# CO-ORDINATION OF CLASSIFICATIONS FOR PRODUCT MODELLING AND ESTABLISHED BUILDING CLASSIFICATIONS

Classification for product modelling

A. EKHOLM
Computer Aided Architectural Design, Lund Institute of Technology/Lund University, Lund, Sweden

Durability of Building Materials and Components 8. (1999) *Edited by M.A. Lacasse and D.J. Vanier*. Institute for Research in Construction, Ottawa ON, K1A 0R6, Canada, pp. 2465-2475.


## Abstract

Computer based information management introduces new possibilities and puts partly new requirements on information and classification systems. The rapid develop-ment and dissemination of information technology within the construction sector demands international co-ordination of standards and classification systems. This paper presents the preliminary results of a study of the product model classification systems IFC and POSC/Caesar in relation to the established building classification systems as defined in ISO 12006-2 and the Swedish BSAB 96. The study is part of a project with the aim to analyse the structure, scope and direction of development for these different systems, and to investigate the possibilities of co-ordination and co-operation between them. It shows a need for co-ordination of concepts in the ontological and the domain levels.

Keywords: CAD, design, databases, building classification, interoperability.

## 1 Introduction

### 1.1 Co-ordination of systems for information sharing

The rapid development and dissemination of information technology within the construction sector demands international co-ordination of standards and classification systems. Computer based information management introduces new possibilities and puts partly new requirements on classification systems. This paper reports some preliminary results of an ongoing investigation of the prerequisites for co-ordinating classification systems for product modelling in the construction context, with established building classification systems. The study is financed by the Swedish national research program ”IT Bygg&Fastighet 2002”

(IT Construction & Facility Management 2002), concerned with information technology for Construction and Facility Management.

Established building classification systems studied here are the ISO standard 12006-2 “Organization of information about construction works – Part 2: Framework for classification of information” (ISO 1997), and the Swedish building classification system BSAB 96 (BSAB 96 1998). Classifications for product modelling studied here are the POSC/Caesar Data Model and Reference Library (POSC/Caesar 1998), developed by the POSC/Caesar Association, and the Industry Foundation Classes, IFC, developed by the International Alliance for Interoperability, IAI (IAI 1997b). The project analyses the structure, scope and direction of development for these different classification systems and investigates the possibilities for co-ordination.

## 2 Established building classification

### 2.1 ISO 12006-2

The recently developed international standard ISO/CD 12006-2 ”defines a framework and a set of recommended table titles supported by definitions, but not the detailed content of these tables”. It is intended to be used as a framework for developing specific building classification systems by organisations on a national or regional basis (ISO 1997).

ISO/CD 12006-2 identifies the main classes that are of interest to the construction sector’s building classification for purposes of CAD, specification, product information and cost information systems (ibid:iv). The scope of the standard is the complete life cycle of construction works within building and civil engineering. It lists recommended tables according to particular views or principles of specialisation and gives examples of entries that may occur in these tables (ibid:1).

The identification of classes in ISO/CD 12006-2 is based on views on construction works. The functional and spatial views are based mainly on relations to the user or the environment, see figure 1, sec. 2.2. Other views are related to a process model and identify construction resources, construction processes and construction results. The classes *construction entity* and *construction complex*, are based on a functional view in relation to use. The class *space* is based on a combined spatial and functional view on a construction entity. Physical parts of construction entities are classified in three separate classes based on different views. These classes are: *element*, e.g. floor, based on a functional view, *work result*, e.g. structural precast concrete, based on a construction result view, and *designed element*, e.g. floor of structural precast concrete, based on a combination of functional and construction result views.

Each view consists of classes that together enable a complete classification of the things in the collection; i.e. every thing is classified, and belongs to only one class in each level of abstraction of the view. This enables quantity take-off to be carried out independently in the *element*, *designed element*, and *work result* views, for example with the purpose of cost estimates.

Established classification systems do not handle geometrical properties of construction works, they are supposed to be used together with drawings that contain this information. A physical part of a building is determined by the combination of drawing information and the classifications as *element* and *work result*.

