Precast Concrete BIM Standard Documents

Volume I

Model View Definitions

Executive Overview

This is an intermediate release of the Model View Definitions for data exchanges of precast concrete information. It outlines the MVDs in three volumes:

- **Volume 1 (this volume):** Overview, with specification of Exchange Models for use various use cases and the High Level Concepts making up the Exchange Models. These define what information is to be included in the exchanges and to what level of detail. They are presented in Model View Definition (MVD) diagrams.

- **Volume 2** contains the detailed binding definitions for all of the exchange concepts that are used in the diagrams of volume 1. These are IFC 2x4 bindings, and graphic depictions of the IFC entities and property sets used with their relationships, implementation rules, figures showing typical situations in precast buildings, and Part 21 file examples using EXPRESS. All of these are also available on the "IFC Solutions Factory" IAI MVD website, at http://blis-project.org/IAI-MVD/.

- **Volume 3** contains the detailed binding definitions for all of the exchange concepts that have been defined by other groups developing MVDs and that are re-used in the MVDs of volume 1. All of these are also available on the "IFC Solutions Factory" IAI MVD website, at http://blis-project.org/IAI-MVD/.

The MVD is based on the functional specification called an Information Delivery Manual (IDM), completed in January, 2009, under this contract. This Model View Definition (MVD) specifies at the implementation level the information needed to support the workflows defined in the IDM, covering the major digital exchanges dealing with precast concrete. The exchanges are defined to be implemented using the Industry Foundation Classes (IFC) model schema. The IFC bindings will support a set of standard export and import exchange capabilities for commercial software products, enabling seamless exchange of digital information and enhancing work processes.

A particular exchange will depend on varied subsets of the overall model data. Thus these subsets or modules will be used selectively in multiple different contexts. We adopt the idea of information 'Concepts' to represent the information items in a way that allows them to be composed for different workflow cases. Implementers can re-use the software modules and testing and certification of modules may also possible. The notion of Concepts has grown from previous implementation activities, led by the BLIS effort in the early 2000s.

The Concepts are drawn directly from the Information Delivery Manual (IDM) that was prepared by the Precast Concrete BIM Project team, including the PCI Advisory Board and the Technical Support Team. The Concepts include such varied issues as the degree of detail needed, connectivity, aggregation and nesting relationships, type of geometry representation, and others used in different workflows. Our Concepts were also drawn from a library of Concepts that have been proposed or defined by other BIM standard efforts in the IFC Solutions factory website: (http://www.blis-project.org/IAI-MVD/). These Conceptsare available for possible re-use and some of them may have already been implemented by BIM software companies. The Concepts for any domain such as precast form a hierarchical lattice, from high-level aggregated Concepts to leaf ones, the leaves providing bindings to IFC (e.g., the associated IFC implementation code). Most of the bindings are based on the new 2.x4, currently in beta release, available on the buildingSMART website: http://www.iai-tech.org/products/ifc_specification/ifc-releases/ifc2x4-release/beta1-release.

The current diagrams start with the IDM requirements from which they were derived. These are followed by the Concept structures need to support those functions. These are then associated with the leaf bindings to IFC, using a diagramming method that is widely used within the IFC community. In many cases, certain information about a precast piece needs to have different representation and these alternatives are included in the Concept bindings.

This initial draft addresses the Priority One and Priority Two implementations agreed to at the April 24-25 meeting. It is being reviewed to gain feedback from both precast concrete information users and from software implementers regarding definitional issues and adequacy of documentation. The current bindings reflect the technical team's judgment regarding best implementation of the IDM requirements. These will be reconciled, as needed, with software implementation capabilities and limitations.

Model View Definitions

Name	Model Exchange for Precast

Change Log		
8-September-09	Version 2.1 for draft review by PCI BIM Committee	chuck.eastman@coa.gatech.edu cvsacks@techunix.technion.ac.il ivan.panushev@gatech.edu manu.menon@gatech.edu shiva_aram@gatech.edu
31-July-09	Version 2.0 for draft review by PCI BIM Committee	
9-Jun-09	Version 1.0 for draft review by PCI BIM Committee	
16-Apr-09	Version 0.5 for draft review by PCI BIM Committee	

c

Model Views	(all)

TABLE OF CONTENTS

Background

The National Building Information Model Standard (NBIMS) is a set of interoperability standards for exchange of facility and infrastructure data throughout the life-cycle of a project. NBIMS is a joint project coordinated by the National Institute of Building Sciences (NIBS) and the buildingSMART Alliance, a working committee within NIBS, in conjunction with many other facilities-related associations and software companies worldwide.

The goal of the project within which this document was produced is to develop a national BIM standard for precast concrete design, engineering, fabrication and erection. The method of this work is to define first, the user functional requirements in a report called an Information Delivery Manuals (IDM). It was completed in January, 2009, and provided four process models that covered four distinct workflows, reflecting different project procurement methods and construction types (architectural and structural). Each of the process models was used to functionally define the information content required in each exchange.

Based on the IDM specifications, the Model View Definitions (MVD) are defined for the significant communication and data exchanges associated with all use cases associated with precast concrete. The primary orientation is that of the precast concrete fabricator. The project is funded by the Charles Pankow Foundation and the Precast/Prestressed Concrete Institute (PCI).This project was preceded by a feasibility study sponsored by the Charles Pankow Foundation to determine the issues of information exchange, focusing on architectural precast. This work has grown from and expands upon that study.

Working teams were formed within the membership of the industry BIM Advisory Committee, which functions under the auspices of the Precast/Prestressed Concrete Institute to provide the domain expertise for this endeavor. The initial step was to define the IDM use case specification. Now, the MVD are being specified and communicated with both the advisory committee for functional correctness and with software implementers.

The members of each of the four working teams are listed in the following table. Their contributions are gratefully acknowledged.

PCI BIM Advisory Committee Working Teams

Chair: Michael LaNier   BERGER/ABAM Engineers Inc.

A. Projects led by Precasters:

1. Jason Lien - chair   Encon United
2. Jennifer Huber Encon United
3. Monty Overstreet FDG, Inc
4. Mike Sloter   IPC Inc
5. Mark Kraft   CEG Engineers

B. Projects where Precaster is a subcontractor:

1. Michael Slobojan – chair Structureworks
2. Mark Potter Finfrock Industries
3. Davis Chauviere HKS Inc
4. Charles Pool   Tekla
5. Dieter Maucher Tindall Corporation
6. Jim Davis Stresscon

C. Architectural Precast:

1. David Orndorff - chair Shockey Precast
2. John Wang Mid State Precast
3. Mike LaNier BERGER/ABAM Engineers Inc.
4. Aaron Fink   Oldcastle Precast

D. Fabrication through Erection:

1. Mike Putich - chair
2. Dan van Vieren   Concrete Vision
3. Wayne Norris Metromont
4. AJ Scarfato   Metromont
5. Earle Kennett NIBS
6. Wayne Kassian   Kassian Dyck& Assoc
7. Karen Laptas   Blue Ridge Design, Inc.

Technical Advisory Team:

Chuck Eastman   Georgia Tech
Rafael Sacks   Technion, Israel Institute of Technology
Ivan Panushev   Georgia Tech
Shiva Aram   Georgia Tech
Manu Venugopal Georgia Tech

Technical Consultants:

Richard See   Digital Alchemy
Thomas Liebich AEC3

MVD Overview

Purpose

This Model View Definition (MVD) specifies for the precast concrete domain the digital information needed to support the exchanges in a variety of workflow 'use cases'. This MVD also prescribes implementation bindings using the Industry Foundation Classes (IFC) model schema. The IFC bindings are intended for software developers to implement a set of standard export and import exchange capabilities for their commercial software products, which will in turn enable seamless exchange of digital information in multiple precast concrete workflows. Because there are multiple possible workflows, and selection of a particular exchange will depend on particular work practices and project contexts, the approach is modular: exchange models are composed of different information items at different possible degrees of detail. Accordingly, we have defined information 'Concepts' as modules to represent the information items in a way that allows them to be composed for different workflow cases and implementers to re-use software modules and makes modular testing possible.

The Concepts are drawn directly from the Information Delivery Manual (IDM) that was prepared by the Precast Concrete BIM Project team, which included domain experts (organized in a Precast BIM Committee sponsored by the Precast Concrete Institute (PCI) and chaired by Mike LaNier), and technical advisors (in a team led by Prof. Chuck Eastman at Georgia Tech). The IDM document defined the functional requirements for data exchanges needed for precast concrete design, fabrication and construction, defined as Exchange Models (EMs). For each EM defined in the IDM, the information requirements were defined and related to the Concepts needed for that exchange. The parameters included such issues as the degree of detail needed, connectivity, material properties, aggregation and nesting relationships, type of geometry representation, and others.

Concepts were also drawn from a library of Concepts that have been proposed or defined in the IFC Solutions factory website: (http://www.blis-project.org/IAI-MVD/). These are the Concepts developed by other BIM standard efforts and available for possible re-use..

MVD Diagrams

The exchange requirements expressed in the IDM formed the basis for preparation of this MVD document. The MVD lays out the Concepts. The Concepts serve as "building blocks" for composing the use cases associated with particular workflows. The Concepts reflect both the initial functional requirements defined in the IDM and their binding to an implementation, using the Industry Foundation Classes schema. Concepts are also aggregated into higher level Concepts that are also available for re-use. Only the base-level Concepts contain a mapping to modular implementation units – or bindings -- expressed in the diagrammed IFC format

In line with the practice adopted in earlier MVD projects, we distinguish three kinds of Concepts: variable, adaptor and static. The Static Concepts are re-usable bottom level mappings to IFC structures, and are re-used as needed in different contexts. Static Concepts are composed together to define the higher-level modular units (called Variable Concepts). Where needed, mid-level 'Adaptor Concepts' are used to collect Static Concepts together and associate them as groupings within the Variable Concepts. The top level Variable Concepts are full object or relation definitions and bindings.

In the following diagrams, each Concept is defined in a rectangle and numbered in the upper left with prefix designating which project authored them. All the Concepts and binding defined by this project team are labeled "PCI". The numbers for the PCI Concepts may change over time, as they are re-assigned in the IFC Solutions Factory as they are uploaded.

Globally, Concepts are represented in a lattice, where different Concepts are re-used in different workflows. This would be too complex to diagram. So top level Concepts are defined into lower level ones that identify a context for use. These define the function and semantics required for the medium level Concepts .These are called Adapter Concepts. Sometimes, special use or “business rules” apply to the Adapter Concepts In most places there are multiple levels of Adapter Concepts. The variable and adapter Concepts provide good reference to the information concepts making up a particular use case or Exchange Model, for review by the domain users.

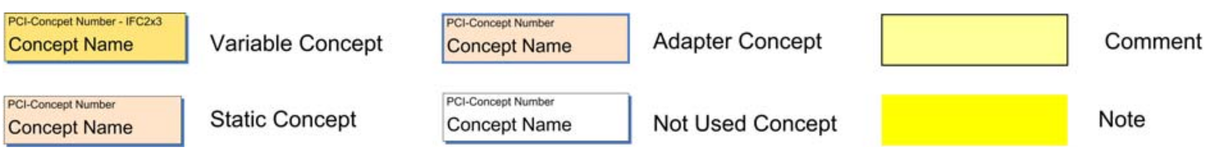
At the bottom are Leaf Concepts and these have IFC bindings. They identify the specific IFC constructs used to implement the Leaf Concepts. These are of critical importance to software companies and their implementers.

