INTERNATIONAL STANDARD

ISO 29481-1

First edition
2010-05-15

Building information modelling — Information delivery manual —

Part 1: Methodology and format

Modèles des informations de la construction — Contrat d'interchange —

Partie 1: Méthodologie et format

Reference number
ISO 29481-1:2010(E)

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

Published in Switzerland

Contents

Page

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 29481-1 was prepared by Technical Committee ISO/TC 59, *Building construction*, Subcommittee SC 13, *Organization of information about construction works*.

ISO 29481 consists of the following parts, under the general title *Building information modelling — Information delivery manual*:

— *Part 1: Methodology and format*

The following part is planned:

— *Part 2: Management communication*

Introduction

Building information modelling provides a concept for describing and displaying information required in the design, construction and operation of constructed facilities. It can bring together the diverse sets of information used in construction into a common information environment - reducing, and often eliminating, the need for the many types of paper documentation currently in use.

An information delivery manual (IDM) provides significant help in getting the full benefit from a building construction information model (BIM). If the information required is available when it is needed and the quality of information is satisfactory, the construction process itself will be greatly improved.

For this to happen, there should be a common understanding of the building processes and of the information that is needed for and results from their execution.

This part of ISO 29481 sets out a methodology and format for the provision of an integrated reference for the processes and data required by a BIM. It describes how to identify and describe the processes undertaken within construction, and the information required for their execution and the results. This part of ISO 29481 also describes how this information can be further detailed to support solutions provided by building-information-system providers in a form that enables its reuse and how it can be configured to meet national, local and project needs.

In doing so, this part of ISO 29481 provides a basis for reliable information exchange/sharing for users so that they can be confident that the information they are receiving is accurate and sufficient for the activities they need to perform. The development of this part of ISO 29481 has been driven by the need of users for reliability in information exchange.

Examples and guidelines related to the development of IDMs will be published at: <http://www.standard.no/IDM>. The site will developed and maintained by the ISO/TC 59/SC 13 secretariat.

Building information modelling — Information delivery manual —

Part 1: Methodology and format

1 Scope

This part of ISO 29481 specifies a methodology and format for the development of an information delivery manual (IDM).

This part of ISO 29481 specifies

- a methodology that unites the flow of construction processes with the specification of the information required by this flow,
- a form in which the information should be specified, and
- an appropriate way to map and describe the information processes within a construction life cycle.

This part of ISO 29481 is intended to facilitate interoperability between software applications used in the construction process, to promote digital collaboration between actors in the construction process and to provide a basis for accurate, reliable, repeatable and high-quality information exchange.

2 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

2.1
actor
person, organization or organizational unit (such as a department, team, etc.) involved in a construction process

2.2
building construction information model
BIM
shared digital representation of physical and functional characteristics of any built object (including buildings, bridges, roads, etc.) which forms a reliable basis for decisions

NOTE “Building information model” is frequently used as a synonym for BIM.

2.3
building information system
system used to create, maintain, disclose or expire elements of a building information model

NOTE The components of the system can include actors, hardware (servers, clients, peers) and software solutions.

2.4
business process modelling notation
BPMN
notation for use in the development of business process diagrams that is designed to be readily understandable by all business users

2.5
business requirement
requirement that describes in business terms what needs to be delivered or accomplished

2.6
business rule
statement that formally defines or constrains some aspect of the business, a rule under which an organization operates or a policy or decision that influences a process

2.7
exchange requirement
ER
set of information that needs to be exchanged to support a particular business requirement at a particular process phase (or phases) / stage (or stages)

NOTE Information delivery requirement can be used as a synonym for exchange requirement.

2.8
exchange requirement model
ERM
technical expression of an exchange requirement expressed as a schema

NOTE An exchange requirement model describes the binding of an exchange requirement to a particular standard information schema and version.

2.9
functional part
FP
unit of information within an exchange requirement that may be fully specified in its own right

2.10
interaction map
representation of the roles and transactions relevant for a defined purpose

2.11
management communication
sharing information for a management purpose

2.12
model
representation of a system that allows for investigation of the properties of the system

NOTE "Representation" is defined in http://www.businessdictionary.com/definition/representation.html.

2.13
process map
PM
representation of the relevant characteristics of a process for a defined purpose

2.14
role
functions being performed by an actor at a point in time

NOTE The role of an actor is determined by action and outcome and not by the profession or trade followed by the actor.

2.15
schema
schema is a description of the formal structure of a defined set of information

2.16
transaction
communication event that fulfils a relationship between two roles

3 IDM (Information delivery manual)

3.1 Complete schema

A complete information schema that covers all of the information required for all actors throughout the construction process will be large and comprehensive. Such a schema is relevant in defining all of the project information needs for all business requirements at all life-cycle stages (see Figure 1), but it is not the way that project information is usually delivered.

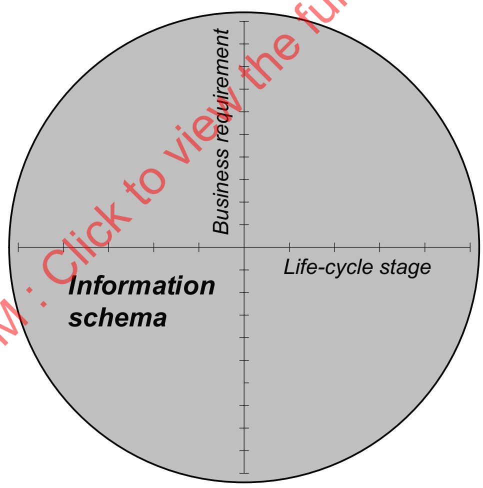

Figure 1 — The information schema supports all business requirements at all life-cycle stages

3.2 Breaking a complete schema to support requirements

It is more usual for information to be exchanged about a particular topic and the level of detail provided to be driven by the life-cycle stage. The need is (generally) to support one business requirement over one or more life-cycle stages (see Figure 2). This is a matter of deciding which components of the information schema should be used to meet requirements.

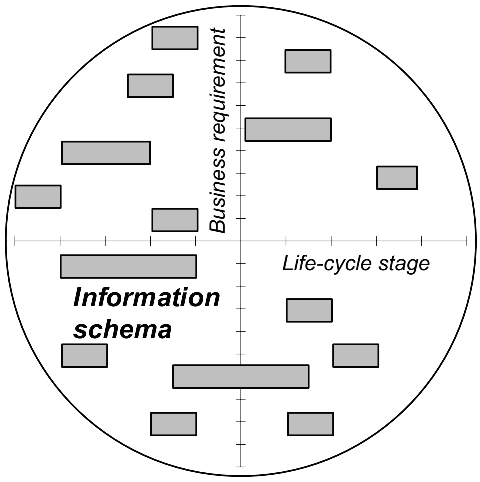

Figure 2 — Need to support a business requirement at a life-cycle stage

3.3 Supporting the building information modelling

Elements of the overall information schema are used in a building construction information model (BIM) (see Figure 3). For a particular business requirement, only certain classes of information are required. Multiple objects are derived from each class, each object having an identity (determined by a unique identifier) and a state (determined by the values given to each attribute of the object). The classes that support the business requirement form a unique and identifiable schema.