A main feature of building classification systems that adhere to ISO 12006-2 is that they support design by independently allowing functional and compositional views on physical parts of buildings (Ekholm and Fridqvist 1998).

### 2.2 BSAB 96 and its theoretical foundation

The BSAB system has a long-standing tradition as the Swedish de facto standard for product specification, materials specification, cost estimates and document management (BSAB 96 1998). The conceptual scope of the BSAB system is similar to that of the ISO Framework standard. However, the use scope of the BSAB system is the Swedish construction industry. The classes are developed together with the users of the system, and represent established industry practice. The classes and their names are well known and used in documents of all sorts throughout the construction and facility management processes.

The classification tables of BSAB 96 adheres to the ISO 12006-2 Framework standard. The tables in BSAB 96 contain subclasses of the main classes in the Framework. Both standards are based on common principles for building classification established already with the SfB-system from 1950 (ibid). Just like the ISO Framework standard, the BSAB 96 system is based on the idea that the main classes express different views on construction works. In BSAB 96 this idea is grounded in a theoretical model based on generic property theory and systems theory (ibid). According to this model, a system is composed of interrelated parts, that may be systems themselves, and has relations to an environment. A functional view, ”top-down”, on a system concerns its interaction with the environment, and a compositional view, ”bottom-up”, concerns its composition and intrinsic relations. The main classes in ISO 12006-2 and BSAB 96 are developed from these views. The principles for applying these theories to building classification are presented in (Ekholm 1996).

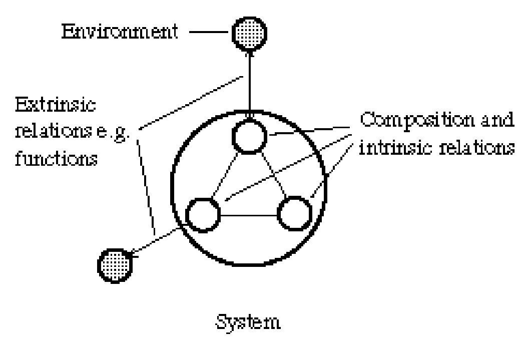


**Fig. 1: Basic properties of systems**

## 3 Levels in databases for product design and documentation

### 3.1 Level structure

Research within product modelling has led to a basic understanding of how to structure databases for the purpose of product documentation. Björk has identified five levels of abstraction in building product modelling: the information modelling language, the generic product description model, the building kernel model, the aspect model and the application model (Björk 1995). The first three levels correspond to those identified in this analysis, the information modelling language level, also called the data definition language level, the ontological level, and the domain level. The domain level may be specified in subsequent levels of relevance, e.g. those proposed by Björk.

An *information modelling language*, or *data definition language*, supports the definition of database objects, e.g. entities and relationships (Date 1990:36). The EXPRESS language is an example of a data definition language developed to enable product modelling. EXPRESS is part of the ISO STEP standards (ISO 1994). The EPISTLE, POSC/Caesar, and IFC information systems all use EXPRESS.

The concepts in the *generic product description level* or *ontological level* represent the most generic properties of objects in the domain of discourse. The concepts in this level may be related in an ontological model. One example of an ontological model is the theoretical foundation of BSAB 96 presented above (BSAB 96 1998). The ontological model makes up a theoretical foundation for defining and relating the most generic aspect views on the objects in the domain. The result is a generic classification framework. In order to support design, the framework must differentiate among and allow combinations of spatial, functional and compositional views (Ekholm and Fridqvist 1998).

The classes in the *domain level* constitute a framework for classification of the domain specific objects. Every domain framework is based on presuppositions in the ontological level, explicit or implicit. The domain classes are subclasses in different degrees of specialisation of the main aspect classes defined in the ontological level. The domain framework may also contain domain specific relations and attributes. The domain framework may be organised and stored as a library of predefined classes from, for example, classification systems and earlier projects. During design this information can be combined to make up project specific descriptions of domain objects.

# 4 POSC/Caesar

## 4.1 Introduction

POSC/Caesar is a joint initiative of the Petrotechnical Open Software Corporation and the CAESAR Offshore Project. The objective of the POSC/Caesar project is to develop a standardised product model for handling life-cycle information about oil and gas facilities (POSC/Caesar 1998). The product model is developed on the basis of the EPISTLE framework (Angus and Dziulka 1998). The work has resulted in practical means for handling product information based on a data model and additional library classes.