There are often multiple Variable or Leaf Concepts to represent an object or shape or properties, for use in different workflow contexts. These may represent alternative geometric representations to use in different contexts, or different property sets. These are also grouped together in this report, so the variations in implementation and intended use can be more easily seen.

Most of the bindings identified here use IFC constructs from Release 2.x4. Some of the bindings identified in this report are based on IFC entities that have only recently been added to the draft IFC 2x4 release.

Because of this structure, we have grouped the high level variable and adapter Concepts with this Overview, for use by all users and implementers. The Precast Bindings, including the new ones developed by this group as extensions to IFC Release 2.x3, are included in Volume Two. The Concepts defined previously and available from other specification groups from the IFC Solution Factory, are replicated for implementers in Volume Three.

Concepts have the following coding:



3. Exchange Model Views

In the next version of this report, when all the Concepts have been defined, this section will include the mapping of Concepts to EMs, identifying the information contents to be supported for each use case defined in the IDM. This version omits these mappings because the Concepts are incomplete.

General MVD Diagrams not Specific to Precast

The MVD specifications in this section are part of all IFC Exchanges. They place all objects in a building Spatial Composition structure used to access and structure a project. The section also defines project-level layout grids that may be used in the placement of all systems, and the key elements of metadata that are commonly used (such as review and approval status for precast pieces),

The following set of diagrams first deals with overall project information, not specific to precast. The diagrams define an overall spatial access hierarchy specified within the IFC structure. This is Project – Site – Building – Building Storey – Space, which have been defined by others.

The building is the top level aggregator for different systems – structural, HVAC, etc. Thus it carries references to the precast pieces through this structure.

Storey and Spaces are the means to identify building spaces so as to represent their live loads and allow those loads to be associated with the appropriate elements and nodes of the structure.

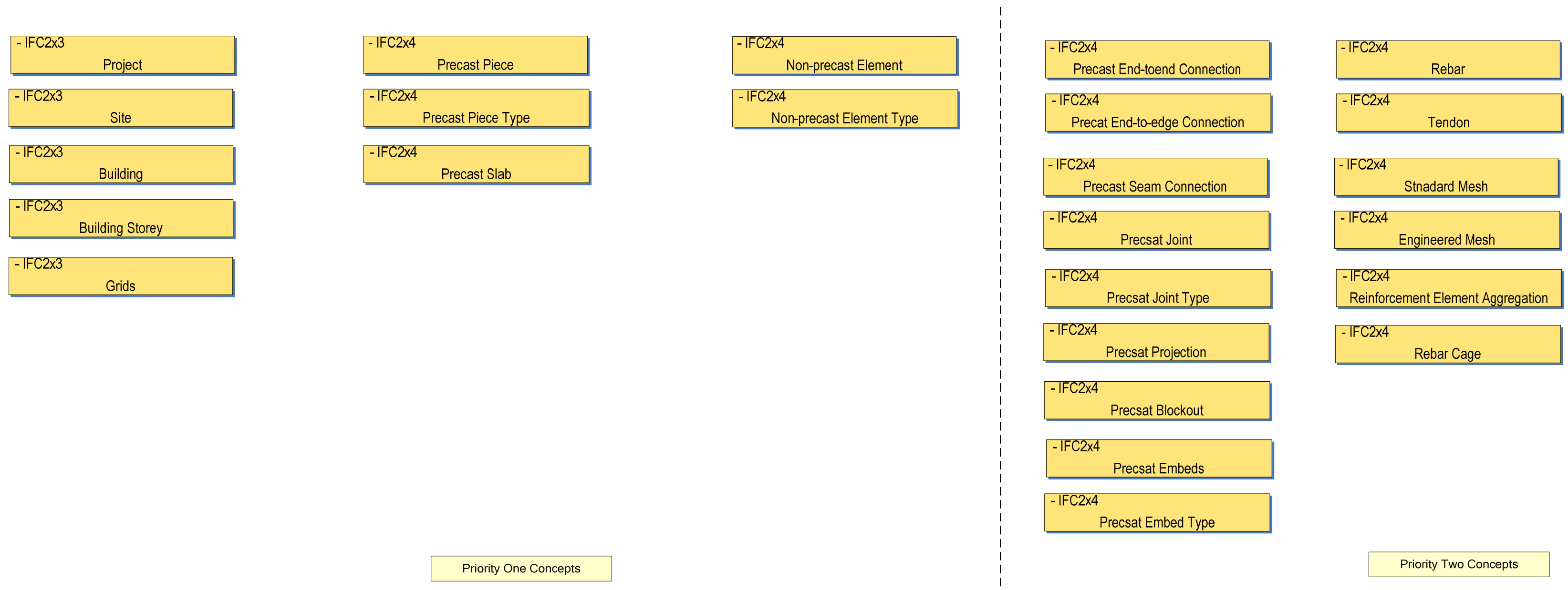
Next, the diagrams define the grid structures that may be used for relative placement.

Finally, they define metadata, for addressing who is the author on objects, and provide space for status information.

Like all MVD diagrams, each model view includes a reference to the IDM requirements, has a description as an MVD Concept Diagram. The IFC2x4 Binding Documents are available in Volumes II and III.

IFC Model View Definition Diagram : [PCI-001-IFC2x3] Precast Concrete

APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM AUTHORS
Generic	N/A	Generic	Rafael Sacks, Chuck Eastman and Ivan Panushev



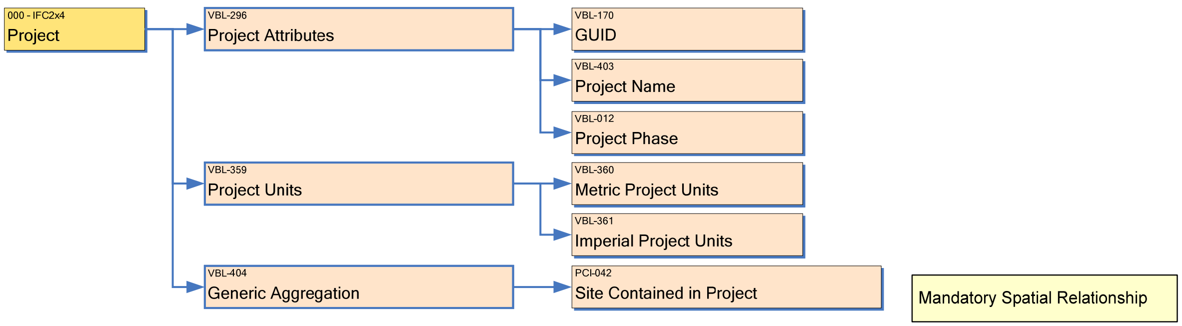
Document owner GT and Technion

This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org) The content of this document has to be certified by the IAI before becoming part of an official IFC Model View Definition.

IFC Model View Definition Diagram : Project IFC2x4

COVER PAGE

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
PCI-001	Precast Concrete	Generic		Generic	Draft	1	16.3.2009	Rafael Sacks, Chuck Eastman and Ivan Panushev

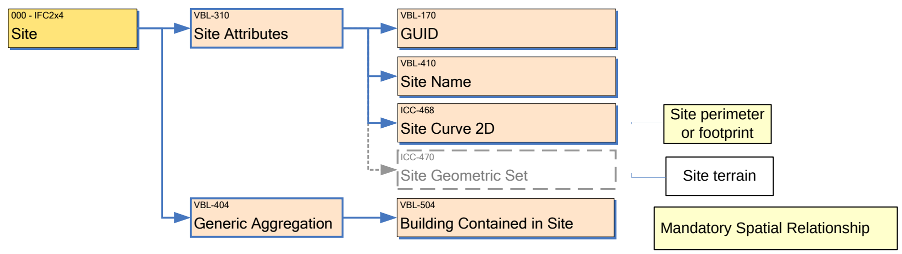


Document owner: GT and Technion -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Site IFC2x4

COVER PAGE

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
PCI-001	Precast Concrete	Generic		Generic	Draft	1	16.3.2009	Rafael Sacks, Chuck Eastman and Ivan Panushev

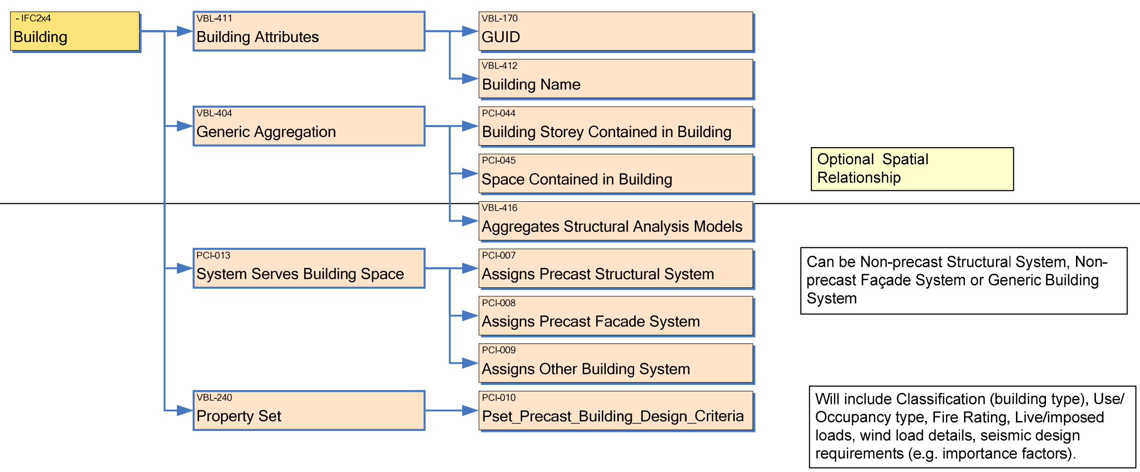


Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Building IFC2x4

COVER PAGE

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
PCI-001	Precast Concrete	Generic		Generic	Draft	1	16.3.2009	Rafael Sacks, Chuck Eastman and Ivan Panushev

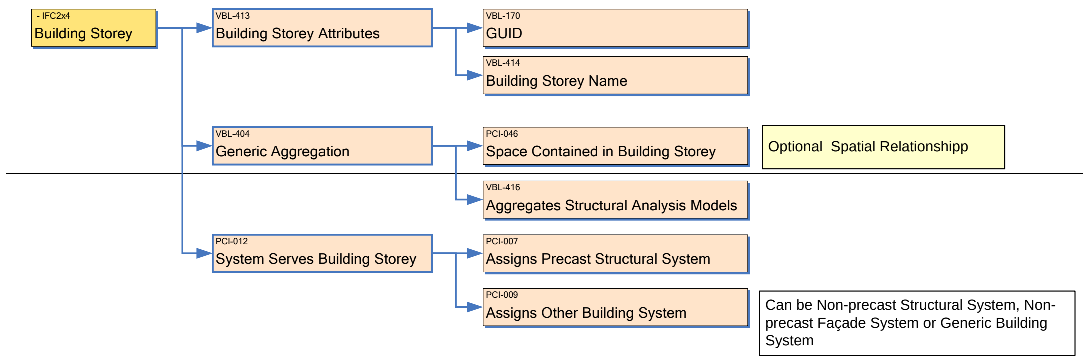


Document owner: GT and Technion -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Building Storey IFC2x4

COVER PAGE

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
PCI-001	Precast Concrete	Generic		Generic	Draft	1	16.3.2009	Rafael Sacks, Chuck Eastman and Ivan Panushev

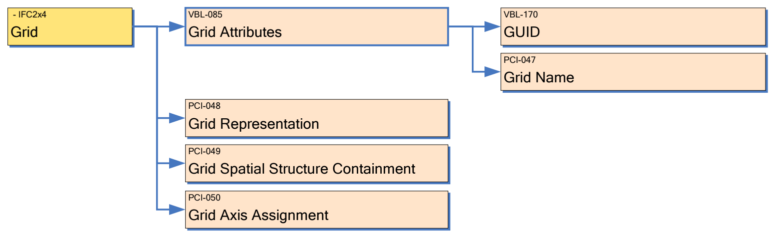


Document owner: GT and Technion -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Grid IFC2x4

COVER PAGE

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
PCI-001	Precast Concrete	Generic		Generic	Draft	1	16.3.2009	Rafael Sacks, Chuck Eastman and Ivan Panushev



Document owner: GT and Technion -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

Precast Specific MVD Diagrams

- The following section provides the specifications for precast specific model views, specified as Concepts. It is based on the IDM for Precast Concrete and includes reverences to the IDM requirements.