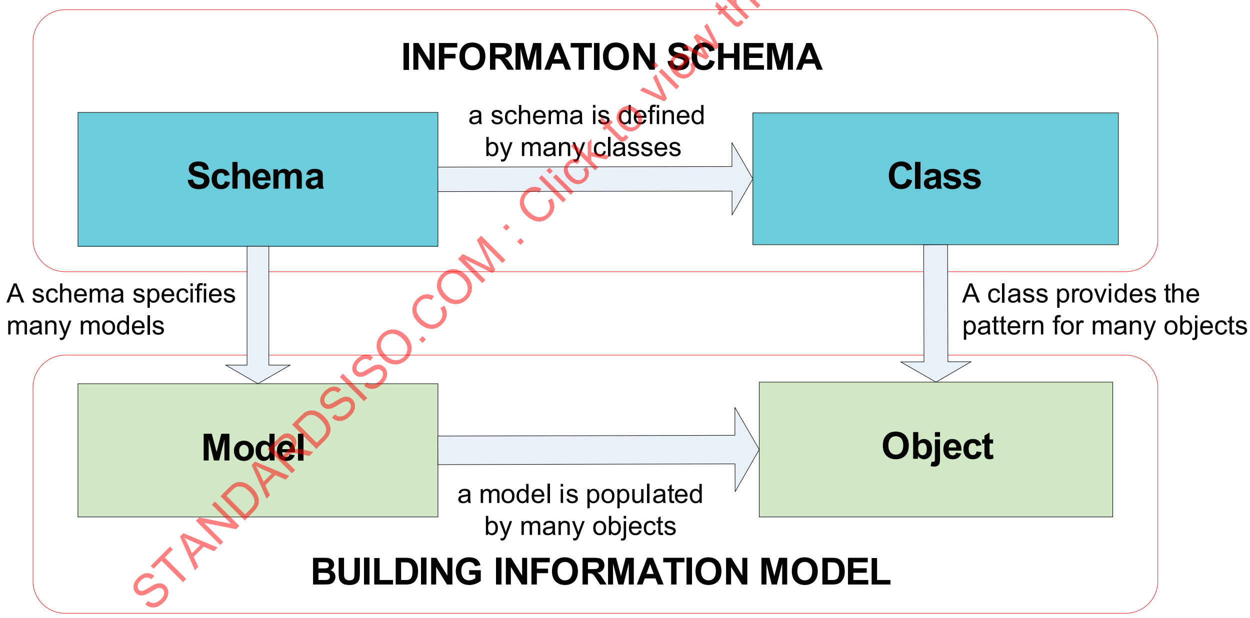

Figure 3 — Supporting the building information modelling

3.4 Supporting the business requirement

To do this means that the set of information that needs to be exchanged to support a particular business requirement in the relevant life-cycle stages must be established. This is termed an exchange requirement.

An exchange requirement provides a description of the information to be exchanged in non technical terms. An exchange requirement may support the communication of object information enabling the construction and operation of a project or it may support the communication of management information that controls the project execution.

3.5 Supporting the software solution

The technical content required by solution providers to support an exchange requirement is provided as a series of units of information. A unit of information is termed a functional part.

A functional part provides the technical expression of information content as a subset of the complete information schema.

3.6 Supporting the construction process

Software solutions typically support users across several exchange requirements. Exchange requirements are used to support an overall construction process. The connection between exchange requirements and a construction process is captured within a process map.

A process map typically deals with the development of information within the boundary of a particular topic or software view. It shows the roles of actors engaged in the process and references the transactions between them.

3.7 Defining the connection between IDM components

Functional parts are used together to create exchange requirement models. An exchange requirement model provides a version of the exchange requirement that can be understood by a computer. It includes business rules which are computer interpretable versions of the business propositions described in an exchange requirement.

3.8 Content in the specific IDM

The content in a specific IDM will

— describe the need for information exchange between processes,

— specify how to capture the information needing to be exchanged between these processes,

— identify the actors sending and receiving information,

— define, specify and describe the information being exchanged to satisfy the requirements at each point of the business process,

— ensure that definitions, specifications and descriptions are provided in a form that is useful and easily understood,

— create detailed specifications of the information captured within exchange requirements to facilitate the development of software building information systems,

— ensure that the information specifications can be made relevant to local working practices.

3.9 Users of this part of ISO 29481

The main purpose of this part of ISO 29481 is to provide guidance for those who develop specific IDMs. Thus, the main users are expected to be the IDM developers who create process maps, exchange requirements, functional parts, exchange requirement models and business rules using knowledge elicited from end users and solution providers.

Other actors will mainly be using the specific IDMs which are developed by using this part of ISO 29481. In addition, some users of specific IDMs might identify needs for new IDMs and thus become users of this part of ISO 29481. These users include

— professional IDM-developers and solution providers — according to very technical specifications,

— information users, i.e. executive users and end users concerned with producing the content of the IDMs and benefiting from the result.

4 IDM framework

4.1 Overview

Figure 4 provides a generalized view of the main components used in an IDM and how they relate to each other. The organization of the components within the framework is based on two ideas:

a) the components at the top layer of the framework relate to processes, progress through data specifications in the middle layers and include application software elements at the bottom layer;

b) similarly, the components relate to industry practitioners at the top layer of the framework and to ICT analysts and programmers at the bottom layers.

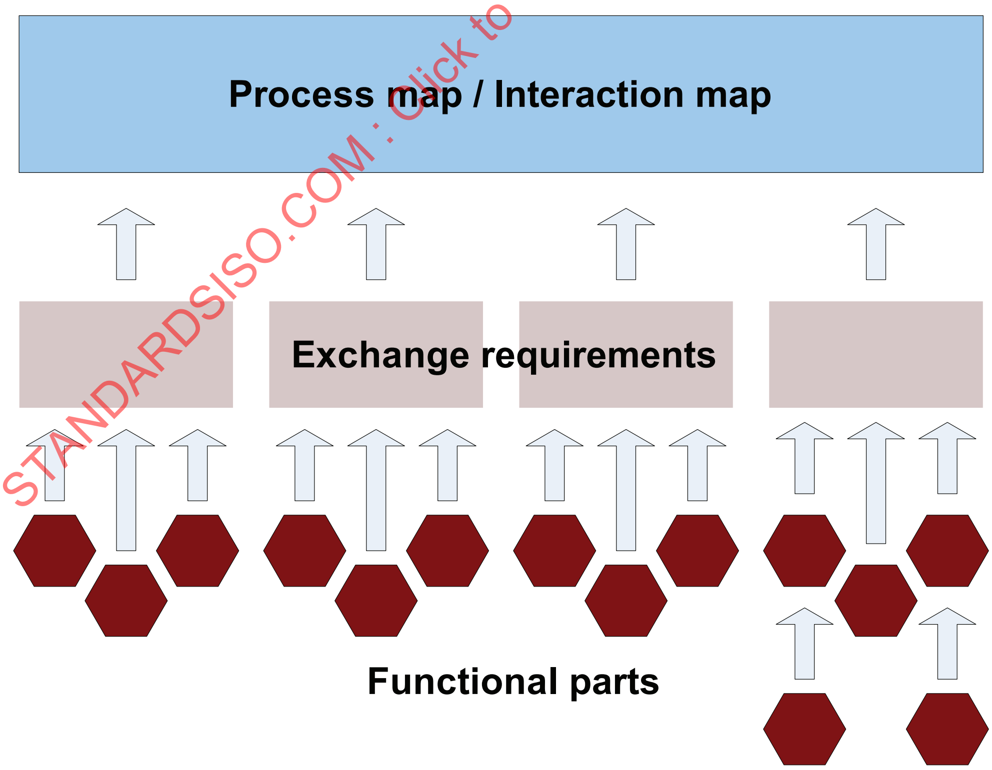

Figure 4 — IDM basic framework

4.2 IDM component header information

Each IDM component described below includes a set of administrative information that enables it, its current author and its change history to be captured. Administrative information includes

— a name or title conforming to the naming rules given in this International Standard,

— a unique identifier,

— a change log that identifies creation or change made together with the author and date.

4.3 Description of the use case

Each IDM component shall start with a short plain language description of the use case the IDM is intended to solve or about which particular topic or business requirement information shall be exchanged.

4.4 Interaction maps

The purpose of an interaction map is to identify the relevant roles and transactions for a specific purpose. IDM draws a distinction between the role “initiator” (makes a request) and the role “executor" (effectuates that request). If there is such a relationship between two roles, it is termed a “transaction”.

An interaction map identifies the relevant roles, transactions and initiator – executor relations.

A transaction contains a set of exchange requirements that are exchanged for a particular purpose. The transaction also stipulates the participating roles, point in the life cycle and the sequence in which exchange requirements should be delivered (if appropriate). A message is a populated information model and contains data. Attachments may be linked to messages.

In an interaction map, all transactions needed for the handling of required contributions of relevant roles to the BIM shall be included. All transactions within the interaction map have a unique identity and name.

Using transactions, the business cooperation and communication requirements are defined. Use of exchange requirements (ER) is optional in transactions.

Using transactions, the contributions of relevant roles to the BIM can be controlled. For that purpose, in specific transactions, the following components can be added as attachments to specific messages

— exchange requirement,

— exchange requirement model,

— window of authorization: in the context of a transaction an executive role (executor) can access the building information system. The window of authorization describes what information in this transaction by the role may be read or changed.

4.5 Process maps

4.5.1 General information

The purpose of a process map is to describe the flow of activities within the boundary of a particular topic, the roles played by the actors involved, together with the information required, consumed and produced.