The POSC/Caesar project is of specific interest to the construction sector since it represents an attempt to create a computer based platform for integrated information management. It concerns the oil- and gas industries, but considering plants and other civil engineering works, these have much in common with building construction. For example, many of the library classes concern construction works.

## 4.2 The POSC/Caesar meta model

The data definition language for the POSC/Caesar approach is based on a subset of the concepts of the EXPRESS information modelling language. This subset of EXPRESS is called the POSC/Caesar Meta model; it is the most abstract POSC/Caesar model.

## 4.3 The POSC/Caesar data model

### 4.3.1 Background

The following presentation of the POSC/Caesar Data model concerns the version called ”Snapshot E” (POSC/Caesar 1998). The scope of the POSC/Caesar Data model is classes considered of importance for handling life-cycle information in a generic way about oil and gas facilities (ibid). The POSC/Caesar Data model is based on the principles of the EPISTLE Framework, but Snapshot E, marks a radical change in fundamental conceptual structure compared with the EPISTLE Framework (ibid).

### 4.3.2 The class/instance dichotomy

The most fundamental entity in the POSC/Caesar Data model is the *thing*. A *thing* represents something material or abstract. It has two subclasses, *object* and *association*. Association entities relate object entities. Subclasses of *association* are both abstract, representing conceptual associations like *classification* and *characteristic_possession*, and concrete, representing factual relations such as *connection*, *composition*, and *transportation*.

In POSC/Caesar, an *object* has an ”independent existence”. The *object* is distinguished from an *association*, which relates *object* classes. The *object* class has two subclasses, *instance_object* and *class_object*. The *class_object* refers to ”a collection of well determined things”. The *instance_object* is defined as ”A type of thing that is individual in nature”; it is not a ”class, group, or set of things”. A reason for the distinction between *instance_object* and *class_object* is the concern to differentiate between representations of one identified member of a class, and representations that regard every member of the class. *Person_class*, a subclass of *class_object*, can be used to exemplify how POSC/Caesar implements this. Examples of *person_class* are ”rich people” and ”Australian people”, a rich person and an Australian would be members of these classes respectively. The entity specific_person is defined as an ”instance object that is a particular human being or group of human beings”. A person_classification is a subtype of the classification association. The person_classification entity associates a specific_person with person_class, the former as member of the latter.

### 4.3.3 The classes *functional_object* and *physical_object*

Of specific relevance to discuss in this context are *functional_object_class* and *physical_object_class*, both subclasses of *object_class*. A *functional_object* represents a functional view on an object in the domain. The definition of *functional_object* is: ”A type of instance of object (*instance_object*, author’s remark) that is a function or role and may be involved in many activities, considered independently of the physical objects that may undertake the roles from time to time. Roles may be active (making it happen) or passive (happening to). *Functional_object* must be either Facility, Flow or Service" (POSC/Caesar 1998).

A *physical_object* is "A type of instance object that obeys the laws of physics. This includes material, energy and space. A physical object is either a specific or a typical physical object." The *physical_object* is not defined by other characteristics than that its is a concrete thing; seen as a view, it identifies objects by concrete existence.

In principle the same object may be found both as a *functional_object* and as a *physical_object*. The reason for this is that function may be used as a basis for classification of physical_object. One example of this is the subclass of *physical_object* "gas detector", with the definition: "device supposed to detect certain types of gas molecules" clearly describing functional characteristics. However there are other characteristics than function that determine membership of *physical_object*, for example, an object may be defined as a product of a member of a method activity class. This resembles the construction result view in established building classification.

### 4.4 The reference data library

The objective of the Data Model is to provide the basic class structure of the Reference Data Library, RDL. Each of the main classes in the RDL may be further specialised according to domain specific needs. The RDL is intended to be used both as a design and documentation framework. In the latter case the RDL is intended have a separate project database consisting of subclasses and instances referring to a factual facility e.g. an offshore platform. This is implemented at present with the Åsgård B floating gas platform (Port 1998).