Each Concept Diagram describes all the Concepts needed for all its definition in different use cases. The subset of those used are defined in the EMs. The static Concepts, with their bindings are referenced alphabetically, in Volumnes II and III.

Types and Instances in the Precast MVD

In this Precast Concrete MVD we strongly support the principle of separation of types from instances. For example, a lifting hook embed is instanced in a piece by defining its location and orientation (the instance), but its geometry, material, supplier and other properties are defined by reference to two other entities: a shared lifting hook shape representation entity and a lifting hook embed type entity. This has two important benefits for the exchange:

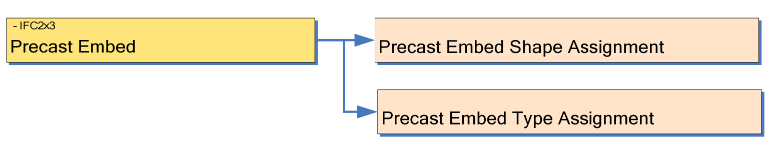
a) Reduced file size – the major part of the information for repeated occurrences of the same part need only be contained once in the exchange file;

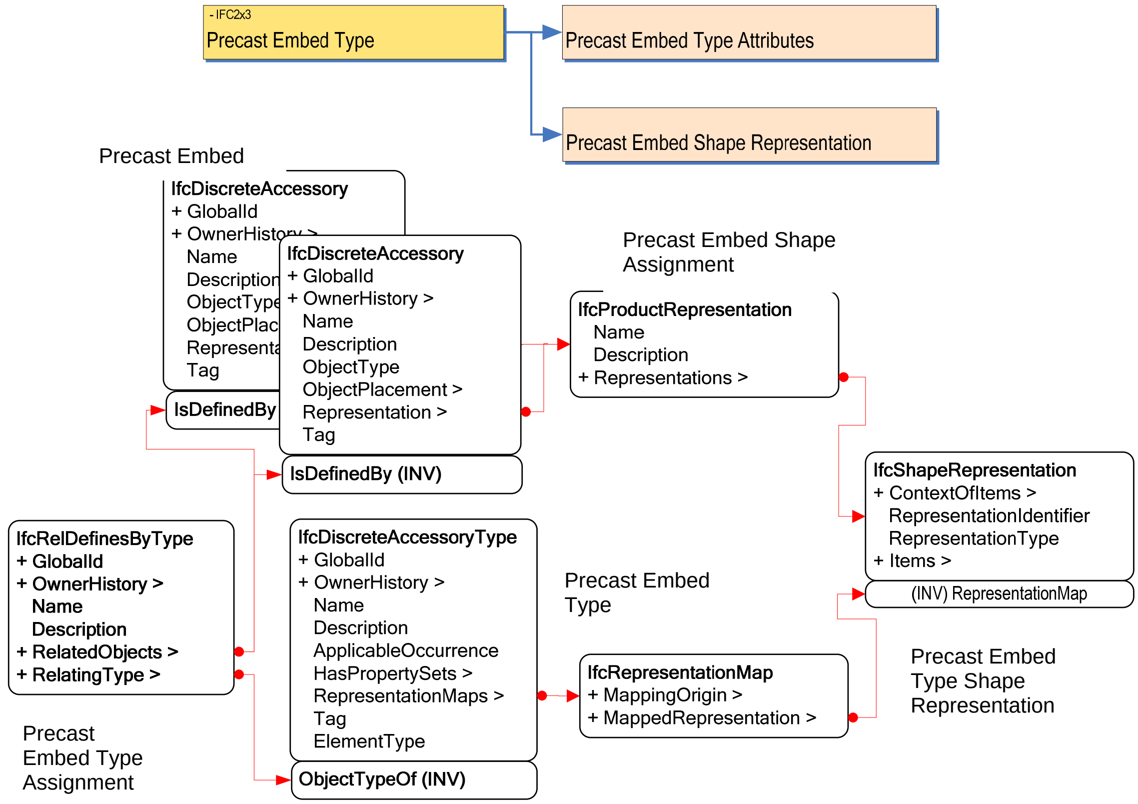
b) Increased utility of the exchange data in the importing application – the exchange file already groups identical or similar objects, which is important for most BIM functionality that involves editing or counting objects.

Types are used for the following cases in the MVD: rebars, lifting hooks, embeds, field-applied connection components, plant-applied connection components, precast pieces, joints, finishes, materials, and concrete mixes.

The general arrangement of this instance to shape and type relationship is through definition of an instance concept (e.g. a ‘Precast Embed’ bound to IFC 2x4 as an IfcDiscreteAccessory), of a shape (e.g. a BREP generic shape representation bound in IFC 2x4 as an IfcShapeRepresentation that refers to an IfcFacetedBrep) and a type concept (e.g. a ‘Precast Embed Type’ , bound in IFC 2x4 as an IfcDiscreteAccessoryType). This arrangement exploits the fact that the geometry of all objects is defined in IFC using a ‘shape representation’ entity that is separate from the object itself, but related to it. As such, multiple objects can refer to a single shape representation object.

The figure below illustrates this with an MVD concept diagram and an IFC 2x4 binding diagram. A typical embed is modeled with a concept such as ‘Precast Embed Type’, whose instances are assigned to ‘Precast Embed’ instances using the concept ‘Precast Embed Type Assignment’.





Note that this arrangement is supported throughout even for situations where there is only one unique occurrence of an object in a model or an exchange. To conform to the general structure, a type instance of the object is created even if it is only referenced by a single located instance. This is true even for piece marks and pieces.

Element types and instance is enforced if the exporting application models objects that conform to the type-instance structure. If elements are not structured in the exporting application in this manner, then this structure will not be exported.

The precast piece and the precast piece mark are a special case, because there are two more levels of typology: typical beams (or columns, spandrels, etc.) and typical cross section profiles. The four concepts are as follows:

1. A **Precast Piece** is a specific occurrence of a prefabricated concrete beam, column, spandrel or other part of a building. It has a location that is defined within the building's coordinate system and by a 'location ID'. In some cases, such as double tees that form a slab, the support conditions may be designed to induce a 'twist' or 'warp' in the piece, thus the exchange information includes provision for communicating the geometric distortion, if any, by using a precast specific property set. These are the ***instances*** that are referred to in the discussion above. As such, they do not have their own geometry, although in some cases minor local adaptations, such as a specific hole blocked-out for a pipe to pass through the piece, can be defined locally.

2. A **Piece Mark** is a unique design for a precast piece. In precast fabrication practice, it is very common that multiple pieces are identical (or nearly identical) to one another. Instead of

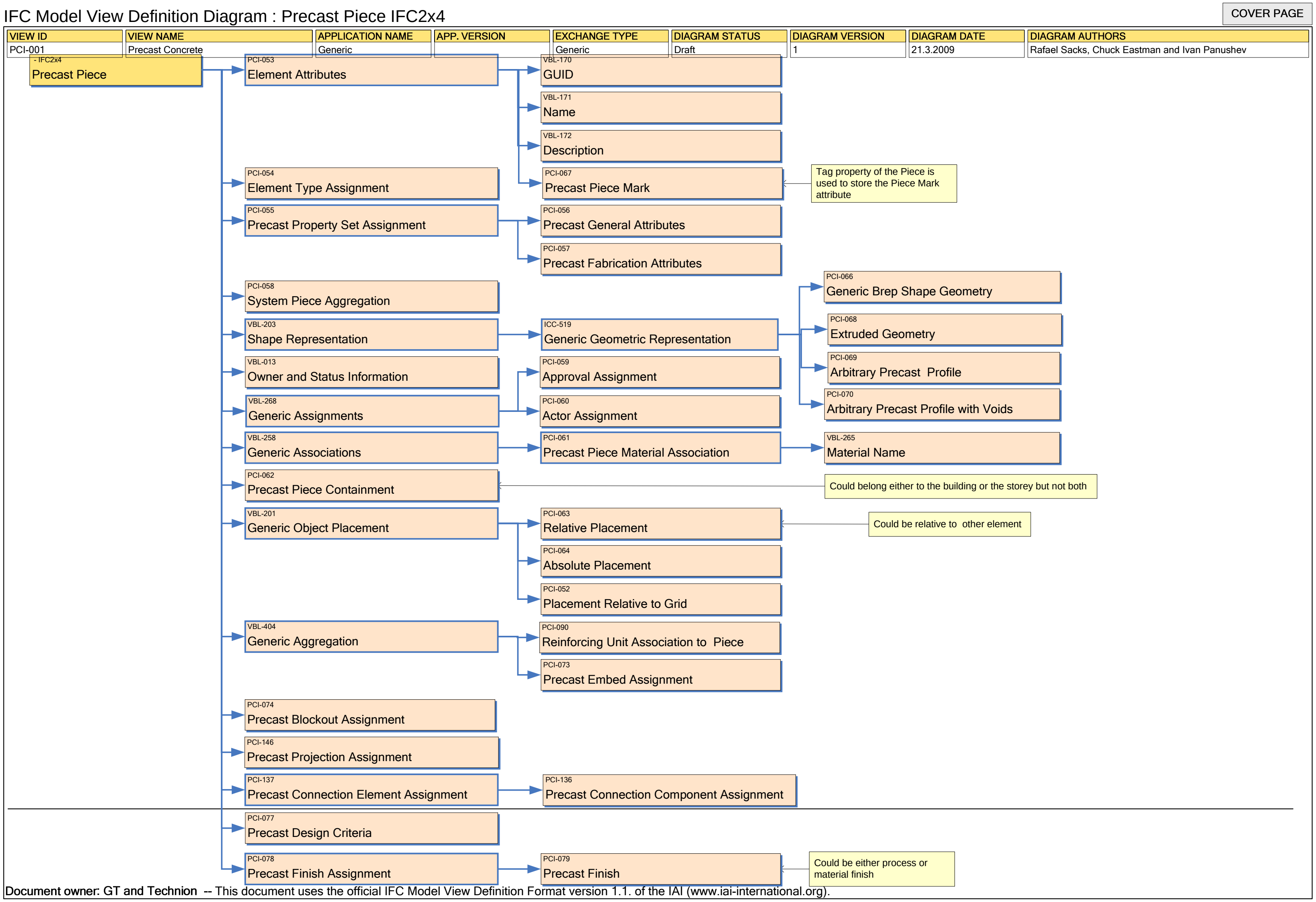
designing and drawing all of them, they are represented as a group by a single design, called a 'Piece Mark'. A 'Piece Mark' is thus a collection of pieces that can be interchanged when installed. This aggregation is made within a design application. In the exchange, it is carried as an attribute of the piece itself (bound to use the ObjectType property of the IfcBuildingElement..

