO 10303 series is a member of one of the following series: description methods, implementation methods, conformance testing methodology and framework, integrated generic resources, integrated application resources, Core model, application protocols (APs), abstract test suites (ATs), application interpreted constructs (AICs), application modules (AMs) and application domain models (ADM).

There is a set of standing documents that provide guidelines for developing International Standards produced by ISO/TC 184/SC 4. These are listed in the SC 4 organization handbook^[1].

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-1:2024

Industrial automation systems and integration — Product data representation and exchange —

Part 1: Overview and fundamental principles

1 Scope

This document provides an overview of the ISO 10303 series.

This document defines the architectural principles of product information representation and exchange used in the ISO 10303 series. It specifies the characteristics of the various sets of parts in the ISO 10303 series and the relationships among them.

The following are within the scope of this document:

- scope statement for the ISO 10303 series as a whole;
- overview of the ISO 10303 series;
- architectures of the ISO 10303 series;
- structure of the ISO 10303 series;
- overview of data specification methods used in the ISO 10303 series;

NOTE This includes the EXPRESS data specification language and graphical presentation of product information models.

- introduction to the ISO 10303 series:
 - integrated resources (IRs);
 - application interpreted constructs (AICs);
 - application modules (AMs);
 - application domain models (ADMs);
 - Core model;
 - business object models (deprecated);
 - application protocols (APs);
 - implementation methods;
 - usage guides;
 - conformance testing methodology and framework;
 - abstract test suites (ATs);
- scheme for identification of schemas and other information objects defined in the ISO 10303 series.

2 Normative references

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

ISO 10303-2, *Industrial automation systems and integration — Product data representation and exchange — Part 2: STEP Vocabulary*

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

3 Terms, definitions and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 10303-2 apply.

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

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

3.2 Abbreviated terms

AO	application object
CTC	core technical capability
DTD	document type definition
HDF	hierarchical data format ^[2]
SDAI	standard data access interface ^[3]
URL	uniform resource locator
UML	unified modeling language
XMI	XML metadata interchange ^[4]
XML	extensible mark-up language
XLSX	Microsoft Excel open XML spreadsheet

4 Overview of the ISO 10303 series

4.1 Purpose

The purpose of the ISO 10303 series is to specify a form for the unambiguous representation and exchange of computer-interpretable product data throughout the life of a product. This form is independent of any specific computer system. This form enables consistent implementations across multiple applications and systems. The ISO 10303 series architecture permits different implementation methods to be used for storing, accessing, transferring, and archiving product data. The ISO 10303 series define a rigorous process for testing implementations for conformance.

4.2 Scope of the ISO 10303 series

The ISO 10303 series provides a representation of product information along with the necessary mechanisms and definitions to enable product data to be exchanged. The exchange is among different information systems and environments associated with the complete product life cycle, including product design, manufacture, use, maintenance and final disposition.

Specifically, the ISO 10303 series provides:

- representation of product information, including components and assemblies;
- implementation methods for exchange of product data, including storing, transferring, accessing, and archiving.

4.3 Fundamental principles

4.3.1 General

The ISO 10303 series separates the representation of product information from the implementation methods used for product data exchange, product data access and product data archival.

NOTE The following are assumed:

- data access is machine-to-machine sharing of data;
- data exchange is exchange of machine-interpretable files;
- data archival is persistent storage of machine-interpretable data and associated defining schemata.

EXAMPLE ISO 10303-22, Standard data access interface (SDAI), specifies data access.

The ISO 10303 series implements a three-layer architecture where the business concerns are separate from the specification of the information models, which are separate from implementation concerns.^[5]

A common resource provides a single representation of product information for many applications. This representation may be adapted to meet the needs of specific applications.

An application protocol (AP) specifies the representation of product information for one or more applications.

The ISO 10303 series specifies the implementation methods that support the exchange of product data defined in APs.

The ISO 10303 series uses EXPRESS or SysML^[6] to specify the representation of product information. This use of the formal languages EXPRESS or SysML provides unambiguous and consistent representation and facilitates development of implementations. The representation of product information shall be annotated with a literal definition for each construct.

The ISO 10303 series provides a methodology and framework for conformance testing of implementations.

4.3.2 Integrated resources (IRs) and Core model

4.3.2.1 Integrated resources (IRs)

A set of IRs shall provide the specification of a representation of product information. Each IR comprises a set of descriptions, written in a formal data specification language, applicable to product data known as resource constructs. One set may be dependent on other sets for its definition. A single resource construct may represent similar information for different applications.

The IRs in the ISO 10303 series are divided into two groups: generic resources and application resources. The generic resources are independent of applications and may reference other resources. The application resources may reference other resources and may add other resource constructs for use by a group of

similar applications. The IRs may reference product data descriptions written using EXPRESS from other International Standards.

4.3.2.2 Core model

The Core model shall provide the specification of a representation of product information that is independent of applications. It is equivalent to the IRs.