For representing process maps, the approach recommended is the BPMN. (Further information on BPMN is given in A.4.)

Within IDM, a process map

- sets the boundary for the extent of the information contained within the process,
- establishes the activities within the process, and
- shows the logical sequence of the activities.

The actual information that is within the process boundary is determined by the contents of the exchange requirements specified in the process.

4.5.2 Content

All activities described within a process map should be related to the defined life-cycle stages as they appear in exchange requirements' documentation.

A process map includes the following administrative information

- the exchange requirements that are within the boundary of the process,
- an overview that provides a comprehensive description of the overall process. Illustrations may be used to illustrate particular points within the overview.

4.5.3 Specification of processes

In a process map, all of the diagrams and sub-diagrams created for describing the process shall be included. Each process within the process map has a unique identity and name.

Each process within the process map is described in such detail as required. The aim is to describe the purpose of the process to a reader.

4.5.4 Specification of data objects

A data object is a named collection of data. It may be a collection of data available from an external source (e.g. library data) or it may be the data exported from an activity to enable other activities to occur (e.g. an exchange requirement).

Data objects that are not exchange requirements shall have a name that is indicative of their purpose and a description that outlines their purpose and content.

4.5.5 Specification of exchange requirements

An exchange requirement is a particular type of data object within a process map that is located within the information model role.

Exchange requirements shall have the following

- a name that is indicative of the purpose (naming rules are given in Clause A.3),
- a description that outlines the purpose and content.

NOTE The description provided for an exchange requirement is expected to be more detailed than that given for a general data object. The description can be reused as the overview description within the exchange requirement documentation (described below).

4.5.6 Specification of coordination point gateways

A coordination point gateway is a named point within a process map at which the information from exchange requirements is brought together to enable coordinated decision making to occur. Each coordination point gateway shall have a name and its intended purpose should be described.

Decisions made at coordination point gateways may provide

— a hard gateway at which all information shall be valid according to the exchange requirements and without which further progress is not allowed,

— a soft gateway at which information may not be fully valid according to the exchange requirements but at which progress is allowed on the expectation that full validity will be provided later.

4.6 Exchange requirements

4.6.1 General information

An exchange requirement is a description of a set of information that needs to be exchanged to support a particular business requirement at a particular stage of a project. It is intended to provide a description of the information in non technical terms. The principal audience for an exchange requirement is the end user (architect, engineer, constructor, etc.). It should however also be used by the solution provider since it provides the key to the technical detail that enables the solution to be provided.

An exchange requirement represents the connection between process and data. It describes a set of information from a process that has been performed by an actor in the role of initiator to enable a downstream process to be performed by another actor in the role of executor.

4.6.2 Content

An exchange requirement contains the following information:

— the life-cycle stage(s) for which the exchange requirement is used; an exchange requirement may be applicable to one or more life-cycle stages,

— an overview that states the aims and content of the exchange requirement using terminology that is familiar to the user. The aim is that it should be understood by a person who needs to be aware of what the exchange requirement is intended to achieve but who does not need to know the detail of how it is achieved. Illustrations may be used to amplify particular points within the overview.

4.6.3 Information units

The information required is provided in a set of information units. An information unit typically deals with one type of information or concept of interest such as the overall project, walls, windows, etc.

Preconditions for the exchange requirement are identified first. A precondition is an exchange requirement that should have been completed prior to the execution of the current exchange requirement.

Information units are then broken down further to provide the following:

— an identifying name,

— a description about the information exchanged,

— the identity of the functional part within which the detailed technical content of this information unit is described,

— the information which needs to be exchanged for the provisions of this exchange requirement to be satisfied. This should include any special provisions, propositions or rules related to the information.

4.7 Functional parts

4.7.1 General information

A functional part is a description of a unit of information used by solution providers to support an exchange requirement. It describes the information in terms of the required capabilities of the information model upon which it is based. A functional part is fully described as an information model in its own right as well as being a subset of the information model on which it is based.

A functional part focuses on the individual actions that are carried out within a business process. An action is concerned with a particular unit of information within an exchange requirement. For instance, to exchange a building model, it is first necessary to model the walls, windows, doors, slab, roof, etc.

Each functional part provides a detailed specification of the information that should be exchanged as a result of the action. This is both in terms of user description and also binding to a particular information schema or version. It may participate in several exchange requirements. Therefore, functional parts are designed to be reusable within many exchange requirements.

NOTE IDM is in principle independent of specific data schema such as IFC 2×3, IFC 2×4 or versions of XML. In IFC 2×, a functional part (FP) refers to an entity (object) or an attribute.

4.7.2 Content

A functional part contains an overview that states the aims and content of the functional part in non technical text form. The intention is that, whilst a functional part is primarily intended for solution providers, a user should still have an awareness of the content since they will be using it in conjunction with an exchange requirement.

NOTE The description of an information unit within an exchange requirement may be derived from the overview of the corresponding functional part.

4.7.3 Technical information

A functional part provides for a detailed breakdown of technical information. It describes in detail the entities and properties required and how they are to be configured.

The technical information section is developed on the basis of a flow of events. That is, it also establishes a reasonable sequence in which the entities and properties may be defined. Table 1 provides a list of what the given information includes.

Table 1 — Technical information in functional parts

Technical information	Description
Description	A detailed description of the information which needs to be asserted within the functional part. Each individual data item is described in approximate sequence. An individual data item here is considered to be an object and attribute or a property set and property.
Entity or property set or art	A specification of the entity/attribute or property set/property combination that fulfils the information described or a reference to a functional part that provides results into that currently considered. Attributes and properties should also identify the datatype in which they will be expressed. The convention defined within IDM for the expression of object/attribute/datatype and property set/property/datatype is — Functional PObject.Attribute → Datatype, — PropertySet.Property → Datatype.
Mandatory/ Optional/ Required/ Excluded	An indication of whether, for the functional part, the information is mandatory (shall be provided), optional (may be provided but is not mandatory), required (optional within the information model but considered to be mandatory for this functional part) or should not be asserted for this functional part.

4.7.4 List section

The list section of the functional part provides lists of the names of the various components used. What these include is described in Table 2.

Table 2 — List of components

Component	Description
Entities	The entities of interest within the current part
Datatypes (defined, enumeration and select)	Named types of data that may be used including labels, text descriptions, identifiers, enumerated ranges of possible values from which a selection should be made and select types for alternative routing through a schema
Function	Extended rules forming part of a schema that may be processed to validate data (such as determining that a particular date is within the legal range of possible values for that month and year)
Property sets	Those property sets that are relevant to the current functional part
Functional parts	Other functional parts whose services are used

4.7.5 Schema section

The schema section provides the formal description of the functional part as a subset of the information model from which it is derived, using the form and notation of the parent information model.

This section is specifically for solution providers to guide their implementation development.

4.7.6 Example section

The example section includes particular examples that show how the functional part might be used. It is useful to provide more detailed guidance to implementers and may be used by them for preliminary testing to ensure that they are returning the correct results from their solutions.

4.8 Units of functionality

A unit of functionality (UoF) is a collection of application entities, relationships and property sets that defines one or more concepts within a functional part or exchange requirement such that the removal of any unit would render the concepts incomplete or ambiguous.

A unit of functionality can be used to capture the basic functionalities within a schema such as naming, identification, etc. (See Figure 5.)

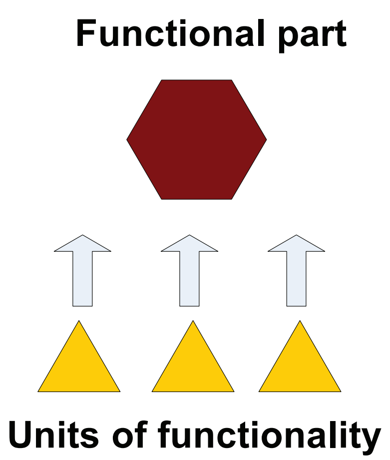

Figure 5 — Units of functionality within a functional part

5 Implementing and validating IDM components

5.1 Exchange requirement model

An exchange requirement model is the technical solution of an exchange requirement. It is created from the set of functional parts defining the units of information that support the underlying exchange requirement. (See Figure 6.)