3. A **Piece Type** is a type of beam, column, hollow-core panel, or other member that is commonly produced by a precast fabricator. For example, a fabricator may maintain a catalog of typical hollow-core plank sections with standard prestress tendon patterns. Another example might be a typical 20"x20" column, of 30' length, with a specific pattern of rebars stirrups; the same typical column may be used in configurations with corbels in different places, with different main rebars, etc. A designer could choose the same piece type for multiple piece marks, and multiple piece marks would share a common piece type. In an exchange, it is modeled as a Type, with reference to the discussion above. In an exchange, its properties can be detailed in one of two ways: a) by reference to an external data library that was shared among all parties to the exchange in advance; or b) by carrying explicit property set values for the piece type and by mapping to its own geometry (using a shape representation object), rebar arrangement and prestress tendon placement pattern. Alternatively, a piece mark may be defined by a piece type. In exchanges, piece types will reflect the method by which pieces and piece marks have been constructed.

4. A **Piece Profile** is a cross-section shape definition that can be commonly used for multiple piece types. They can be defined in libraries or catalogs and used as the basis for multiple piece type occurrences where extruded solid geometry is used.

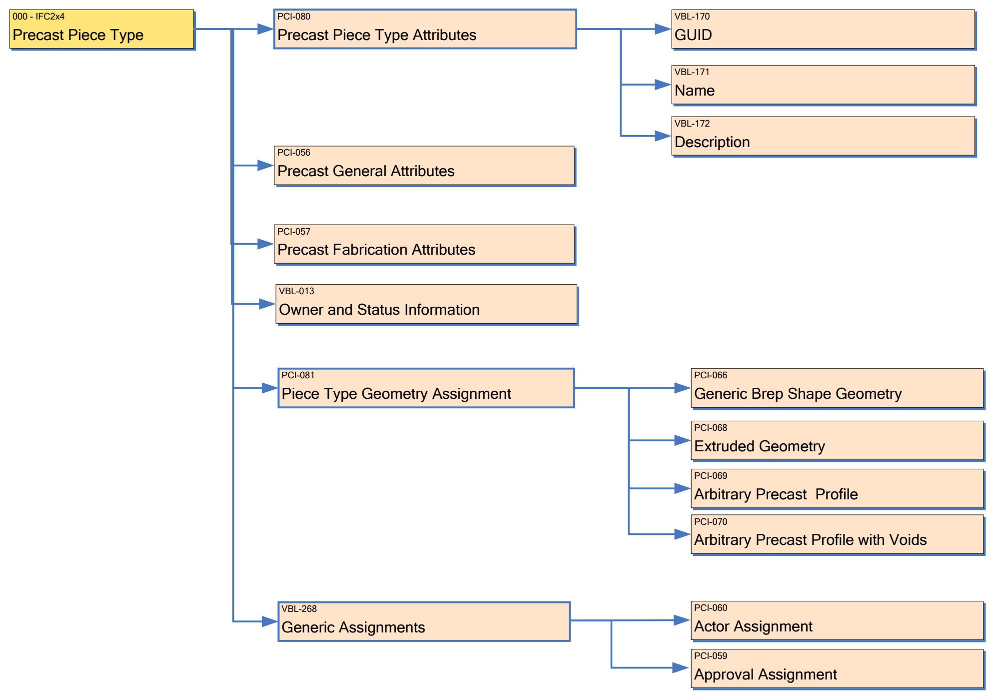
Note also that piece marks and piece types can also make reference to other types, by containing instances of embeds, rebars, lifting hooks, joints, etc. which each in turn refer to embed types, rebar types, etc.

PRECAST PIECES


IFC Model View Definition Diagram : Precast Piece IFC2x4
COVER PAGE
VIEW ID
PCI-001
VIEW NAME
Precast Concrete
APPLICATION NAME
Generic
APP. VERSION
EXCHANGE TYPE
Generic
DIAGRAM STATUS
Draft
DIAGRAM VERSION
1
DIAGRAM DATE
21.3.2009
DIAGRAM AUTHORS
Rafael Sacks, Chuck Eastman and Ivan Panushev
- IFC2x4
Precast Piece
PCI-053
Element Attributes
VBL-170
GUID
VBL-171
Name
VBL-172
Description
PCI-067
Precast Piece Mark
Tag property of the Piece is used to store the Piece Mark attribute
PCI-054
Element Type Assignment
PCI-055
Precast Property Set Assignment
PCI-056
Precast General Attributes
PCI-057
Precast Fabrication Attributes
PCI-066
Generic Brep Shape Geometry
PCI-058
System Piece Aggregation
VBL-203
Shape Representation
ICC-519
Generic Geometric Representation
PCI-068
Extruded Geometry
PCI-069
Arbitrary Precast Profile
VBL-013
Owner and Status Information
PCI-059
Approval Assignment
PCI-070
Arbitrary Precast Profile with Voids
VBL-268
Generic Assignments
PCI-060
Actor Assignment
VBL-258
Generic Associations
PCI-061
Precast Piece Material Association
VBL-265
Material Name
PCI-062
Precast Piece Containment
Could belong either to the building or the storey but not both
VBL-201
Generic Object Placement
PCI-063
Relative Placement
Could be relative to other element
PCI-064
Absolute Placement
PCI-052
Placement Relative to Grid
VBL-404
Generic Aggregation
PCI-090
Reinforcing Unit Association to Piece
PCI-073
Precast Embed Assignment
PCI-074
Precast Blockout Assignment
PCI-146
Precast Projection Assignment
PCI-137
Precast Connection Element Assignment
PCI-136
Precast Connection Component Assignment
PCI-077
Precast Design Criteria
PCI-078
Precast Finish Assignment
PCI-079
Precast Finish
Could be either process or material finish
Document owner: GT and Technion -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Piece Type IFC2x4