The Core information model is divided into a set of core technical capabilities (CTC). Each CTC comprises a set of objects, relationships and descriptions, written in a formal data specification language. One CTC may reference objects in other CTCs. A CTC can be reused in, or customised by, applications.

The Core model is mapped to the Reference ARM for SysML mapping (see [6.6](#)) using the artefact ARM in SysML. This relates it to the IRs.

4.3.3 Support for application protocols (APs)

The IRs and Core model each defines a generic information model for product information. They are not sufficient to support the information requirements of an AP without the addition of application specific constraints.

The ISO 10303 series defines APs in which the IRs or Core model are interpreted to meet the product information requirements of specific applications. The interpretation is achieved by selecting resource constructs from the IRs or the Core model, and their meaning may be refined by applying application specific constraints. In the modular architecture (see [5.2.3](#)), this interpretation results in an application interpreted model (AIM) and a mapping specification. In the extended architecture (see [5.2.4](#)), this interpretation results in an application domain model (ADM). The AIM or ADM is documented as part of an AP.

The ISO 10303 series has a mechanism to ensure consistent interpretation when a resource construct represents the same information requirement in different APs. The mechanism uses application modules (AMs) that document the harmonized requirements along with the interpreted resource constructs.

NOTE 1 There is an alternative legacy mechanism that uses interpreted resource constructs appearing in more than one AP as a documented application interpreted construct (AIC). New AICs cannot be created.

NOTE 2 See [6.5](#), [6.6](#) and [6.9](#) for more details on AICs, AMs and ADM.

The scope and information requirements of the APs are specified using the terminology of the application domain. The AP provides a mapping to show how the interpretation of the IRs or Core model is used to meet the information requirements of the application domain.

The ISO 10303 series provides ADMs that present complex models in a form that can be more understandable to application experts.

4.3.4 Implementation methods

Each implementation method included in the ISO 10303 series (see [6.3.3.2](#)) is specified by a mapping from a formal information modelling language used in ISO 10303 (such as the EXPRESS language) onto the formal language used for the method. The mapping is independent of the AP. The mapping is expressed in a formal notation, where the notation is defined in a description method for transformation (see [6.2.3](#)).

NOTE Standardized implementation methods that do not have a mapping from the EXPRESS language can be used. These have a description method for transformation (see [6.2.3](#)).

EXAMPLE Other standardized implementation methods are XML and Open API.

4.3.5 Implementations

An AP may specify one or more applicable implementation methods. These may be chosen from the set of implementation methods in the ISO 10303 series (see [6.3](#)), or other standardized implementation methods

for which there is a transformation description method (see 6.2.3). An AP implementation shall apply one or more of the implementation methods specified in the AP.

4.3.6 Conformance testing

Conformance of an implementation to an AP is specified by the conformance requirements in the AP.

A set of tests, specified in an abstract test suite (ATS), may be defined for each AP. When associated with an abstract test method (ATM), these tests can be used to assess the conformance of an implementation. The overall framework for conformance assessment is specified in ISO 10303-31.

An ATM for each implementation method is specified in one of the conformance testing methodology and framework set of the ISO 10303 series.

The scope of conformance testing of a specific implementation is the requirements specified for the conformance classes claimed for the implementation in the protocol implementation conformance statement (PICS). The tests are selected from the ATS, based on the conformance class claimed by the implementation. Test results provide the basis for conformance assessment.

The ISO 10303 series contains ATs and defines ATMs for performing conformance testing to provide a basis for test results that are repeatable, comparable, and auditable. Inclusion of conformance testing procedures in the ISO 10303 series is intended to promote widespread acceptance of test results.

5 Architecture of the ISO 10303 series

5.1 Overview

The ISO 10303 series architecture is designed to support the development of standards for product data exchange, product data access and product data archival. The architecture is constrained by the following concepts:

- the scope of what is standardized and what is conformance tested is set at the level of an application;
- information requirements are based on a model of a business activity;
- information requirements are standardized using a modelling language;
- a mapping is a specification of the data structures used from the IRs or the Core model and population constraints against that usage satisfying application requirements.

EXAMPLE 1 An ARM-AIM mapping of the AO Activity is a query against an AIM data population of the entity executed action. For each member of executed action in the data population, a member of Activity would be created.

EXAMPLE 2 A mapping from one entity in model X uses six entities as a target in model Y. It is not possible to map from one of the six entities in model Y to the entity in model X. Instead, a query is used to locate the six entities in model Y.

The ISO 10303 series is implemented through APs. An AP consists of the following major elements:

- an application activity model (AAM) describing the business process that the information model supports;
- an application data planning model (ADPM) describing the primary concepts of an application domain and the relationships among the concepts;
- an application reference model (ARM) specifying the information requirements (optional for the extended architecture^{[7][8][9][10]});
- an information model based on the IRs, called an AIM, that is a basis for implementations of the APs using the modular architecture^[11] shall include an AIM. APs using the extended architecture may include an AIM;

- an information model for data structures, called an ADM, that is a basis for implementations of the APs using the extended architecture shall include an ADM.