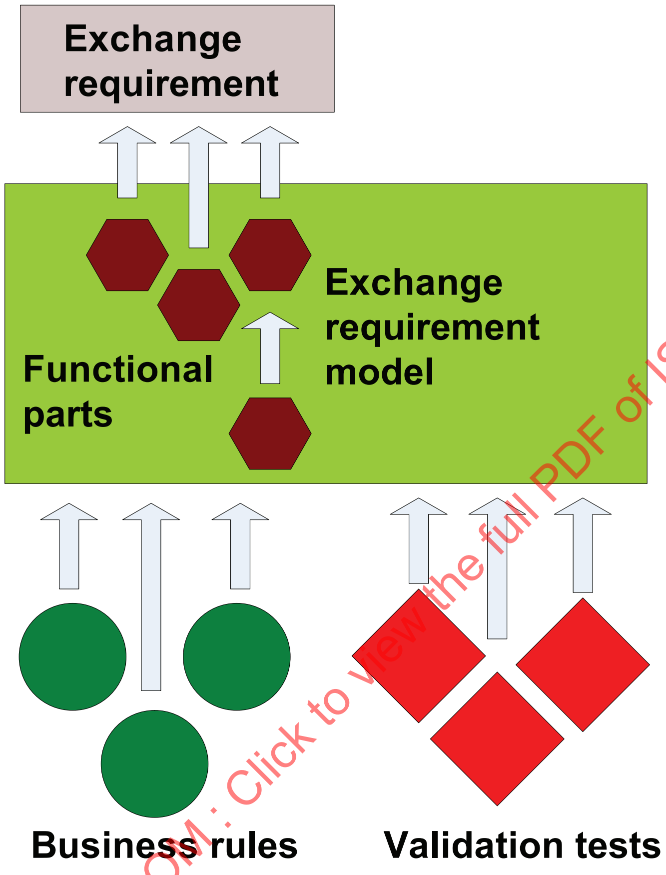

Figure 6 — Defining an exchange requirement model

Note that there is a 1:1 correlation between an exchange requirement model and an exchange requirement. Whilst the expression of an exchange requirement is completely independent of any schema or any particular version of a schema, an exchange requirement model is schema dependent, due to the fact that it is created from schema dependent functional parts.

Exchange requirement models are particularly significant as they are the IDM components that

— will be supported within software applications,

— form part of the model view definition that is certified,

— are the components to which business rules are applied,

— are the components against which validation tests can be applied.

See Clause 8 for information on how exchange requirement models are compiled from functional parts.

See Clause A.5 for information on model view definitions.

5.2 Business rules

Business rules describe operations, definitions and constraints that may be applied to a set of data used within a particular construction process. They enable controls to be applied to

- use of specific entities,
- attributes and properties that shall be asserted (or not asserted),
- values, ranges of values or value limits that should be observed,
- dependencies between entities or attributes or attribute values.

Business rules can be used to vary the result of using an information schema without having to change the information schema itself. This provides the schema with agility so that, through the application of different sets of business rules to the same information schema, different local applications can be defined.

Note that it is possible to add to, amend or even delete business rules without affecting the underlying information schema.

Business rules should be expressed as formal propositions in terms of their actions on exchange requirements. However, they should be expressed in an appropriate coded form for specific actions on the functional parts that are contained within an exchange requirement.

An example of a business rule expressed as a proposition is the requirement that the area of a space whose type is “executive office” shall be greater than or equal to 10 m^2. In this form, it is applicable to the exchange requirement. When applied to a functional part, this is coded in the logical form appropriate to the manner in which the attribute or property is expressed.

Each set of business rules is applied to an exchange requirement model. That is, the assertion of values for attributes or properties that are derived from the functional parts can be controlled by a set of business rules.

However, where a particular functional part is used within a different exchange requirement model, a different set of business rules should be applied. This can be seen from Figure 7 where a single functional part may be used in two separate exchange requirement models. Each exchange requirement model has a separate set of business rules that may be applied to the functional part contents.

Thus, an identified set of business rules applies to a single exchange requirement model.

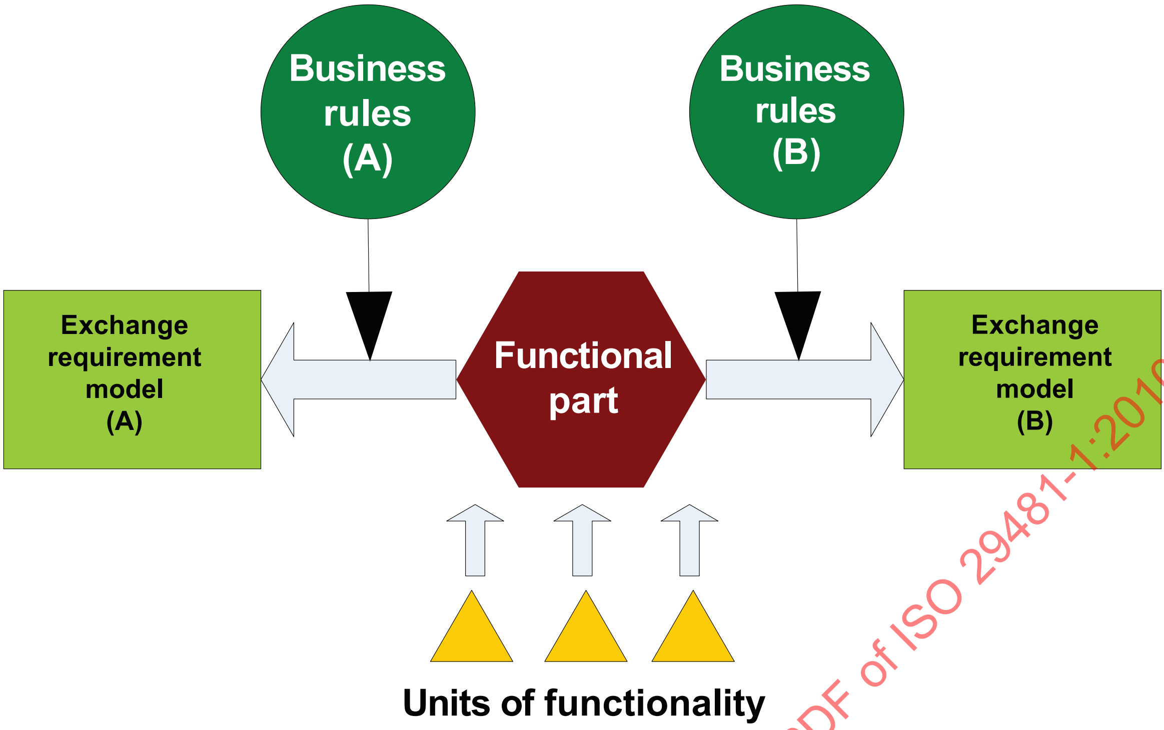

Figure 7 — Applying business rules

Business rules are collected together into sets where each set is applicable to a particular understood level of application. For instance, a global rule set may be applicable to all functional parts used in IDM on a global basis. However, a local rule set for use in a particular place (e.g. Norway) will only be applicable to exchange requirement models used in that place.

Each business rule defined should have a unique identification that indicates

— where it was defined or who it was defined by,

— a level of applicability, e.g. is it applicable to an entity, an enumerated data type or for a property within a property set,

— an index number or other sequential reference.

It is recommended that unique rule identifiers draw on local or global classification resources to guide the numbering.

Each business rule should also have a unique name that provides a short indicator of its purpose. This is the name by which the rule will generally be addressed.

Each business rule should include a complete formal proposition of its purpose. This will provide the logical statement against which a coded form of the rule can be developed. Since the rule is expressed here as a proposition, the intent is that any rule or constraint language that is appropriate may be used.

NOTE For example, when using IFC-based information exchange, it is probable that business rules will be expressed according to the provisions of the IfcConstraintResource schema.

5.3 Validation tests

Validation tests are tests carried out on the information exported from a software application according to the schema of an exchange requirement model. They are used to ensure that a stated exchange requirement is being satisfied according to a set of applied business rules. (See Figure 8.)

Validation tests should be carried out using test files that have a known performance and that are specifically designed to validate particular aspects of the exchange requirement model.

The values assigned to attributes and properties within a test file may vary between locations in which validation tests are carried out. This is because different sets of business rules may be applied to the same exchange requirement model in different places.

Validation tests are applied for the purposes of

— verifying that the export of information from a business information system meets the quality criteria set out in an exchange requirement,

— improving the quality of business information system solutions,

— providing metrics against which claims made for business information system performance can be verified,

— making comparisons between business information systems fulfilling the same objectives (when compared using the same tests).