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		4/3/2009	

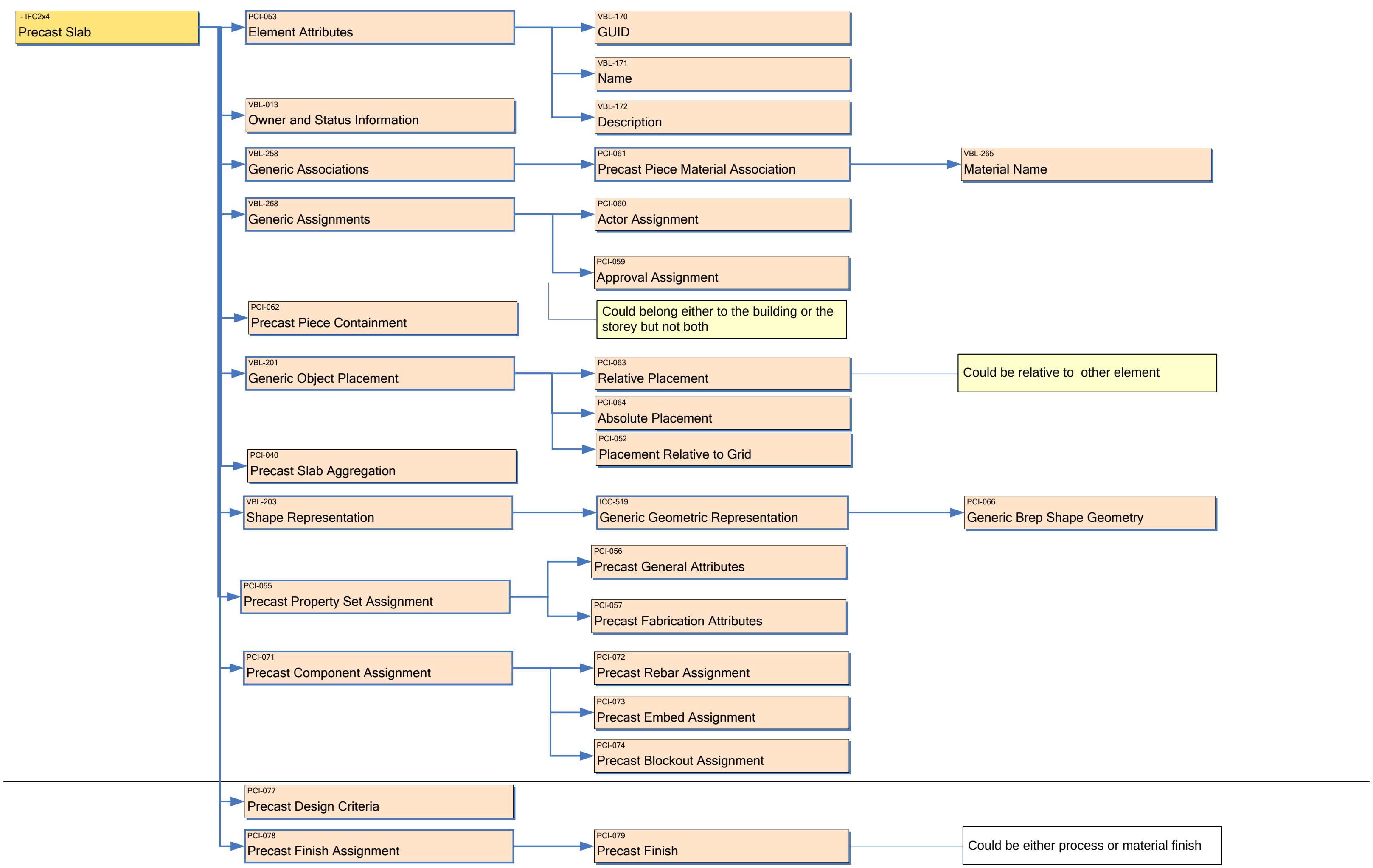


Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Slab IFC2x4

COVER PAGE

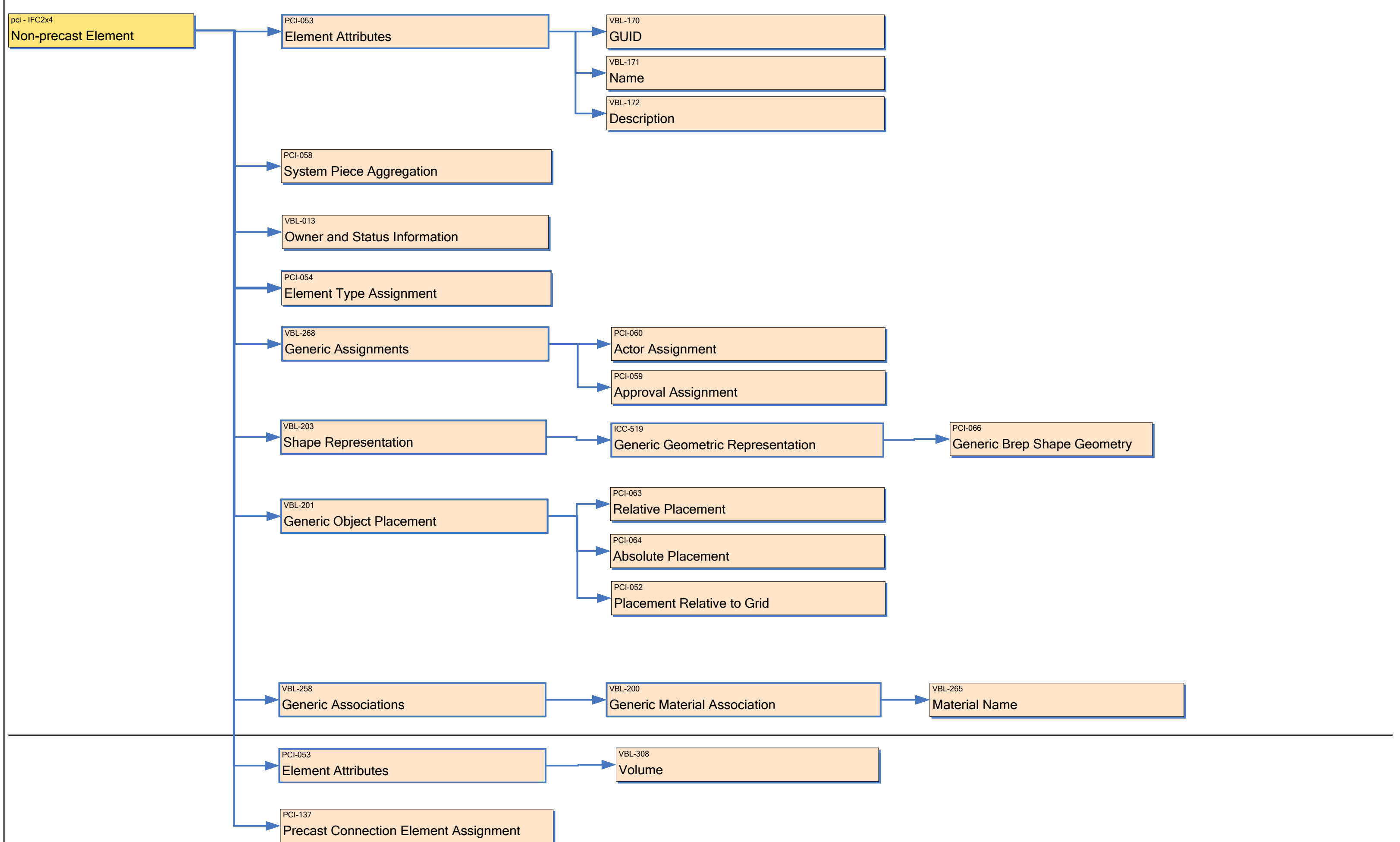
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
PCI-001	Precast Concrete	Generic		Generic	Draft	1	21.3.2009	Rafael Sacks, Chuck Eastman and Ivan Panushev



Document owner: GT and Technion -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Non-precast Element IFC2x4

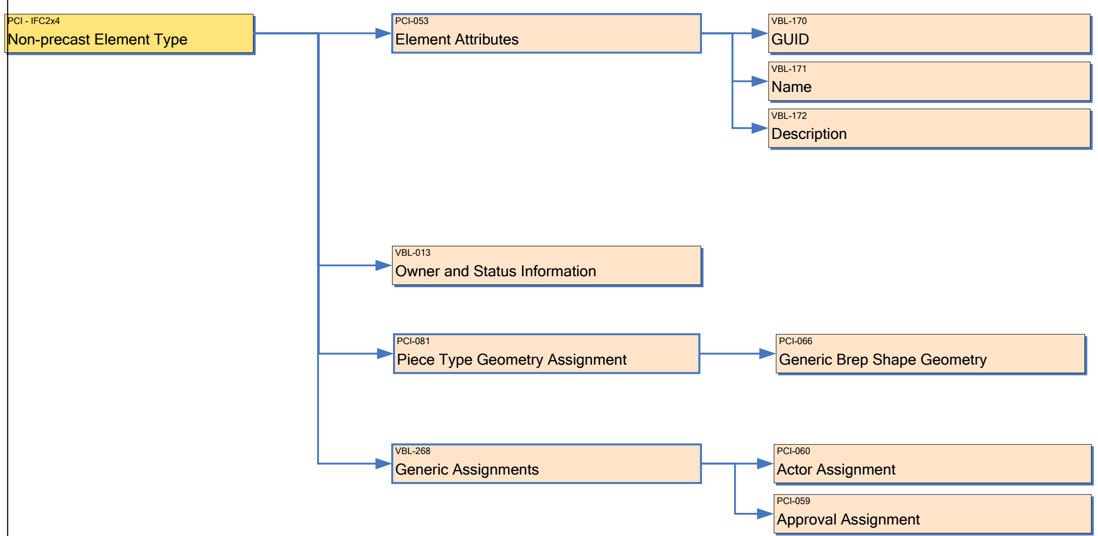
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		5/14/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Non-precast Element Type IFC2x4

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		5/14/2009	



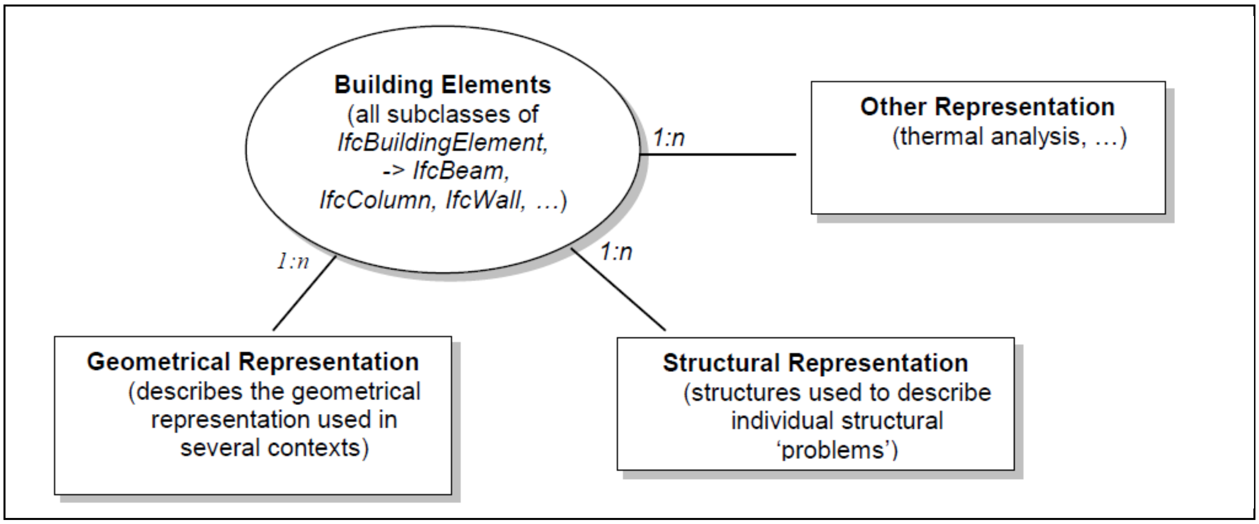
Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

Structural Analysis Model

Correlation of building elements and their actual mechanical representations allows integrating the architectural model with the structural analysis model. This integration allows establishing the relationship between the building configuration and its corresponding structural behavior.

The precast designers are not interested in the full range of structural information. So, this report focuses on the information about the structural model that is required to fully understand the design intent in a precast model. This commonly needed information is separated out from the specialized structural domain layer. For this purpose, we have referred to the work done on Structural Analysis domain by the Virtual Building Laboratory (VBL) group. A more detailed description and binding documents can be obtained from VBL under the IFC Solutions Factory (http://www.blis-project.org/IAI-MVD/).

The following figure (Figure 1) shows how the integration of structural analysis domain is intended with



other domains using IFC, with Building Elements being the core concept.

FIGURE 1 Integration of Structural Analysis Domain with other Domains using IfcBuildingElements as a core concept (Structural Analysis Extension for IFC, 2002)

Structural Analysis Domain

The inheritance structure and the main attributes and relationships defined in the structural model are depicted in the following EXPRESS G diagram (Fig

where is this relation?