The architecture is based on standardizing industry information requirements and mapping those information requirements to an information model based on the IRs or the Core model. The Core model is mapped to a reference ARM that is then mapped to the IRs. The process of mapping information requirements to the IRs is known as "interpretation".

The ISO 10303 series architecture focuses on the information required by industrial processes rather than on the processes themselves, as the processes can change over time, while the underlying information requirements are longer lasting. This focus on information allows the ISO 10303 series to support data exchange, data access, as well as product data archival.

Whenever a resource construct is used to represent the same information requirement in different APs, the same interpretation of that resource construct shall be used.

5.2 Types of architecture

5.2.1 General

The general architectural principles described in [5.1](#) have been elaborated into three architectures:

- the "initial architecture",
- the "modular architecture", and
- the "extended architecture".

All architectures adhere to the fundamental principles described in [4.3](#).

NOTE The term "ISO 10303 series architecture", without qualification, refers to the general principles described in [5.1](#) and the common aspects of its elaborations.

An ISO 10303 series project may use the initial architecture, the modular architecture, or the extended architecture, although new projects are not encouraged to use the initial architecture.

5.2.2 Sharing interpretations in the initial architecture

An AIC shall be used to specify a piece of an AIM that can be used to exchange product data common to two or more APs.

NOTE An AIC does not document the common information requirements or the mapping of those information requirements into the AIM.

An AP that complies with initial architecture is called a "monolithic AP".

5.2.3 Sharing interpretations in the modular architecture

5.2.3.1 General

Rather than relying on harmonization occurring as a by-product of consistent interpretation across APs, application requirements are first harmonized across domains and the resulting mappings are standardized in AMs. AMs are reused by other AMs and ultimately by APs.

The implementable portion of the modular architecture has two core components:

- AM:
 - a reusable data specification documented with an ARM;
 - mapping;

- MIM;
- optional usage guide.
- AP: The use of a data specification to meet the requirements of business processes.

NOTE The business object model, originally included in the modular architecture, is deprecated.

The objectives and function of the architectural components are described in [5.2.3.2](#) and [5.2.3.3](#).

5.2.3.2 Application module (AM)

The design of an AM should maximize re-usability of the

- a) harmonized requirements,
- b) associated interpretation into the IRs,
- c) data specification, and
- d) software implementations.

The design shall support re-usability by the standards developer, implementer and user.

AMs replace AICs in the modular architecture. Each AIC shall be included in one AM. The objectives of AICs and AMs are similar. They both standardize interpretation results for reuse in multiple APs. However, AICs and AMs are created differently and have different content. AMs, unlike AICs, contain harmonized information requirements and specifications of the mappings of those requirements to the IRs. An objective of modularization is to document a concept one time and then to directly reuse that concept in other AMs. The modular architecture uses EXPRESS rather than natural language for the documentation of requirements, i.e. the ARM, in an AM. This allows the use of tools to validate the dependencies between AM ARMs.

5.2.3.3 Modular application protocol (AP)

An AP that complies with modular architecture is called a "modular AP". A modular AP is a documented use of a set of AMs for a specific business process. The AMs used by an AP are organized in a tree structure. A single, normatively referenced AP AM is the data specification for the AP. This AM normatively references a collection of related AMs that provide the documentation of its information requirements and standardized interpretations. The AP AM may include specific business process rules or constraints. The AP document contains an activity model and conformance class definitions. Industry specific terminology can be mapped (see guidelines^[12]) to the generic AM terminology defined in an AP to make it more understandable to reviewers from the application domain.

5.2.3.4 Business object model (deprecated)

The business object model is deprecated and replaced by the ADM.

5.2.4 Sharing interpretations in the extended architecture

5.2.4.1 General

The extended architecture is based on the modular architecture, and allows additional implementation forms at the level of the ADM.

The additional implementable portion of the extended architecture has two core components:

- ADM: The data specification that provides the documentation of information requirements for a domain that are parametrically mapped to the Core model.
- AP: The use of a data specification to meet the requirements of business processes.

The objectives and function of the ADM are described in [5.2.4.2](#). The objectives and function of the AP are described in [5.2.4.3](#). The objectives and function of the Core model are described in [5.2.4.4](#).

5.2.4.2 Application domain model (ADM)

The ADM describes the information requirements and constraints of a specific application context in a normative schema definition. The ADM is a specific view of selections of the STEP integrated model constrained by the AP requirements.

The implementation schemas for the ADM shall be derived by applying implementation bindings specified in the description methods.

5.2.4.3 Extended architecture application protocol (AP)

An extended architecture AP has a single documented ADM for a specific business process. The ADM is the data specification for the AP that provides the documentation of its information requirements and standardized interpretations. The ADM may include specific business process rules or constraints. The AP document contains an AAM, ADPM and conformance class definitions.

5.2.4.4 Core model

The Core model is an information model shared by all ADMs. The Core model represents the product data through its entire life cycle.

The Core model internal structure is composed of a set of Core Technical Capabilities (CTCs).