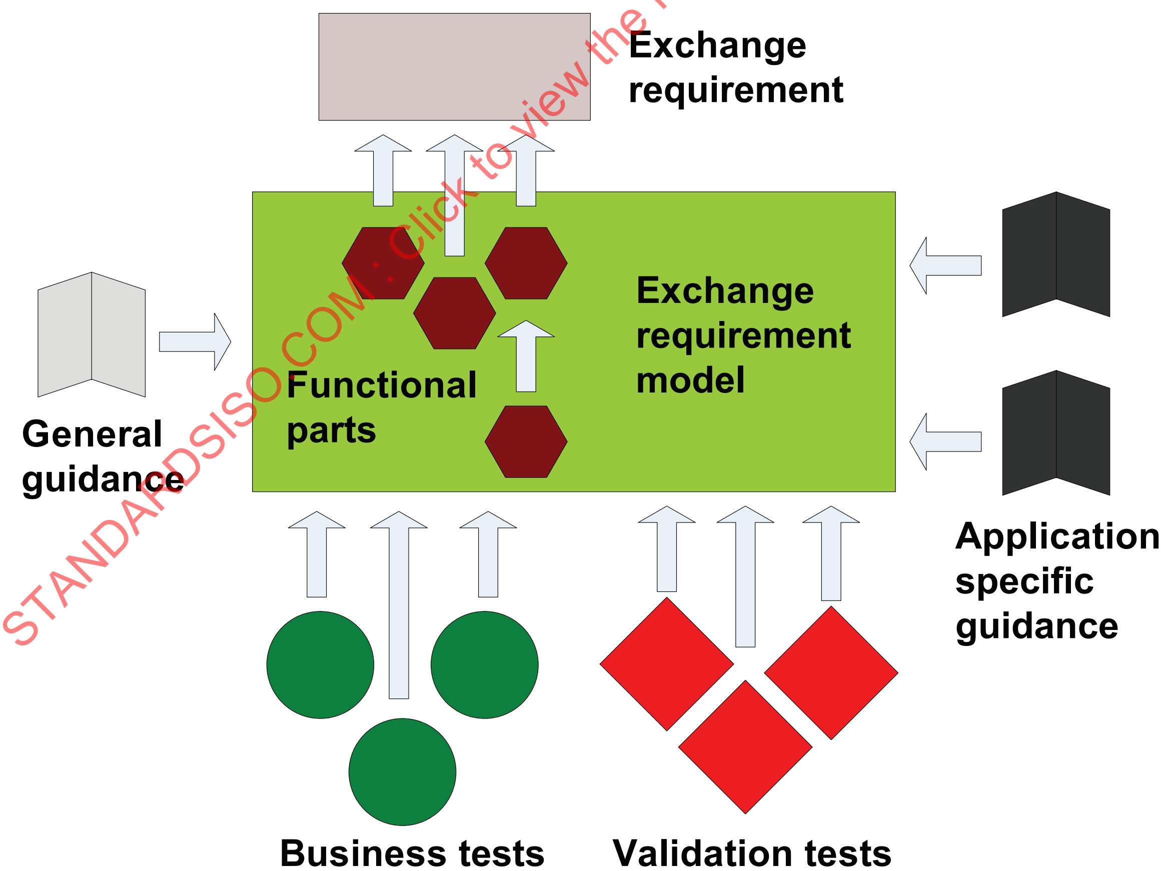

Figure 8 — Providing general and application specific guidance

6 General and application guidance

6.1 General guidance

General guidance is guidance provided to users about the extent and quality of the information that they will need to prepare within a BIM for export according to the provisions of an exchange requirement. It provides a preliminary statement of quality which, in conjunction with the application specific guidance, can be used to define the quality objectives against which the performance of BIM users can be tested.

6.2 Application specific guidance

Application specific guidance is particular guidance provided by building information system providers or by other third party information providers that describes how a software application meets the business needs expressed by an exchange requirement. This may also include guidance on how to use the software in the circumstance of the exchange requirement and may also describe how the results are presented and applied.

Note that application software guidance is not provided as part of the IDM but is included within the technical architecture as an important and related provision.

7 IDM development process

7.1 Propose an IDM development

A proposal to undertake an IDM development is a preliminary stage that sets the scene for work to be done. It is concerned with

— defining the scope,

— setting the development approach,

— identifying resources,

— establishing a project plan.

7.1.1 Define scope

The scope should set the boundaries for the work that is to be done and provide a continuing reference to ensure that the work boundaries do not grow beyond a point at which the planned or available resource ceases to be sufficient.

7.1.2 Establish the development approach

The development approach selected will be determined by the extent of information, software or other exchange requirements available. The approaches are described under 7.2 below.

7.1.3 Identify resources

Resources are the people who need to participate in an IDM development. Resources need to be properly balanced between project management, development of IDM components and industry knowledge to guide both component development and provision of software solutions. The balance of resources required will be affected by the development route that is selected.

7.1.4 Project plan

The project plan sets the period over which the development is to occur, determines the tasks to be undertaken, assigns the available resources and sets the deliverables required.

7.2 Undertake an IDM development

7.2.1 General information

There are three approaches to IDM development:

— process discovery;

— business rule customized;

— reverse engineering.

7.2.2 Process discovery

7.2.2.1 General information

Process discovery is the conventional approach used in IDM development. It assumes that there is no existing software from which requirements may be mined or other exchange requirements that may be customized.

The development approach is described below as a linear sequence. In practice, feedback between development stages and cyclic developments can be expected.

7.2.2.2 Discover process

This involves working primarily with industry experts and specialist building information system providers to determine the extent of the construction processes that are within the defined scope. The result is a process map.

Process mapping usually requires several cycles of development and review to achieve a satisfactory conclusion. On completion, the process map may represent the construction process as currently practiced or it may represent a proposal for an improved construction process. Whether to create an 'as-is' process or a 'to-be' process is a decision that needs to be taken as part of the development.

The process maps will identify the 'packages' of data that need to be exchanged at various points in the construction process. These are the exchange requirements.

7.2.2.3 Create exchange requirement

Exchange requirements should then be created. Wherever possible, they should re-use the services of existing functional parts. Only when absolutely necessary should new functional parts be created.

7.2.2.4 Create functional parts

Where new functional parts are needed, they should be created to provide added support to exchange requirements.

7.2.3 Business rule customization

7.2.3.1 Define business rules

The set of business rules that may need to be applied to further configure the exchange requirement should be defined. These may be used to control properties to be asserted or values that can be assigned.

7.2.3.2 Business rule localization

Business rule localization assumes that an exchange requirement exists for the purpose required but that it does not meet the needs for use within a particular location. Location may be a place (country, region, etc.), a project or a framework of working agreed between organizations.

Business rule localization also assumes that the process map within which the exchange requirement is defined exists and that all functional parts that support the exchange requirement are defined.

Business rules are applied to individual units of functionality within a functional part within the context of the exchange requirement model to make it applicable at a particular location. Note that the same functional part may have different business rules applied in the context of a different exchange requirement or for a different location.

7.2.4 Reverse engineering

7.2.4.1 General information

Reverse engineering assumes that software already exists that is capable of dealing with the information exchange(s) required but that there is a need to specifically capture from that software the exchange requirements that it can support.

The most appropriate way of doing reverse engineering is to define the scenario that the exchange requirement is to support and then work through the software application to provide the data.

7.2.4.2 Define scenario

Define the scenario that the exchange requirement is to support. This should provide a detailed textual description that can be used as the overview description for the exchange requirement.

7.2.4.3 Recover data

Working through the defined scenario within the software application, recover all of the data that needs to be specified to achieve a result within the application.

For each data item recovered, determine if it could be acquired from an upstream software application. If so, it should form part of the exchange requirement.

7.2.4.4 Create exchange requirement

Create the exchange requirement using the defined scenario as the overview and the identified data within the technical sections. Check data items to determine if they are specified within existing functional parts that satisfy the needs of the exchange requirement.

7.2.4.5 Create functional parts

Where new functional parts are needed, they should be created to provide added support to exchange requirements.

7.2.4.6 Define business rules

The set of business rules that may need to be applied to further configure the exchange requirements/functional parts should be defined. These may be used to control properties to be asserted or values that can be assigned.