### 4.5 POSC/Caesar and established building classification

The POSC/Caesar system may in principle be used for construction information. The classes of ISO 12006-2, and BSAB 96 can be part of a specific Reference Data Library developed for the construction industry. In the Reference Data Library, these classes may be accommodated as subclasses of *functional_object_class* and *physical_object_class*. The classes that are based on a functional view can be sub-classes of the *functional_object_class* "facility" in POSC/Caesar. Designed elements and construction products, both grouped by function but based on combinations of views, as well as work result class based on a compositional view could be subclasses of *physical_object* in POSC/Caesar. However, it remains to investigate in greater detail how classes for construction can be accommodated in POSC/Caesar.

## 5 IFC

### 5.1 Introduction

The goal of IAI, Industry Alliance for Interoperability, is "enabling interoperability between AEC/FM applications from different software vendors" (IAI 1997a). This goal is to be achieved through development of the Industry Foundation Classes, IFC. The objective of IFC is to "provide a framework for sharing information between different disciplines within the AEC/FM industry".

The Industry Foundation Classes, IFC, are divided into four separate model levels: *Resources Layer*, *Core Layer* with *Kernel and Core Extensions*, *Interoperability Layer*, and *Domain Model Layer*. All the IFC schemas are developed using the EXPRESS data definition language.

## 5.2 Resources layer

The IFC Resources are classes that may be referenced by classes in every level of the IFC class hierarchy. The Resource classes are *IfcUtilityResource*, *IfcMeasureResource*, *IfcGeometryResource*, *IfcPropertyTypeResource*, and *IfcPropertyResource*. The *IfcUtilityResource* includes IFC-classes for the concepts Identifier, Ownership, History, Registry and Table, to be used in project administration. The *IfcMeasureResource*, is adapted from ISO 10303 part 41 Measure schema, and specifies units and measures that may be assigned to quantities. The *IfcGeometryResource* specifies the resources for geometric and topological representation of the shape of a product. These resources are partly developed as adaptations of ISO 10303 part 42: Integrated Generic Resources. Geometric and Topological Representation (IAI 1997b:4-40). The *IfcPropertyTypeResource* defines the kinds of property definitions that can be attached to objects and relationships. The I*fcPropertyResource* comprises classes for the concepts ”actor”, ”classification”, ”cost”, ”material”, and ”date/time”. Among others, one subclass of *material* is a list of all kinds of material. The class *IfcClassification* makes it possible to categorise objects according to specific classification systems.

## 5.3 Core layer with kernel and core extensions

The Core Layer consists of the IFC Kernel which provides all the basic concepts required for IFC models, it is ”a kind of Meta Model that provides the platform for all model extensions” (IAI 1997a:6). These constructs are not AEC/FM specific, as are the extensions to this core (IAI 1997b:7-100). The classes of the Kernel are *IfcObject*, *IfcRelationship* and I*fcModelingAid*. The *IFC Resources* together with the *Core Layer* constitute a generic framework for product modelling in the ontological level. However, the identification of framework classes is not based on an ontological model.

The Core Extension Layer contains extensions for the Kernel classes *IfcProduct*, *IfcProcess*, *IfcDocument*, and *IfcModelingAid*. The core extensions are a specialisation for the AEC/FM domain of classes defined in the Kernel. The *IfcProduct-Extension* defines concepts like element, space, site, building, and building storey (ibid:8-111). The *IfcProcessExtension* has subclasses, which intend to ”capture information concerning the work required in order to create a product”. It is possible to define work tasks and resources used in these. Subclasses of *IfcDocumentExtension* will allow ”specifications for the information content of typical document types encountered within building construction”. Presently only cost schedules are covered. *IfcModelingAidExtension* has subclasses to aid in the development of Project models, e.g. *IfcDesignGrid* and *IfcReferencePoint*.

## 5.4 Interoperability layer

The *IfcSharedBldgElement* in the interoperability layer ”covers the definition of building elements that are shared among several IFC domain or application type models” (ibid:12-147). Subclasses here are e.g. *IfcBeam*, *IfcBuiltIn*, *IfcColumn*, *IfcCovering*, *IfcDoor*, *IfcFloor*, *IfcRoofSlab*, *IfcWall*, and *IfcWindow*.