A CTC is constrained and defined as follows:

- a CTC shall be a consistent group of objects;
- an object is declared in one and only one CTC (i.e the CTCs do not overlap);
- a CTC is formalized as a set of objects and relations that may include references to objects in other CTCs and which are fully defined, including all attributes and constraints, and fully documented;
- objects and relations may be used to derive an implementation form;
- a CTC shall provide interfaces and may provide auxiliary information;
- a CTC shall group functionally related elements;
- a CTC may be decomposed into sub-CTCs.

EXAMPLE The Multidisciplinary Simulation CTC is decomposed into Thermal Analysis CTC and Finite Element Analysis CTC; or Product and Manufacturing Information (PMI) CTC is decomposed into Generate and Test (G&T) CTC, Dimension and Tolerance (D&T) CTC, and Weld CTC.

5.2.4.5 ARM in SysML

The ARM in SysML is an artefact that provides an interface between the modular architecture and the extended architecture. There is an AM ISO/TS 10303-400 "Reference ARM for SysML mapping"^[13] (see [6.2.2](#)). The transformation of "Reference ARM for SysML mapping" into SysML is specified in the description methods^[14] (see [6.2](#)). The result of the transformation is the ARM in SysML artefact that is contained in Core model (see [5.2.4.4](#)).

The ARM in SysML contains the application objects (AOs) to which the core objects shall be mapped. The purpose of the mapping is to ensure that each core object is traced to the AOs in an AM contained in the AM "Reference ARM for SysML mapping". Each AM has mappings from the AOs to MIM entities and STEP IR entities (see [6.7.3](#)). The ADM objects are mapped to the Core model objects, thus ensuring that each ADM object is traced to the AOs in the AM.

6 Structure of the ISO 10303 series

6.1 General

The ISO 10303 series is divided into the following sets of parts. Each set has a unique function and consists of one or more parts. The sets are listed below with their numbering scheme:

- description methods: ISO 10303-11 to ISO 10303-19 (see [6.2](#));
- implementation methods: ISO 10303-21 to ISO 10303-29 (see [6.3](#));
- conformance testing methodology and framework: ISO 10303-31 to ISO 10303-39 (see [6.11](#));
- integrated resources (IRs) (see [6.4](#)):
 - generic resources: ISO 10303-41 to ISO 10303-99 (see [6.4.2](#));
 - application resources: ISO 10303-101 to ISO 10303-199 (see [6.4.3](#));
- application protocols (APs): ISO 10303-201 to ISO 10303-299 (see [6.7](#));
- abstract test suites (ATSS): ISO 10303-301 to ISO 10303-399 (corresponding to the associated APs ISO 10303-201 to ISO 10303-299) (see [6.12](#));
- reference ARM for SysML mapping: ISO 10303-400 (corresponding to the Core model ISO/TS 10303-4000) (see [6.6](#));
- application protocol (AP) modules: ISO 10303-401 to ISO 10303-499 (corresponding to the associated APs ISO 10303-201 to ISO 10303-299) (see [6.7](#));
- application interpreted constructs (AICs): ISO 10303-501 to ISO 10303-523 (see [6.5](#));
- application modules (AMs): ISO 10303-1001 to ISO 10303-1999 (see [6.6](#));
- business object models: ISO 10303-3001 to ISO 10303-3099 (deprecated);
- Core model: ISO/TS 10303-4000 (see [6.8](#));
- application domain models (ADMs): ISO 10303-4401 to ISO 10303-4499 (corresponding to the associated application protocol (AP) modules ISO 10303-401 to ISO 10303-499) (see [6.9](#));
- usage guides: ISO 10303-5001 to ISO 10303-5999 (see [6.10](#)).

NOTE The business object models are deprecated.

6.2 Description methods

6.2.1 Purpose

The description of product data in the ISO 10303 series requires the use of formal data specification languages to ensure consistency and avoid ambiguity. The languages are intended to be computer-interpretable to facilitate the generation of application software and supporting tools. Supporting tools may be used to present the languages in a human-readable form to facilitate human understanding.

The description methods include the specification for the EXPRESS modelling language (see [6.2.2](#)) and the transformation description methods (see [6.2.3](#)).

6.2.2 The EXPRESS modelling language

6.2.2.1 EXPRESS and EXPRESS-G (ISO 10303-11)

EXPRESS is a formal modelling language that provides the mechanism for the normative description of product data for both common resources and APs. EXPRESS is specified in ISO 10303-11.

EXPRESS allows a description of the data and constraints applicable to product data in a closed domain of discourse suitable for conformance assessment. EXPRESS permits the definition of resource constructs from data elements, constraints, relationships, rules and functions. The language permits classification and structuring of resource constructs. Resource constructs shall be interpreted within APs. The interpretation capability of EXPRESS is a mechanism to facilitate the development of APs by allowing the addition of restrictions on attributes, the addition of constraints, the addition of relationships among resource constructs and application constructs, or all of the above.

EXPRESS-G is a graphical representation for the structural constructs in the EXPRESS language.

NOTE The EXPRESS-G diagrams are provided to facilitate human understanding.