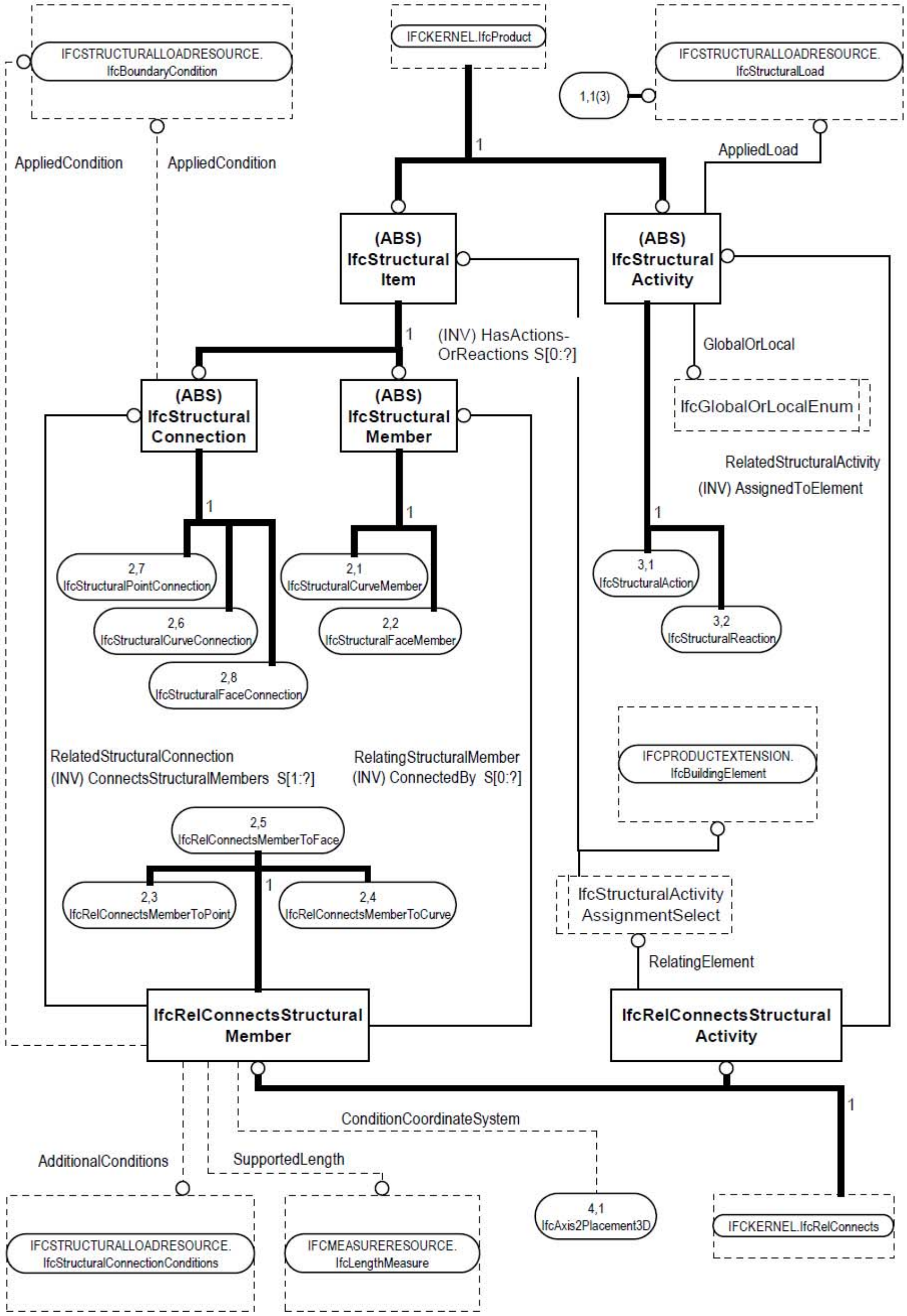


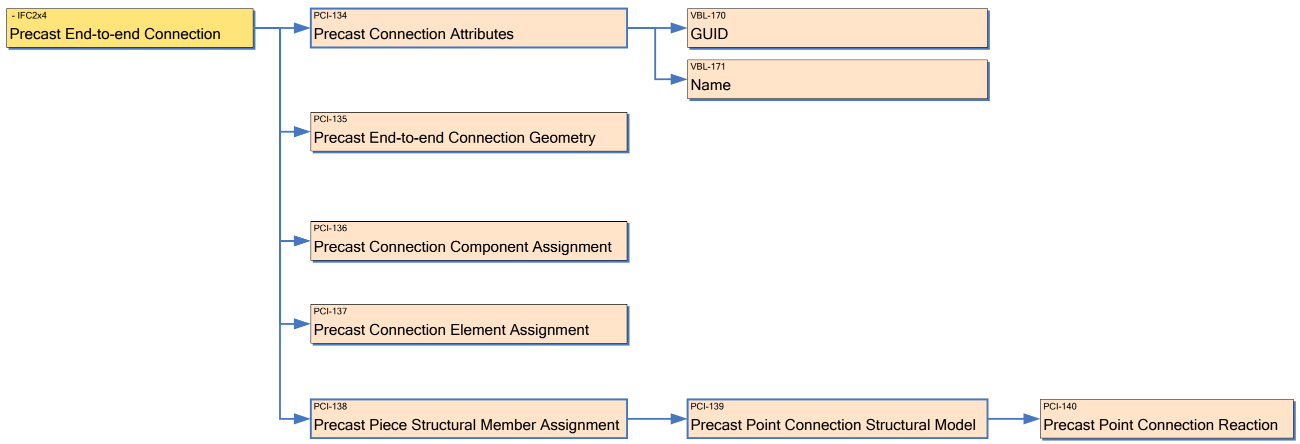
FIGURE 2 EXPRESS G diagram showing the inheritance structure and main relationships and attributes defined in the structural analysis model.

Explanation of the inheritance structure from Structural Analysis Extension for IFC, 2002

The two top-level entity classes *IfcStructuralItem* and *IfcStructuralActivity* both inherit their properties from the kernel entity class *IfcProduct*. In this way, general representational features such as *placement*, s*hape* and other basic product data attributes are consistently provided to all structural analysis entities in the same manner as for all other tangible IFC objects.
IfcStructuralItem is the root entity class for all classes representing structural objects, whereas *IfcStructuralActivity* is the root class for the objects representing the external impacts (loads and other actions) on these objects.
IfcStructuralItem is further specialized in two main branches – structural members and structural connections which are linked through the relationship class *IfcRelConnectsStructuralMember*, subtype of the kernel class *IfcRelConnects*. In this way, the connectivity of the bearing structure is explicitly established. Structural members are further subtyped into linear and planar elements, and structural connections into point, line / curve and face / planar connections respectively. In addition to the general shape and location definitions provided through the inheritance from platform entities, these classes also have a topological representation. This is more efficient for structural analysis applications. Furthermore, connections can optionally be associated with boundary conditions that may specify node and edge restraints in terms of linear and rotational stiffness, as well as warping. As mentioned before, finite elements (which are a pure numerical abstraction) are not part of the model.
IfcStructuralActivity is specialized in the two branches *IfcStructuralAction* and *IfcStructuralReaction*, allowing to differentiate between external and internal actions. External actions are further on subtyped into single (point) loads, linear loads, planar loads (pressure) and enforced displacements. They can be applied both to members and connections as well as to building elements.
The relationship between the structural elements (members, connections) and the actions upon them is established through the *IfcRelConnectsStructuralActivity* class which is also subtyped from the *IfcRelConnects* class. Additionally, each action is also related to an *IfcStructuralLoadGroup* which provides the grouping of actions into load groups, load cases and load combinations. To facilitate re-using load definitions in other structural domain extension models, the load quantities themselves are defined as a specific “structural” resource. In a similar way, material and sectional properties are made available by importing and appropriately extending the respective IFC platform resource schemas.

IFC Model View Definition Diagram : Precast End-to-end Connection IFC2x4

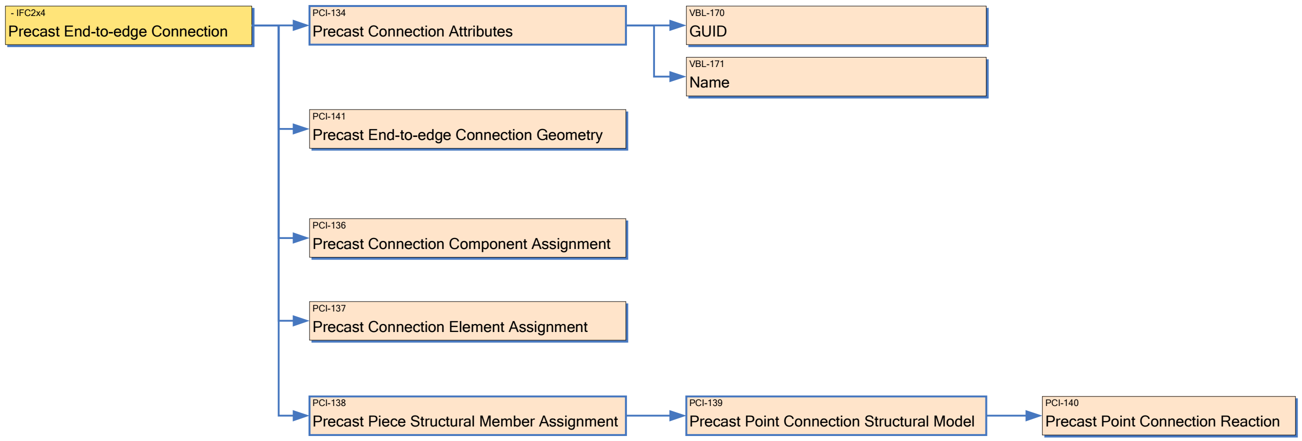
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
		Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast End-to-edge Connection IFC2x4

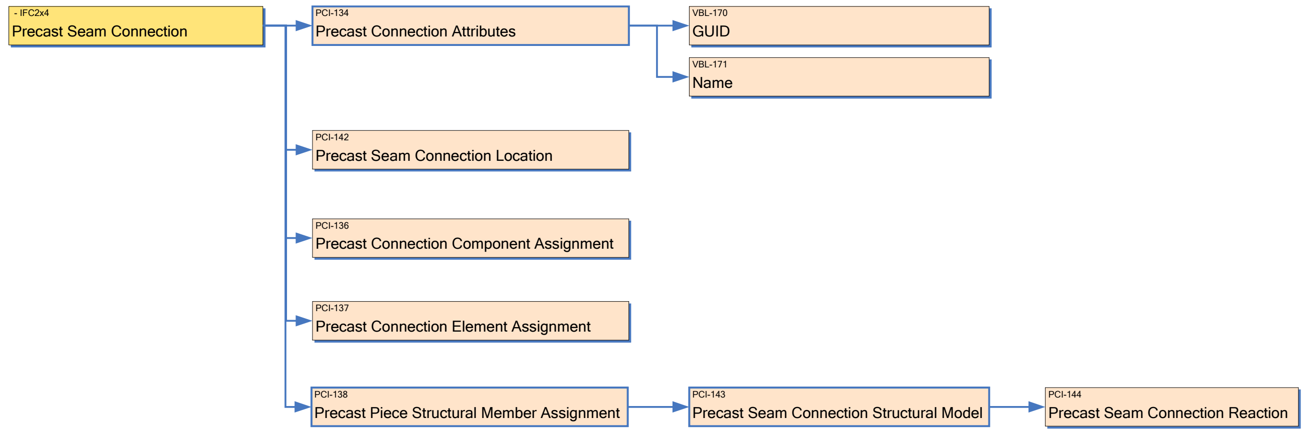
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Seam Connection IFC2x4

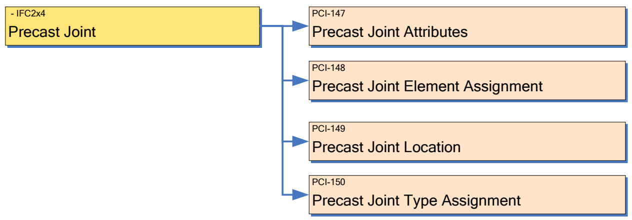
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Joint IFC2x4

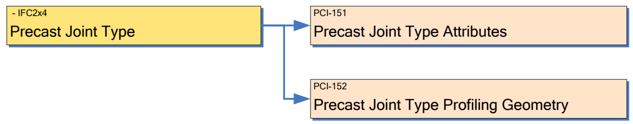
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Joint Type IFC2x4

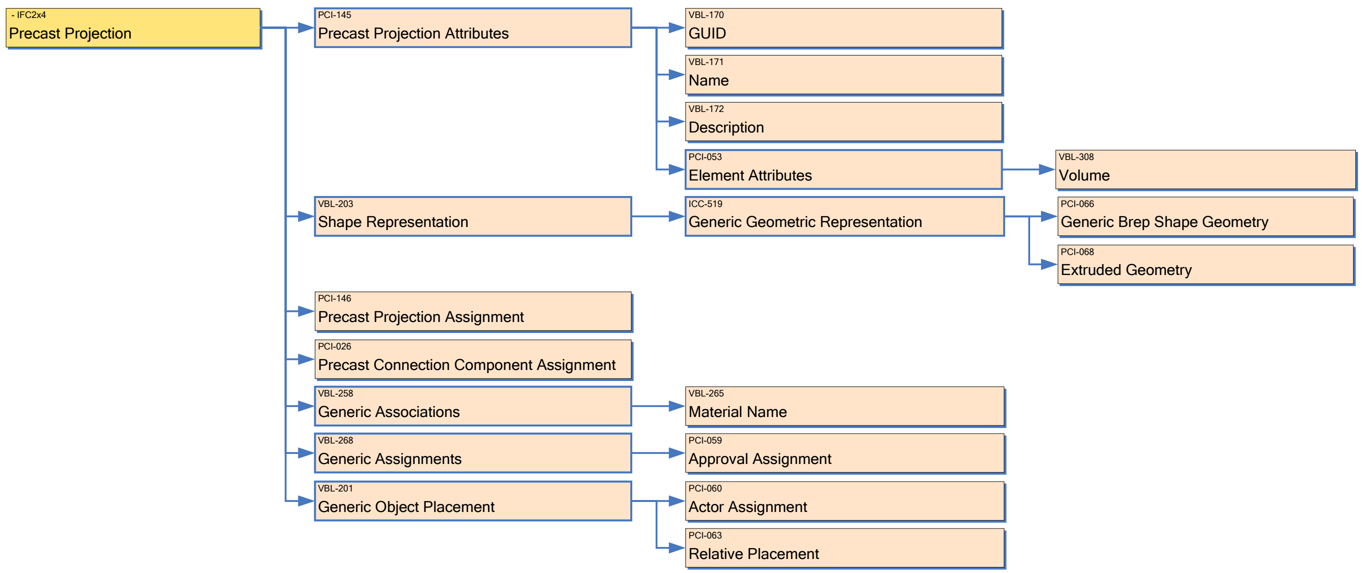
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Projection IFC2x4