7.2.4.7 Capture process

As one or more exchange requirements are reverse engineered from software applications, they can be captured in a process map.

8 Compiling IDM components

8.1 General information

A schema is a description of the formal structure of a defined set of information. It is important that a building information system provider understands what a schema is and what it contains; for a user, it is only important to know what information the schema supports. However, definition of schemas is a vital part in the organization of the exchange requirement models and functional parts in IDM. Both have fully defined schema that are subsets of the complete information schema from which they are derived. In this clause, the way in which schemas are developed is discussed.

8.2 Adding functional parts

The principal unit of the IDM in which a schema is expressed is a functional part. This defines the technical content.

Each functional part has a fully developed schema that includes a set of entities. For instance, the functional part P2 shown below might have a schema that includes the set of entities {U, V, W, Z} whilst the functional part P3 might have a schema that includes the set of entities {T, U, V, X, Y}. Note that these functional parts both include the entities {U, V} in their schema. It is normal for an entity to be used in many functional part schemas.

A functional part can call upon, or include, other functional parts. This is a significant part of IDM because it allows a functional part to be defined once, then used many times.

The schema of a functional part that includes other functional parts is effectively the summation of all sets of entities. For instance, where a functional part P1 includes functional parts P2 and P3, the schema of P1 will be the summation of the sets of entities within P2 and P3, as illustrated in Figure 9. The schema of P1 contains the set of entities {U, V, W, Z} + {T, U, V, X, Y}.

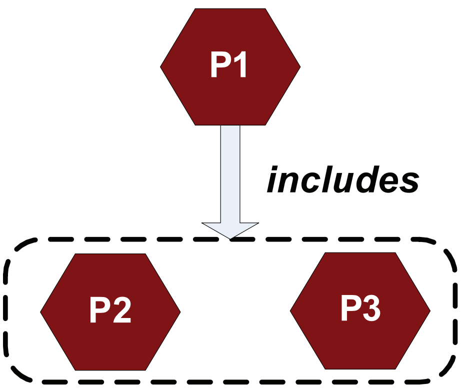

Figure 9 — Functional part including other functional parts

As well as the included functional parts P2 and P3, the functional part P1 may also contain a set of entities that are locally specified. For instance, the entities {S, X} may be directly specified. The fact that {X} is used in functional part P3 does not disqualify it from being included in P1. In this case, the schema of P1 contains the set of entities that are directly specified and the sets of entities of the included functional parts. That is, the schema of P1 contains the set of entities {S, X} + [{U, V, W, Z} + {T, U, V, X, Y}]. Expanded, this would give the set of entities within the schema P1 as {S, X, U, V, W, Z, T, U, V, X, Y}.

This shows the schema for P1 as including two occurrences of the entity {X}, two occurrences of the entity {U} and two occurrences of the entity {V}. This is not allowed. A set of entities forming a schema may only contain one occurrence of each entity. That is, the overlap of entity occurrences must be resolved. For P1, the schema resolves to {S, T, U, V, W, X, Y, Z} and this has only one occurrence of each entity.

This can also be seen in Figure 10 where it should be noted that even though entities are resolved to a single occurrence of each, the relationships between entities remain consistent.

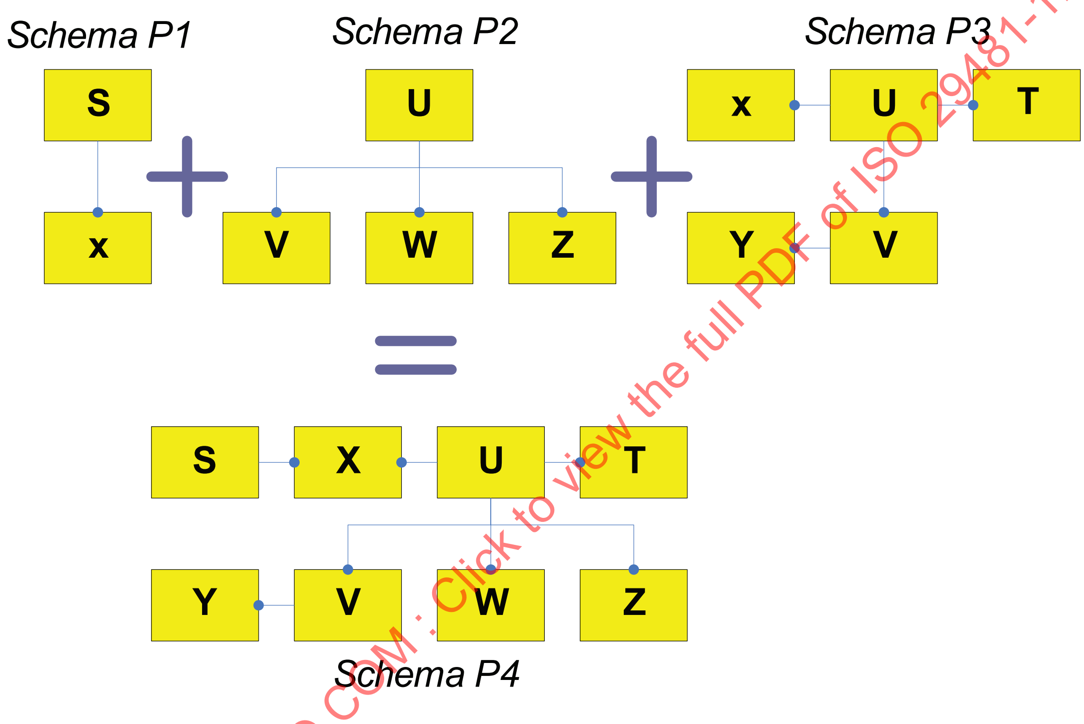

Figure 10 — Resolving entities added from schemas

Since an exchange requirement model is made up of functional parts, it follows that the schema for the exchange requirement model is determined by adding together the schemas of the contained functional parts in the same manner. For example, if the exchange requirement model R1 includes the functional parts P1 and P2, then

R1 = P1 + P2

However, if the functional part P1 itself includes other functional parts A, B, C and if functional part P2 itself includes other functional parts D, E, F, then it follows that

R1 = A + B + C + D + E + F

This is illustrated in Figure 11.

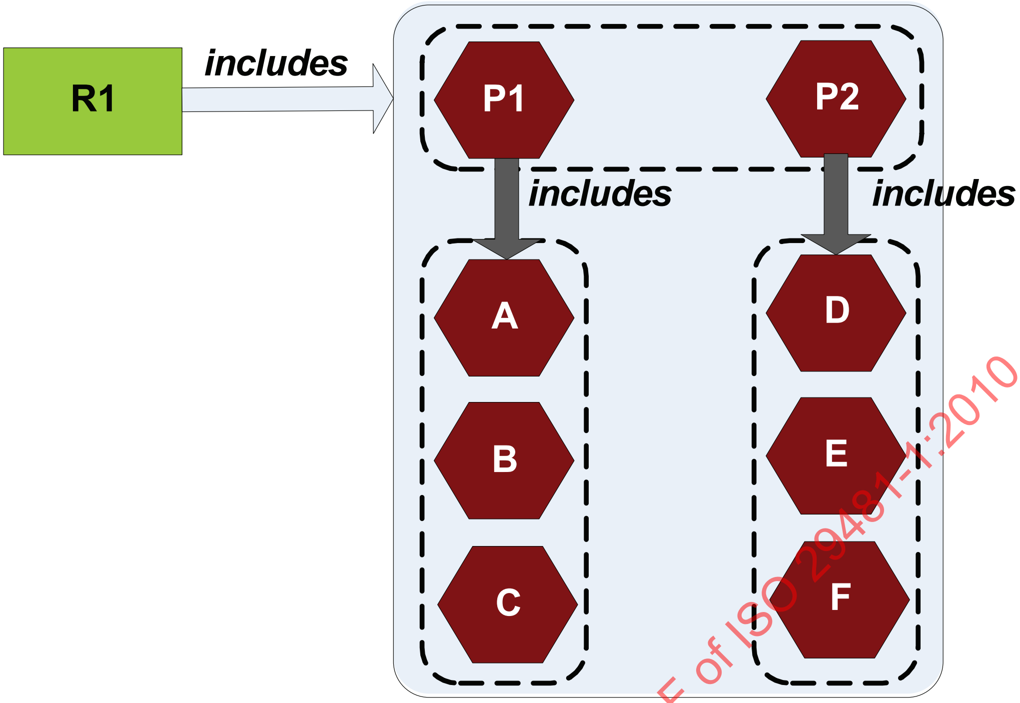

Figure 11 — Decomposition of functional parts

8.3 Adding exchange requirement models

In the same way as an exchange requirement model includes functional parts, it can also include other exchange requirement models. That is, information already defined for a prior exchange requirement model may be used within a current exchange requirement model. This facility can be used effectively to reduce the effort needed in describing the information requirements. For instance, if an exchange requirement R2 contains the exchange requirement R1 and the functional parts P1 and P2, then the schema for R2 may be determined by

R2 = R1 + P1 + P2

In fact, using an exchange requirement model in this way is no different than adding functional parts as shown in 8.2. This is due to the fact that an exchange requirement model ultimately reduces to the functional parts from which it is built. Therefore, reference to the exchange requirement model included is only a reference to a set of functional parts that are used to build it. (See Figure 12.)