The *IfcSharedBldgServiceElements* in the interoperability layer defines concepts for interoperability in the building services domain (HVAC) (ibid:13-162). Classes in this level are *IfcElectricalAppliance*, *IfcEquipment*, *IfcFixture*, *IfcDistributionElement*, *IfcEngineeringMaintenence*, and *IfcManufacturedElement*.

## 5.5 Domain model layer

Domain models provide classes that are specific to a particular application within the AEC domain. In version 1.5 there is a sparse selection of subclasses of *IfcArchitecture*, *IfcHVAC*, and *IfcFacilitiesMgmt*.

## 5.6 IFC and established building classification

The *IfcProductExtension* classes, e.g. *IfcBuildingElement*, *IfcSpace*, and *IfcBuilding*, bear similarities with, and belong to the same level of abstraction as, the main classes defined in the ISO 12006-2 Framework standard. Likewise, the subclasses of *IfcSharedBldgElement* and *IfcSharedBldgServiceElements* in the Interoperability layer seem similar to, and are defined in the same level of abstraction as, the BSAB 96 classes.

As shown, there is a significant overlap between some IFC-classes and established classification systems. There is need for a strategy to resolve this potential conflict. The possibility to make a reference from an IFC-class to an external classification system through the class *IfcClassification* is no solution, since the class definitions may not be consistent, and the class scopes may be different.

In established classification, the class *designed element* serves the purpose of enabling a combined functional and compositional view. The latter seems to be the objective of the *IfcSharedBuildingElement.* Subclasses of *IfcSharedBuildingElement* have names that indicate a functional view, but have in their definitions both material, geometry and functional attributes, e.g. the *IfcBeam* is ”a horizontal, or nearly horizontal structural member designed to carry loads” (IAI 1997b:12-149).

There are no subclasses of *IfcBuildingElement* that remind of *work result* classes like *masonry* or *in-situ-concrete* of BSAB 96. Objects defined by such class attributes may also be of interest to handle separately from the role they play in a construction.

Superficially, the IFC system shows significant overlap with ISO 12006-2 and BSAB 96. However, the IFC system is not based on separation of views. The main principle is to attach attributes to the *IfcElement* and its subclasses. One problem seems to be that the *IfcBuildingElement* classes are already having attributes that might not conform to the ISO 12006-2 standard. It remains to study how the classification structure of IFC can be co-ordinated with established industry standards, e.g. with the purpose to support a differentiation of views.

The approach in established building to ”categorize elements according to primary functional role or as part of a system” is criticised by IAI for limiting the modelling capacity (ibid:2-15). The approach is an ”attempt to avoid this by defining model elements, functional roles, and systems separately so that an element can assume multiple roles and/or be a member of multiple systems” (ibid). In order to support design, there is a possibility to attach attributes to objects and relationships (IAI 1997b:5-74).

The freedom of defining and combining attributes of a model object is in established building classification achieved by separating the spatial, functional and compositional views. For example, any drawing object referring to, and representing spatial, or geometrical, properties of parts of buildings may be assigned to any *element* (functional view) or *work result* (compositional view) or *designed element* (functional and compositional view). However, in practice there are limitations to these combinations. The relations between elements and geometry has been studied in the NICK project (Tarandi 1994), and the relations between element and work result based on industry practice are expressed in the designed elements table in BSAB 96 (BSAB 96 1998).

# 6 Concluding remarks regarding the need for co-ordination between systems

## 6.1 Theoretical background, level structure and views

An overview of the structure of the analysed classification standards is presented in Table 1 below. ISO 12006-2 is a generic classification framework for the construction domain. It is not based on an explicit ontological framework, but adheres to the same principles as BSAB 96. The BSAB 96 is an example of a domain specific classification system. It is based on an explicit ontological model, which is used to define the main views and classes.

POSC/Caesar and IFC have a wider scope than established building classification systems. They are developed to completely support computer based information management. For example, they include geometry definitions and specify relations between classes. The need for co-ordination, mentioned in section 1.1, primarily concerns the established classes covered by the established classification systems.