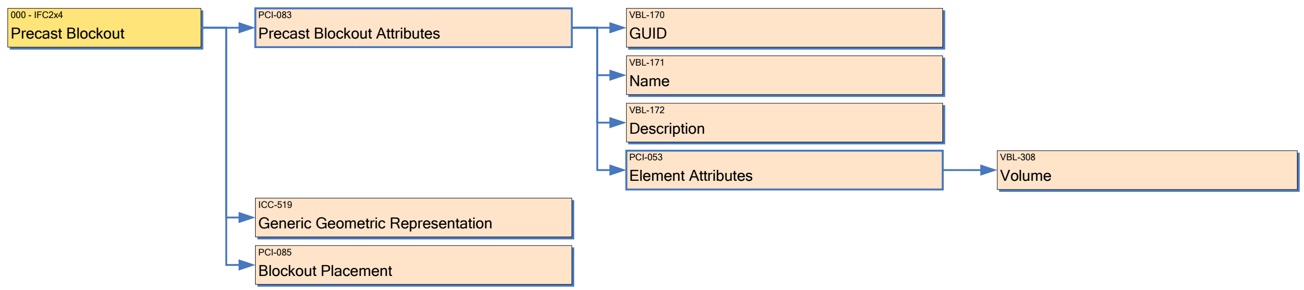
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Blockout IFC2x4

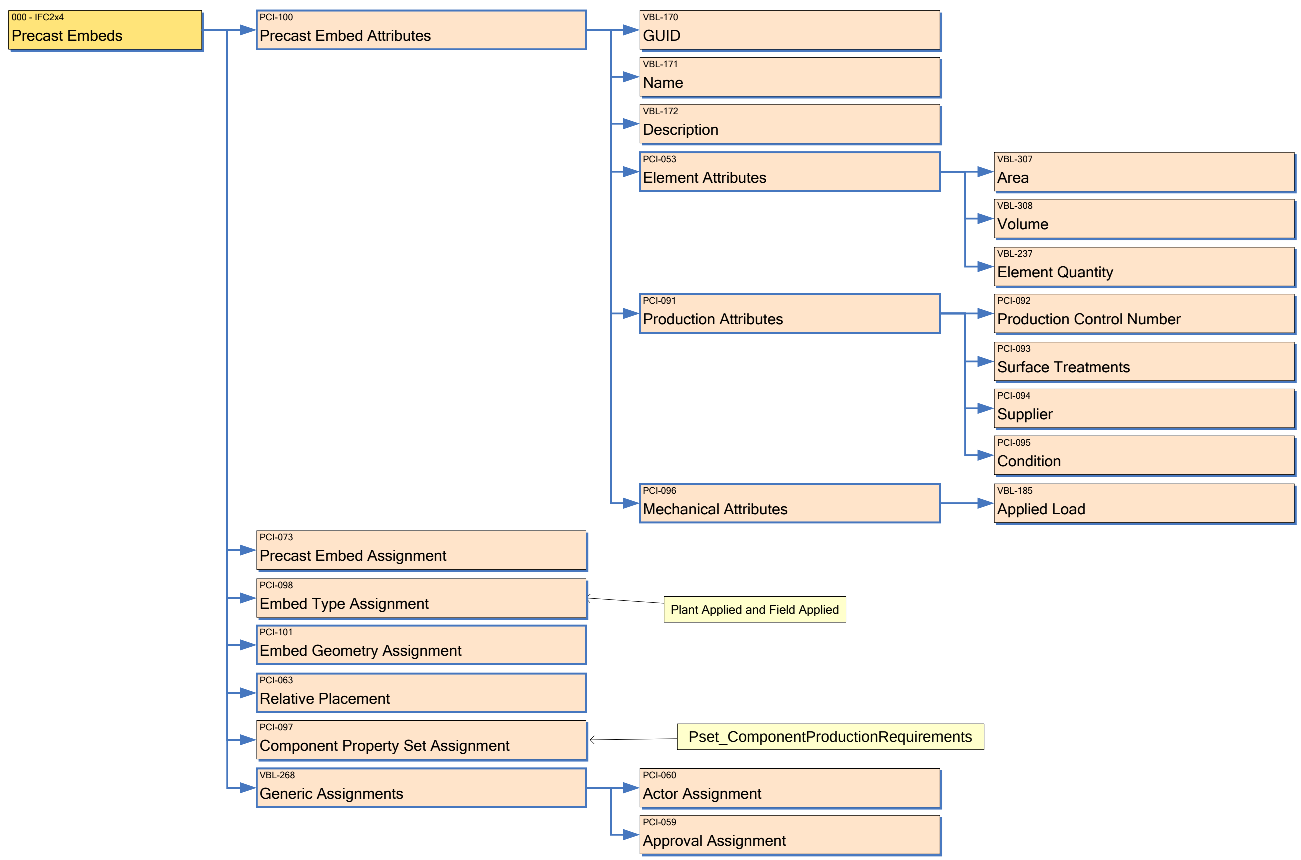
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Embeds IFC2x4

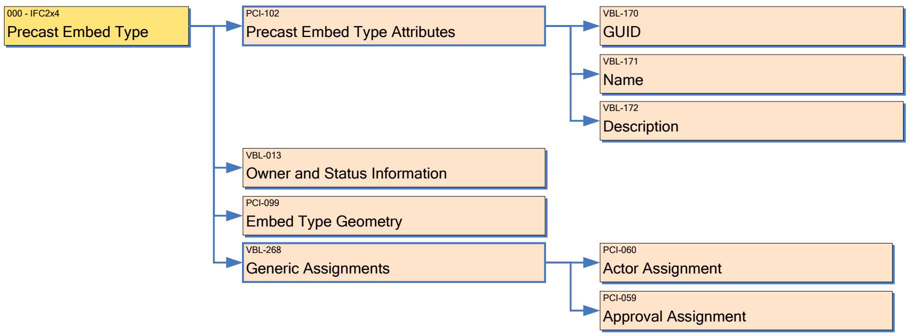
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Precast Embed Type IFC2x4

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		8/21/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

Precast Reinforcement

From the interoperability point of view the goal in this effort is that the data extraction module automatically extracts the data needed for manufacturing reinforcing elements and units from the design.
In order to create the MVD documents reinforcing elements are expressed using six coneptss:

(1) Reinforcing Bar: A steel bar, with manufactured deformations in the surface

(2) Tendon: Pretensioned, Usually from 7-wire strands

(3) Standard Mesh: A series of longitudinal and transverse wires or bars of various gauges, arranged at right angles and welded together

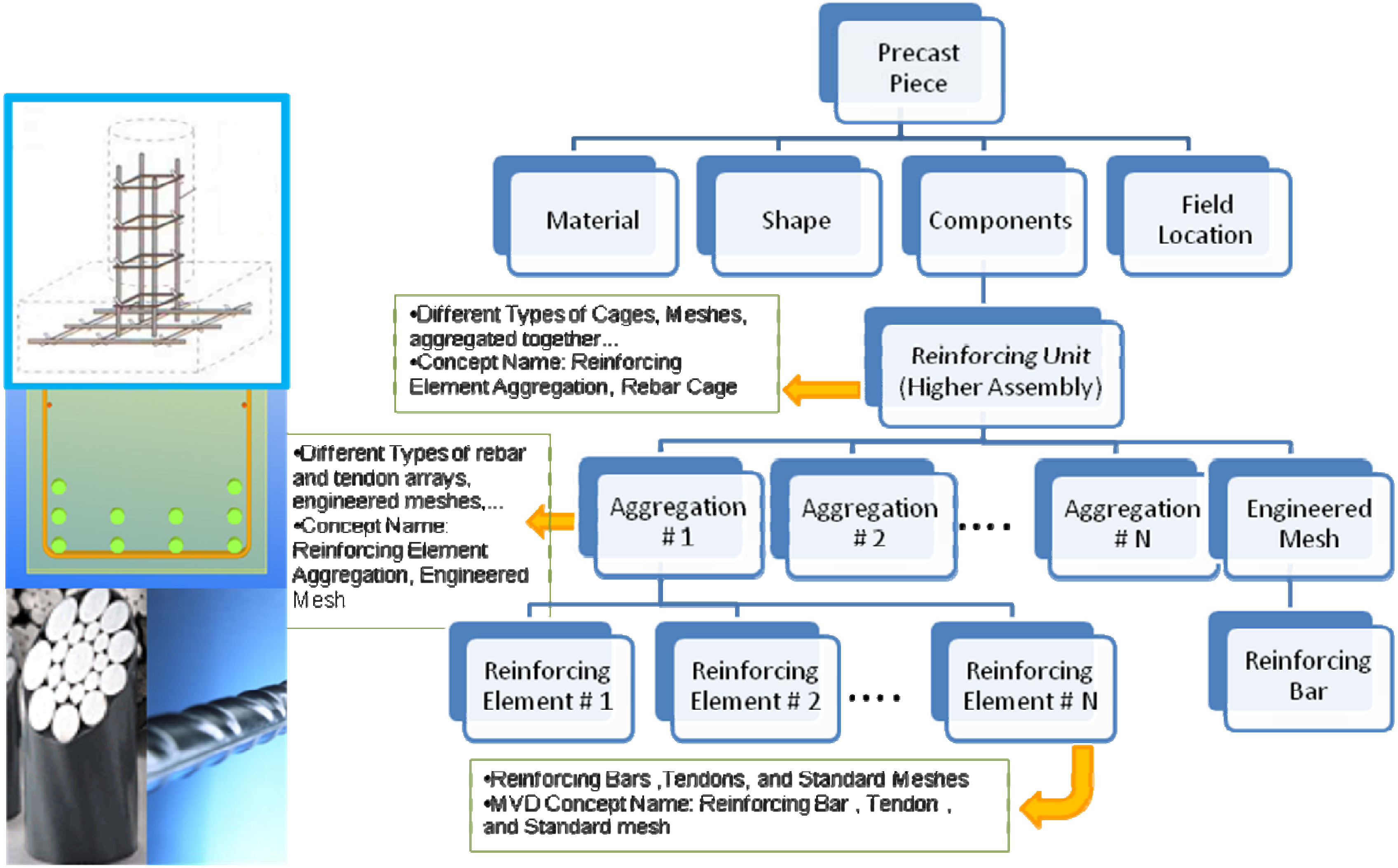
(4) Engineered Mesh: To create custom designed mesh with different spacing and size among the longitudinal and transverse bars, the possibility of bending the bars, having custom peripheral shapes, and the project-specific sheet size

(5) Reinforcing Element Aggregation: Various arrays of tendons, rebar, and standard meshes

(6) Rebar Cage: Three dimensional assemblies of longitudinal and transverse bars. A rebar cage is considered to be a higher level assembly of rebars than a reinforcing element aggregation.