6.2.2.2 The EXPRESS-X language

EXPRESS-X is specified in ISO 10303-14 and is a structural data mapping language. It allows the unambiguous specification of relationship between data that is governed by EXPRESS schemas, and for specifying alternative views of data that is governed by EXPRESS schemas.

6.2.3 Transformation description methods

The transformation description methods define the transformation between formal language constructs used in the ISO 10303 series for information modelling and used for implementation methods. The transformation definition is known as a binding.

NOTE Information loss can occur when a binding is executed.

EXAMPLE Some transformation description methods are:

- transformation from SysML XMI to XML schema^[17];
- transformation from SysML XMI to EXPRESS;
- transformation from EXPRESS to SysML XMI^[4];
- transformation from SysML XMI to web services.

6.3 Implementation methods

6.3.1 Purpose

The ISO 10303 series provides for various methods of implementation. An implementation method provides a specific method for using the APs defined in the ISO 10303 series. The exchange structure provides for the writing and reading of the description of product data in APs, using cleartext or binary encoding.

EXAMPLE ISO 10303-21 specifies an implementation method for an exchange structure.

6.3.2 Use of formal language

An implementation method is defined using a formal language so that computer-based methods can be used for the development of implementations. Formal languages provide a precise description for conformance.

6.3.3 Implementation methods for product data described using the EXPRESS language

6.3.3.1 Mapping from EXPRESS to implementation method

EXPRESS models may be used to provide the basis for all specifications of product information in the ISO 10303 series. Each implementation method based on EXPRESS defined in the ISO 10303 series specifies mapping rules from the syntax of EXPRESS onto the language used for the implementation method. The structure and syntax of each resource construct as it appears in the implementation is derived by applying the mapping rules. The rules to be applied for a specific implementation method shall be conditional on the form of the EXPRESS definition that is being mapped. Any schema defined in EXPRESS can be mapped to the implementation method.

6.3.3.2 Exchange structures for product data (implementation)

The ISO 10303 series provides different ways of exchanging product data described in EXPRESS between information systems such as:

- cleartext encoding of the exchange structure (ISO 10303-21);
- binary representation of EXPRESS-driven data (ISO/TS 10303-26);
- XML representations of EXPRESS schemas and data (ISO 10303-28).

NOTE 1 Additional implementation methods can become available after the publication of this document.

ISO 10303-21 provides a human readable form of the exchange structure. The exchange structure can be implemented to support file exchange. The language used for the syntax of the exchange structure implementation method is based on Wirth Syntax Notation.^[21] The mapping from EXPRESS to the syntax of the exchange structure is defined.

ISO/TS 10303-26 provides a mapping of EXPRESS to the binary Hierarchical Data Format Version 5^[2] (HDF5) that is optimized for the exchange of huge data sets.

NOTE 2 HDF5 is a product of The HDF Group, a spin-off from the National Center for Supercomputing Applications (NCSA) at the University of Illinois (United States) at Urbana-Champaign. The HDF software includes I/O libraries and tools for analysing, visualizing, and converting scientific data.

ISO 10303-28 enables product data described in EXPRESS to be exchanged using XML and the software tools developed to support XML technologies. It also permits product data sets so described to be incorporated into electronic commerce transactions represented in XML. ISO 10303-28 specifies the form of XML documents containing EXPRESS schemas and data governed by EXPRESS schemas. For an arbitrary EXPRESS schema, it specifies an XML schema that corresponds to the EXPRESS schema. It also contains a set of configuration directives that can be used to specify options for the structure of the XML representation of data sets that conform to EXPRESS schemas.

ISO 10303-28 specifies an XML document type definition (DTD) that corresponds to the EXPRESS schema. It can be used by implementers who need to use DTDs instead of XML schemas.

6.3.3.3 Standard data access interface (SDAI) specification

The SDAI specifies the functional characteristics of an application programming data access interface to product data. SDAI defines the operations available to an application for the purposes of accessing and manipulating data with a structure defined using EXPRESS. SDAI is specified in ISO 10303-22. The SDAI is defined independently of any programming language.

The specification of the SDAI functionality in a specific programming language is known as a language binding. A language binding provides access to, and manipulation of, data entities, types and constants, mechanisms for constraint validation and error handling. Language bindings are provided as separate parts of the ISO 10303 series.

At the time of preparation of this document, the following language bindings are available:

- C++ language binding to the SDAI (ISO 10303-23);
- C language binding to the SDAI (ISO 10303-24);
- Java^{TM1)} programming language binding to the SDAI with Internet/Intranet extensions (ISO/TS 10303-27).

Generic conformance testing for SDAI implementations is provided by ISO/TS 10303-35. For each language binding, these ATMs are converted into executable test methods for conformance testing.