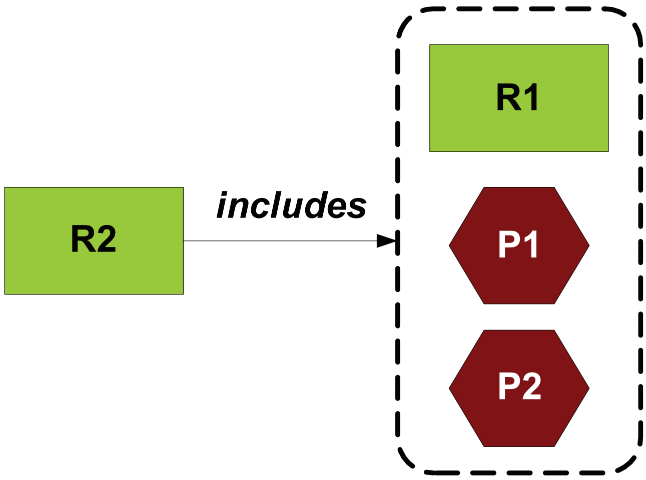

Figure 12 — Building exchange requirements with functional parts

8.4 Information model units

Subclauses 8.2 and 8.3 demonstrate that regardless of whether a functional part or an exchange requirement model is being discussed, the schema expressing the idea of the part or requirement is made up of entities (together with their attributes and relationships). Thus, there is no fundamental difference between a functional part and an exchange requirement model. These are simply convenient ways in which an information model can be broken up into larger or smaller parts (information model units) to facilitate the task of specifying schemas for practical information exchange.

Annex A
(informative)

Reference section

A.1 Example IDM scenario

A.1.1 General information

In this clause, a scenario for the use of IDM is described. The purpose is to provide a detailed introduction to the core terms used within IDM from a user perspective. This is done through a scenario that is from the viewpoint of one particular actor role as undertaken on a project.

A.1.2 Basic ideas

Before presenting the scenario, the basic ideas that it elaborates are described. This is done through the use of an illustration that diagrammatically shows how the different components within IDM interact. Since IDM also plays a role in a broader industry context, some additional ideas are necessarily shown in the illustration and described below.

Within IDM, there are two perspectives (see Figure A.1). These are seen as user requirements and technical solutions. A user requirement is independent of any schema and can be seen as an unvarying specification of what a user wants to achieve.

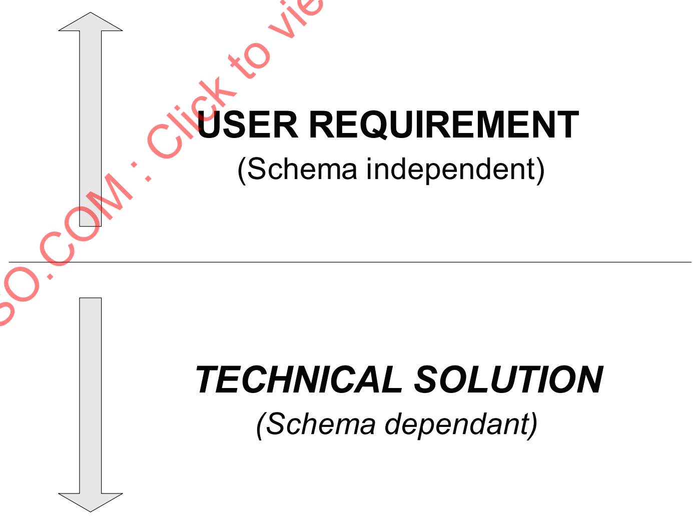

Figure A.1 — IDM perspectives

A technical solution is however schema dependent. That is, it is bound to a particular information model and to a particular version (or release) of that information model. This is the domain of the building-information system provider.

Within the two perspectives, there are a number of zones that characterize the various elements of IDM (see Figure A.2).

Within the user-requirement perspective, these are

- process maps (describing the overall process in which information exchange occurs),
- interaction maps (describing the actor roles and transactions between them; discussed below but not forming part of this part of ISO 29481),
- information delivery (describing the information exchange needs),
- reference processes (stored exchange descriptions captured at information delivery; not described in this part of ISO 29481), and
- the project schedule (occurrences of processes in the context of a project).

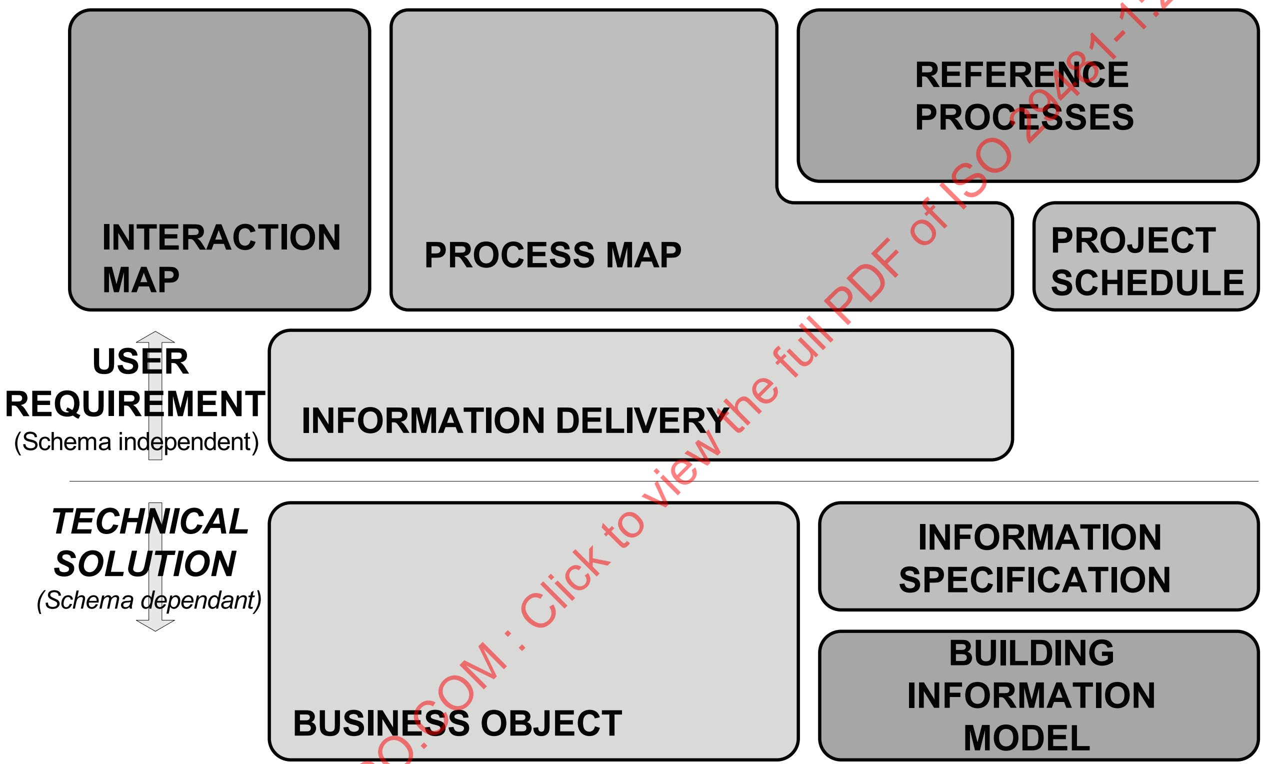

Figure A.2 — IDM zones

The technical-solution perspective includes

- the business objects comprising the exchange requirement models,
- functional parts bound to an information model, and
- business rules together with the information specification from which IDM schemas are derived and building information models whose content is specified by IDM schemas.