The POSC/Caesar Data Model is a generic framework for design and documentation of artefacts in the ontological level, but it is not based on an explicit ontological model. The Reference Data Library is a domain framework for design and documentation of oil and gas facilities, based on the Data Model. It would be possible to develop a Reference Data Library also for the construction industry, based on the POSC/Caesar Data Model.

**Table 1: Level structure of the analysed classification standards**

| Standards/ Levels | ISO 12006-2 | BSAB 96 | POSC/ Caesar | IFC |
|---|---|---|---|---|
| **Data definition language** | (English) | (Swedish) | POSC/Caesar Meta Model | EXPRESS |
| **Ontological level** | (Systems model) | Systems model | POSC/Caesar Data Model | Core Layer and IFC Resources |
| **Domain Generic level** | ISO 12006-2 Framework | ISO 12006-2 Framework | Reference Data Library | Core Extensions |
| **Domain Specific level** | | BSAB 96 Classes | Reference Data Library | Interoperability Layer |

In IFC, the Core Layer together with the IFC Resources constitutes a generic framework for product modelling in the ontological level. However, it is not based on a generic framework for the AEC/FM domain. The Interoperability Layer is to be developed into a domain specific class library. A limitation of the IFC-system is that its basic class structure is not based on a differentiation of views.

In order to enable integration between these different systems, it is necessary that they have a shared ontological view, classification approach and level structure. An international standardisation of ontology and classification principles for the AEC/FM domain would enable this integration. Concerning the generic classes for the AEC/FM domain, co-ordination is required between ISO 12006-2, a future construction specific POSC/Caesar Reference Data Library, and the IFC Core Extensions. Similarly, concerning AEC/FM domain specific classes, co-ordination is necessary between national classification systems like BSAB 96, a future construction specific POSC/Caesar Reference Data Library and the IFC Interoperability classes.

## 7 References


Angus C. and Dziulka P. (1998). EPISTLE framework V2.0 Issue 1.22. (Ed. Peter Dziulka, Keyworth Institute, Department of Mechanical Engineering, University of Leeds, UK.)

Björk B.-C. (1995). Requirements and information structures for building product data models. Espoo: Technical Research Centre of Finland, VTT Publications no 245.

BSAB 96 (1998). BSAB 96 System och Tillämpningar (an English version is forthcoming). Stockholm: AB Svensk Byggtjänst (The Swedish Building Centre).

Date C.J. (1990). An introduction to database systems. Vol. 1. Addison-Wesley Publishing Company.

Ekholm (1996). A conceptual framework for classification of construction works. Electronic Journal of Information Technology in Construction (Itcon). Vol. 1. Stockholm: Royal Institute of Technology. URL: http//itcon.org.

Ekholm A. and Fridqvist S. (1998). A dynamic information system for design applied to the construction context. In *The life-cycle of Construction IT innovations* (Eds. B.- C. Björk and A. Jägbeck). Proceedings of the CIB W78 workshop, June 1998, Stockholm: Royal Institute of Technology.

IAI (1997a). IFC Model Architecture. Industry Foundation Classes Release 1.5 Beta. (Ed. Thomas Liebich and Jeff Wix). International Alliance for Interoperability.

IAI (1997b). IFC Object Model. Industry Foundation Classes Release 1.5 Final version Thomas Liebich and Jeff Wix). International Alliance for Interoperability.

ISO (1994). ISO 10303-11. Product data representation and exchange - Description Methods - The EXPRESS language reference manual. Geneva: International Organisation for Standardization.

ISO (1997). ISO/CD 12006-2, Organization of information about construction works – Part 2: Framework for classification of information. ISO/TC 59/SC 13 N 86, Date 1997-08-25. International Organisation for Standardization. Secretariat: NSF.

Port S. (1998). Plant Information Management at Statoil Norway. CIMdata Inc. 1998.

POSC/Caesar (1998). URL: http://www.posccaesar.com. Snapshot E. Accessed December 1998.

Tarandi V. (1994). NICK II – Vidareutveckling av neutralt format för intelligent CAD-kommunikation. Svensk Byggtjänst, Rapport 19, Stockholm.