6.3.3.4 EXPRESS to OMG XMI binding

ISO/TS 10303-25 specifies a mapping of EXPRESS constructs to the UML meta-model. Because the XMI standard specifies the XML representation of UML meta-model constructs, standardizing the mapping of EXPRESS constructs into UML constructs supports the XMI representation of EXPRESS schemas. EXPRESS schemas, being data specifications, are mapped into the UML meta-model concepts that appear in UML Static Structure Diagrams. ISO/TS 10303-25 does not map all EXPRESS constructs to the UML meta-model, because that meta-model does not support all the corresponding EXPRESS concepts. The specified mapping is a one-way mapping from EXPRESS into the UML Interchange Meta-model. These limitations make the mapping unsuitable for the general interchange of EXPRESS schemas and UML models for information modelling purposes.

6.4 Integrated resources (IRs)

6.4.1 Purpose

The purpose of an IR is to provide the specification of a representation of product information (see also [4.3.2.1](#)). The IRs are divided into two groups: generic resources and application resources. The IRs can reference product data descriptions written using EXPRESS from other International Standards.

6.4.2 Generic resources

The generic resources are independent of applications and can reference other resources.

6.4.3 Application resources

The application resources can reference other resources and can add other resource constructs for use by a group of similar applications.

6.5 Application interpreted construct (AIC)

6.5.1 Purpose

An AIC is part of the modular architecture. The purpose of an AIC is to provide a mechanism to identify and document the common requirements of different APs as represented within their AIMS. AICs provide a consistent and standardized interpretation of the IRs across different application contexts by identifying the semantics and shared data definitions to support a specific functionality.

NOTE The AICs are legacy parts of the ISO 10303 series.

New AICs are no longer created in the ISO 10303 series.

1) Java is the trademark of a product supplied by ORACLE. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO of the product named. Equivalent products may be used if they can be shown to lead to the same results.

6.5.2 Characteristics

AICs shall be referenced in their entirety. AICs can be referenced by APs, AMs or other AICs. In the modular architecture, typically one AIC is encapsulated in one AM.

6.6 Application modules (AMs)

6.6.1 Purpose

An AM is part of the modular architecture. AMs are the key constituent of the modular architecture. The basis of the modular approach is understanding and harmonizing the requirements, both new and those documented in existing APs, grouping the requirements into reusable modules, and using the modules in the development of an AP.

6.6.2 Characteristics

An AM contains the technical content that, under the initial architecture, was documented in an AP. The role of an AP document in the modular architecture is to provide a business context for the industrial use and implementation of the AMs that are the data specification of the AP.

There are four types of AMs: foundation modules (level 1), implementation modules (level 2), AP modules (level 3), and Reference ARM for SysML mapping (level 4).

- Foundation modules (level 1) provide lower-level reusable structures that are not likely to be implemented alone but are highly shareable and reusable.
- Implementation modules (level 2) define a capability that can be implemented and against which conformance classes can be defined. Implementation modules can include AP specific data population constraints omitted from foundation modules.
- Each AP references a single root module, which is called the "AP application module" (AP AM) (level 3). An AP application module is an implementation module, and the contents of an AP application module are the same as other implementation modules, the only documentation difference being in their name and title. The AP application module from one AP can be used by another AP. The AP application module has a part number equal to the AP's part number plus 200.

EXAMPLE The AP application module for ISO 10303-210 is ISO/TS 10303-410.

- The Reference ARM for SysML mapping (level 4) is a list of AP AMs and the resulting ARM schema. It is used for mapping from the Core model. The Reference ARM for SysML mapping has a part number of 400.

A detailed description of the content of an AM is provided in guidelines for the content of AMs.^[29] A detailed description of the content of an AP that uses AMs is provided in guidelines for the content of APs that use AMs.^[41]

6.6.3 Business benefits

The modular architecture provides:

- the ability to implement a combination of subsets of multiple APs, or to extend existing APs to meet a business need;
- the ability to reuse application software developed to support one AP in the development of an implementation of another AP with the same, or similar, requirements;
- the ability to avoid the duplication and repeated documentation of the same requirements in different APs, leading to potentially different solutions for the same requirements;
- the ability to reuse data generated by an implementation of one or more APs by an implementation of one or more different APs;

- the ability to refactor a module into two or more AMs without changing the external application;
- the ability to significantly reduce maintenance cost for APs that share common subset.

6.7 Application protocols (APs)

6.7.1 Purpose

For a given application domain, an AP specifies an information model suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases and archiving. For a monolithic AP, the information model is included in the AP document. For a modular AP, the information model is included in the AP module and is referenced by the AP document. For an extended architecture AP, the information model is specified by reference to the ADM.

6.7.2 Definition of information requirements

An AP includes the definitions of scope, context, and information requirements of an application. The definitions can specify functions, processes or information that are excluded from the application to clarify the scope, context and information requirements. The statement of scope is supported by an APDM that provides the high-level concepts, structure and relationships of the supported information model, and by an AAM that describes the processes, information flows and functional requirements of the application. The APDM is included in a normative clause of an AP. The AAM is included as an informative annex to the AP.

In the modular architecture the information requirements and constraints for the application context are defined by means of a set of AOs included in an ARM. This definition is derived from an ARM. An ARM is a formal information model that is documented in the AP.