This is shown in more detail in Figure A.3.

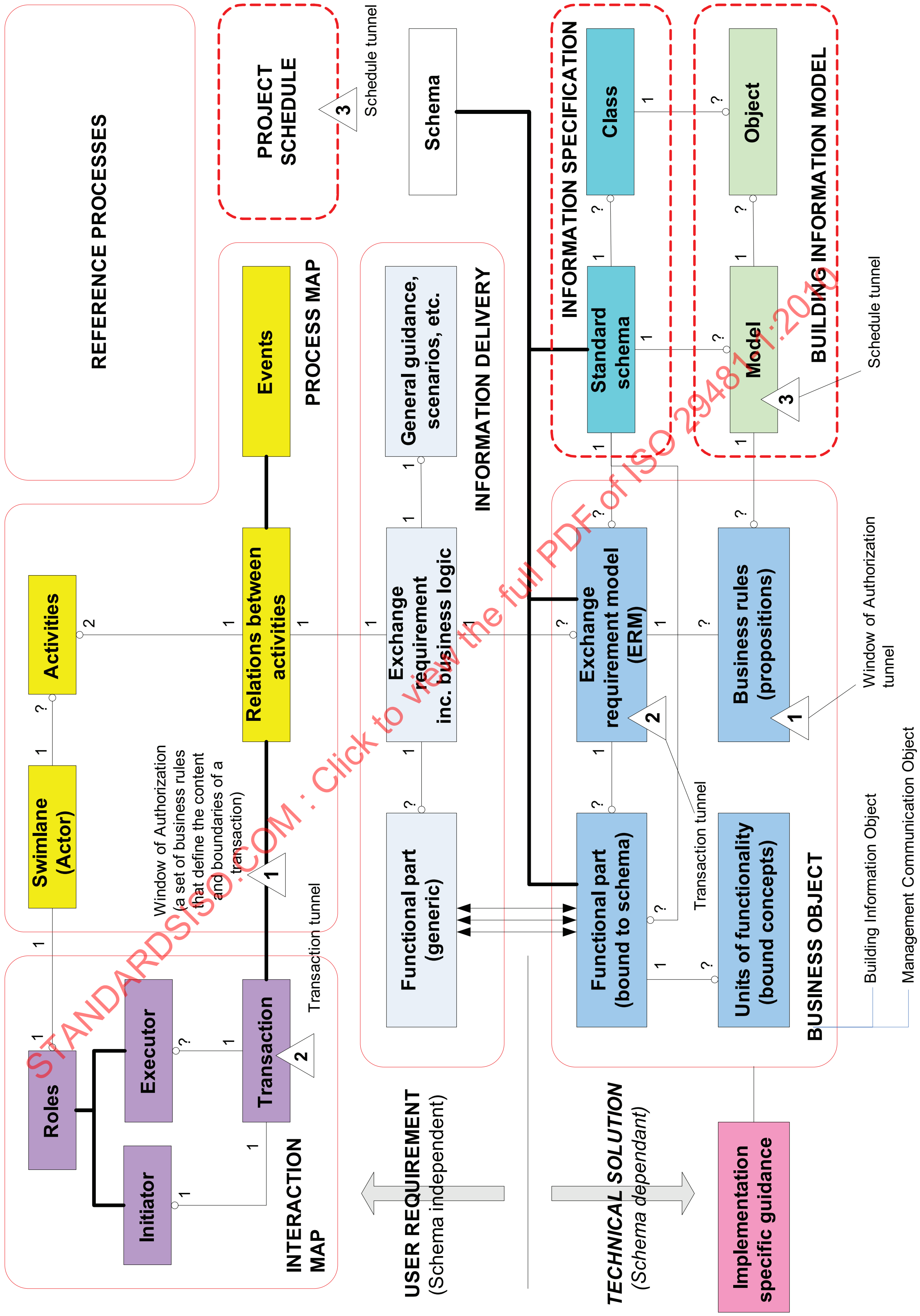

Figure A.3 — Diagram with concepts which are relevant for IDM

A.1.3 Initial position

The scenario is presented from the viewpoint of an actor who takes the role of a project manager who has to develop the co-operation and information exchange for the design of a new building project. For the project, it has been decided that all relevant information should be captured in a BIM using a building information system.

In addition, it has been decided that all information transfer, relevant to the management of the project and to its construction, will be done in digital form. There are many actors involved in the project covering the many disciplines needed for project development. This provides the project manager with a major challenge.

It is at this point that this part of ISO 29481 becomes significant. It provides guidelines and templates that enable the necessary agreements to be established quickly. The big advantage is that many software products for the construction industry support IDM and are certified for use with views of information model exchange standards such as IFC. This means that software can adapt to the required information exchange.

A.1.4 Road map

This scenario describes a road map for implementation of IDM in a building project. This consists of the following steps:

a) define the cooperation;

b) define the structure of the information of the building (and the construction project);

c) define the exchange requirements;

d) select a building information system;

e) completion of the actors.

A.1.5 Define the cooperation — Management communication

To bring cooperation in a project into focus, IDM makes use of the term “role”. A role is an abstract concept and represents a particular responsibility for the provision of information. A role need not be linked to an organization or person (actor). This makes it easy to define the cooperation in a project without needing to know exactly who will participate. Eventually, a role will be assigned to an actor; this may be at the time that the role needs to be fulfilled. IDM allows the assignment of a particular actor playing a role to change for each occasion that the role is played.

An important first step for the project manager is to define and select the roles that will be needed. This should be done in conjunction with a local authoritative listing of roles.

Then, the relationships between the roles are made. A distinction is made between the role “initiator” (makes a request) and the role “executor" (effectuates that request). If there is such a relationship between two roles, this is called a “transaction”. A transaction contains a set of messages that can be exchanged for a particular purpose between the two roles. A map that represents the roles and the transactions occurring between roles is called an interaction map. A message is a populated information model and contains data. Attachments may be linked to messages. For example, transactions are used to

— handle a commission,

— deliver a result,

— handle a request for change,

— review results, and

— request advice.

It is expected that documentation would be available with examples of the application of roles, transactions, messages and data in typical projects. These examples are available in the form of a customized communication schema. These can be read directly by an IDM compatible building construction information system.

The project manager finds one example communication schema that can be used. Only a few adjustments are necessary. Some roles and transactions appear unnecessary and the roles should be slightly adjusted so that certain transactions switch to other roles.

Using IDM, the project manager has defined the cooperation and communication requirements at the management level in a very short time. In addition, the process of information transfer has also been accurately defined. The result can be sourced directly into an IDM-compatible building construction information system.

A.1.6 Define the structure of the information of the building

The next step concerns the definition of the structure of the information from the building. The building information system requires such a structure to ensure that all the information will be stored at the right place and in the proper context. This is achieved firstly by defining the exchange requirement in a human understandable form. This is then converted into something that a computer can understand which is called an exchange requirement model. Creation of an exchange requirement model is helped significantly by the availability of a number of customized information units called functional parts.

The project manager finds an exchange requirement model that he can use. There are only a few adjustments necessary and these are dealt with by specifying a set of business rules.

A.1.7 Define the exchange requirements — Object communication

In the previous step, the exchange requirement models set out exactly what is the structure of the information from the building. The actual filling of the BIM results from the contribution of different roles. But what should be delivered by each role and to what formal requirements that contribution shall meet has not been fixed yet.

The project manager would like to ensure that the information provided by the various roles is delivered directly in the correct form to the building information system. That means that for a number of transactions the manager should specify what information should be supplied (content) and what form this information should have. This part of ISO 29481 indicates that, for a transaction, the following components can be added

— the exchange requirement,

— an exchange requirement model,

— the window of authorization: in the context of a transaction, an executive role (executor) can access the building information system. The window of authorization describes what information in this transaction by the role may be read or changed.

Templates are available for the delivery of a detailed design of a ventilation system. The project manager for this purpose will find an example for the exchange requirement, a requirement exchange model and example of a window of authorization. Only minor changes are needed.

A.1.8 Select a building information system

For the purposes of the new building project, a building information system is used. The building information system shall contain the BIM. An important requirement is compatibility with this part of ISO 29481 because this is a guarantee for interoperability between construction partners who also apply IDM-compatible software.