In the extended architecture, the information requirements and constraints for the application context are defined by means of a set of application domain objects included the ADM. An ADM is a formal information model that is documented in the AP.

6.7.3 Information representation for the modular architecture

The AIM is assembled from resource constructs specified by the IRs. The resource constructs are interpreted to meet the application requirements within the defined context and scope of the AP.

A mapping from the information requirements to the AIM is provided. A mapping language is specified in guidelines for mapping specifications.^[12] The mapping defines the use, within the AIM, of resource constructs from the IRs to represent the information requirements of the application. For each ARM requirement, the mapping provides an unambiguous query of the AIM information base. A mapping from the information requirements to the AIM is provided in each AP.

6.7.4 Information representation for the extended architecture

The information model is specified by reference to the ADM and the AP application module depending on the implementation use cases. Mapping from the information requirements to the Core model is provided. The mapping language is specified in Reference [9].

6.7.5 Implementation methods

The AP is independent of implementation methods. However, an AP may include implementation method specific information in a normative annex.

When an AP includes an EXPRESS-based implementation method it shall include a reference to a table of short names that the exchange structure implementation method uses to encode AP entity names. The table is available at https://standards.iso.org/iso/10303/tech/short_names/short-names.txt

6.7.6 Conformance requirements

An AP includes the conformance requirements to be satisfied by any implementation claiming to support the requirements of the AP. Conformance requirements reflect the capabilities defined in an AP and can be specified in the description methods, implementation methods, or APs of the ISO 10303 series.

6.8 Core model

6.8.1 Purpose

The Core model is part of the extended architecture. It specifies a comprehensive information model shared by all ADMs. The Core model is intended to represent the product data through the product entire life cycle.

6.8.2 Information representation in core technical capabilities

The Core information model is decomposed or partitioned into a set of core technical capabilities (CTC). Each CTC is represented as a SysML model containing block definition diagrams and parametric diagrams using the same methodology as the ADM of an AP (see 6.9). The models are stored independent of the authoring application as XMI representations.

6.9 Application domain models (ADMs)

6.9.1 Purpose

An ADM is part of the extended architecture. It describes the information requirements and constraints of a specific application context in a normative schema definition. The ADM is a specific view of selections of the STEP integrated model constrained by the AP requirements.

6.9.2 Information representation

The ADM is represented as a SysML model containing block definition diagrams and parametric diagrams. The models are stored independent of the authoring application as XMI representations.

The items in the ADM are called the domain objects. Domain objects are defined by their attributes and their relationships with other domain objects. The domain objects are created in SysML. The mapping specification defines how each domain model element maps to one or more Core model element.

An ADM may include an AP Reference Data library that imports the Core Reference Data library.

6.10 Usage guides

6.10.1 Purpose

A usage guide contains guidance on implementing and using an AP. A usage guide may provide guidance to two different audiences: implementers and end users of AP-compliant implementations.

6.10.2 Characteristics

A usage guide can define refined context, scope, and information requirements for a subset of the context, scope, and information requirements of the subject AP, and specify the use of the AP's information representation to satisfy these requirements.

EXAMPLE 1 ISO/TS 10303-5001 defines the context, scope, and information requirements for various development phases during the design of a gear unit and specifies the ISO 10303-214²⁾ resources necessary to satisfy these requirements.

2) Withdrawn standard.

An AP may have more than one usage guide covering the same subject area, but for different disciplines.

EXAMPLE 2 ISO 10303-227 was developed jointly by the shipbuilding and process plant communities. There can be separate usage guides for ISO 10303-227 for ship piping and process plant piping, to document usage of the ISO 10303-227 information representation specific to the respective communities, or the software products they use.

6.10.3 Document structure

A usage guide can be included as an annex of an AP. One or more usage guides can be published as separate documents, in addition to or instead of such an annex.

6.10.4 Content

A usage guide may include its own information requirements and reference model, separate from the information requirements and ARM of the AP. If so, the usage guide shall include a mapping of its information requirements and reference model to those of the AP. The mapping can be to the AP's ARM or AIM, or there can be mappings to both ARM and AIM.

Example product descriptions that are supported by the AP and the corresponding AP exchange files can be included in a usage guide. If exchange files are included, the usage guide should explain the primary data structures and the logic and meaning of the values used in the exchange file.

6.11 Conformance testing methodology and framework

6.11.1 Purpose

The conformance testing methodology and framework of the ISO 10303 series provides the general methodology and requirements for the process of testing the conformance of a product that claims to implement an ISO 10303 series AP part. The goal of the methodology and framework of conformance testing is to ensure:

- repeatability: test results are consistent whenever undertaken;
- comparability: test results are consistent wherever undertaken;
- auditability: test procedures can be confirmed as having been correctly undertaken, following the testing, by review of record.

ISO 10303-31 provides a framework and describes the general concepts for conformance testing of implementations of the ISO 10303 series.

6.11.2 Procedures for conformance testing

Conformance testing of an AP can be performed by applying the ATM for the chosen implementation method to the test cases in the ATS. The procedures for conformance testing are independent of the implementation under test.

If a single implementation combines several APs, conformance testing is performed for each AP separately.

The ISO 10303 series includes parts that define the roles and responsibilities of the testing laboratory and the client submitting an implementation for conformance testing.

6.11.3 Abstract test methods (ATMs)

The ISO 10303 series includes parts that specify an ATM for each implementation method. The ATM describes how an implementation of an implementation method is tested, independent of:

- the specific implementation;
- the testing tools and procedures;

- the specific AP under test.

6.12 Abstract test suites (ATs)

An ATs contains the set of abstract test cases (ATCs) for an AP to support the conformance requirements. Each ATC provides an implementation-independent specification of the actions required to evaluate part of one or more conformance requirements.

Each conformance requirement corresponds to one or more ATCs, designed to satisfy one or more test purposes. For each ATC, verdict criteria are generated from the conformance requirements to allow a testing laboratory to assess the conformance of an implementation with respect to that test case. When a conformance test based on an ATC, the resulting verdict indicates if the implementation meets one or more conformance requirements.

7 Information object registration scheme

To provide unambiguous identification of schemas and other information objects in an open information system, the ISO 10303 series shall employ the registration technique defined in ISO/IEC 8824-1.

Unambiguous identification of this document is defined in [Annex A](#).

This technique identifies objects by their assignment to a tree structure whose root is "ISO". Each node in the tree is identified by a sequence of integers corresponding to the index of the leaf under each node. Thus, "ISO nnnnn" is identified by the object identifier:

$$\{ 1 0 \text{ nnnnn} \}$$

EXAMPLE 1 ISO/TS 10303-442 MIM long form is identified as follows { 1 0 10303 442 3 1 4 }

The elements have the following meaning:

- the initial 1 indicates ISO;
- the 0 following it identifies the object as a standard;
- the number following that is the number of the standard.

ISO/IEC 8824-1 also defines identifiers to stand in the place of these numbers; "ISO" has the value 1 and "standard" has the value 0. For multi-part standards, the next number is required to be the part number.

For the purposes of identifying information objects unambiguously within an open information system, standards produced by ISO/TC 184/SC 4 adopt the convention that the value for ISO and the value for standard shall not be included in the text of the standard for clarity. By convention, the value following the part number shall be the version number. By convention, the value of the version number of the first edition shall be 1. A draft document shall not be given its own unique version number. Thus, this document is identified by the object identifier:

$$\{ \text{iso standard 10303 part(1) version(3)} \}$$

Here, the values for ISO and standard are not displayed; the value of the part number is given explicitly, but the notation allows us to associate a term with this value, thereby providing some semantics. The notation for values of this type is defined in ISO/IEC 8824-1, and the predefined assignments are specified in ISO/IEC 8824-1.

NOTE In the past, the value 0 for the version, was used to refer to documents at the Draft International Standard (DIS) stage. This practice is no longer permitted. Instead, the value given for the version number of a draft document is the version number that the document will have upon publication.

EXAMPLE 2 ISO 10303-203³⁾ edition 1 was published as an international standard in 1994. Technical corrigendum (TC) 1 to edition 1 was published in 1996, and TC 2 was published in 1998. Amendment 1 was published in 2000, and TC 3 was published in 2004. A new, modular version of the AP, ISO/TS 10303-203 edition 1, was published in 2005. The modular version was promoted to an International Standard, ISO 10303-203 edition 2, in 2011. The version numbers under this scenario would be:

- version 1: ISO 10303-203:1994;
- version 2: ISO 10303-203:1994/Cor 1:1996;
- version 3: ISO 10303-203:1994/Cor 2:1998;
- version 4: ISO 10303-203:1994/Amd 1:2000;
- version 5: ISO 10303-203:1994/Cor 3:2004;
- version 6: ISO/TS 10303-203:2005;
- version 7: ISO 10303-203:2011.

For the purposes of identifying information objects unambiguously within an open information system, ISO/TC 184/SC 4 adopts the following conventions:

- The value following the version number is used to identify the type of information object defined within the part. One of the values defined in the [Table 1](#) shall be used:

Table 1 — Information object type assignment

Type	Object identified
1	EXPRESS schema ^[15]
2	XML schema definition ^[47]
3	XML metadata interchange schema (XMI) ^[4]
4	XLSX (Office Open XML Spreadsheet) ^[34]
5	JSON schema ^[35]
6	OWL ^[36]
7	Schematron ^[37]

The type of information object shall be "object" for the IRs and "schema" for all other type of documents.

- The value following the object type is an integer that identifies the instance of the object type so identified. Where applicable (AM for modular AP and Monolithic AP), one of the values defined in the [Table 2](#) shall be used:

Table 2 — Type assignment for monolithic AP and modular AP application modules

Type	Object identified
1	ARM short form
2	AIM short form (for monolithic AP); MIM short form (for modular AP)
3	ARM long form
4	AIM long form (for monolithic AP); MIM long form (for modular AP);

3) Withdrawn standard.