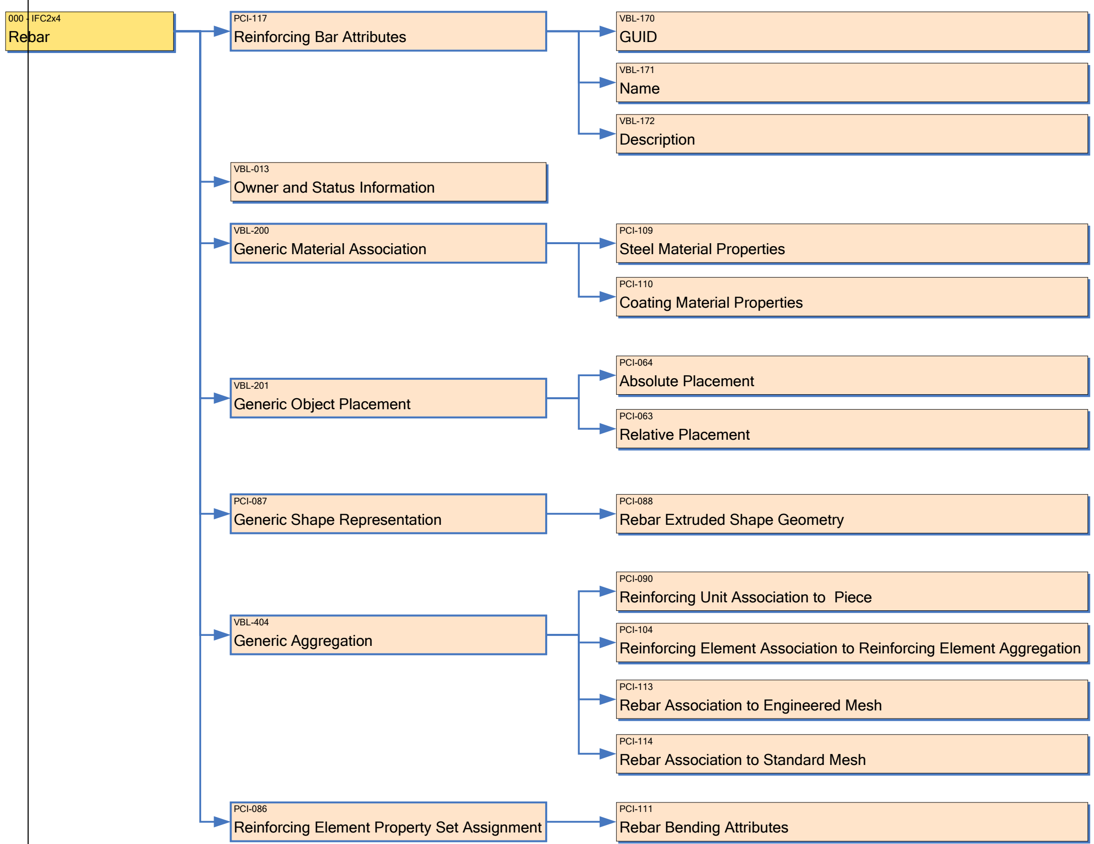
In this effort, because of the binding to the IFC 2x3, the existing entities in IFC2x3 were used, which applied some limitations to modeling of all combinations of reinforcing elements in the form of reinforcing units.
Regarding the reinforcement, the main available entity in the IFC 2x3 is the abstract entity IfcReinforcingElement which has the following subtypes: IfcReinforcingBar, IfcReinforcingMesh, IfcTendon, and IfcTendonAnchor. These entities don't cover various reinforcing units like groups or sets of rebars and tendons (which we call "Reinforcing Element Aggregations"), engineered mesh and rebar cages. So in developing the MVD concepts for these elements, the required attributes of individual entities that build up these reinforcing units are aggregated. This imposes some limitations.
On the other hand, the type and instance distinction has not been made explicit in the IFC 2x3 reinforcement-related entities. Therefore, the MVD concepts are defined in a way that users can choose to apply this distinction. For instance, they can define the geometry and shape representation information once for any rebar type and then assign it to the number of instances of that type in different aggregations. This way the users' approach to the work determines using type and instance distinction.
Given the lack of development of the type instance formalisms in IFC 2x3, there is also no provision in the MVD for rebar or tendon 'patterns', which are typical groupings of rebars or tendons that a plant might offer as a standard library. It is therefore not possible to carry standard patterns, applied to concrete sections, by using of library references alone.

Hierarchy of Reinforcing Elements and Units



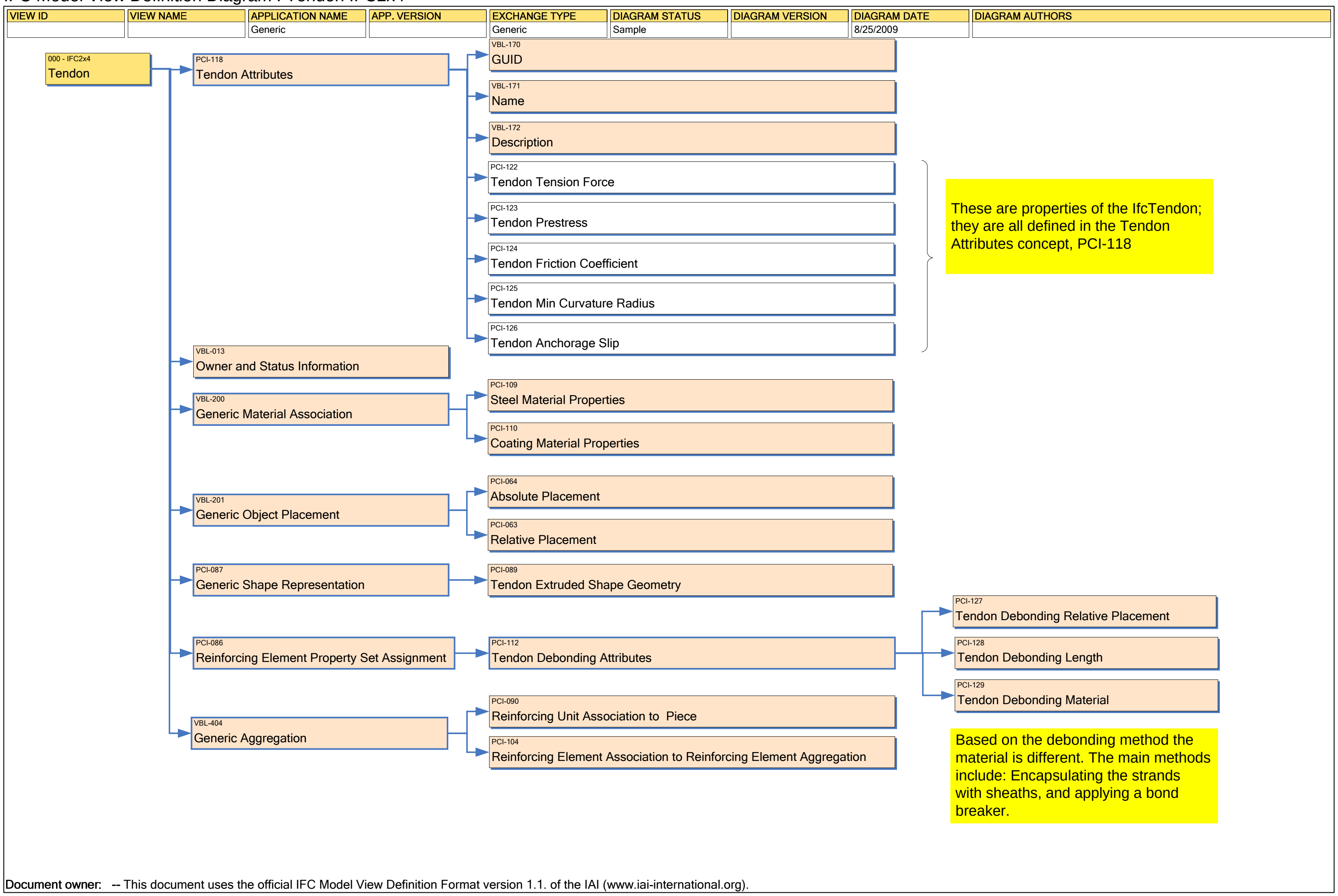
IFC Model View Definition Diagram : Rebar IFC2x4

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
	Generic	Generic		Generic	Sample		4/13/2009	



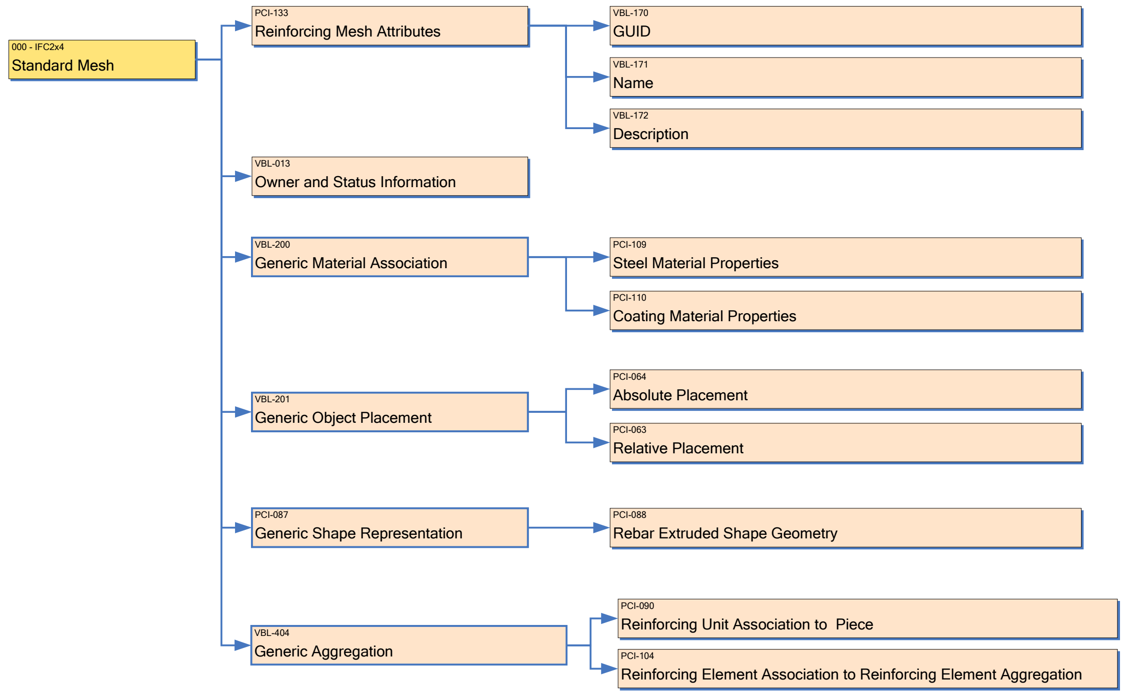
Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Tendon IFC2x4



IFC Model View Definition Diagram : Standard Mesh IFC2x4

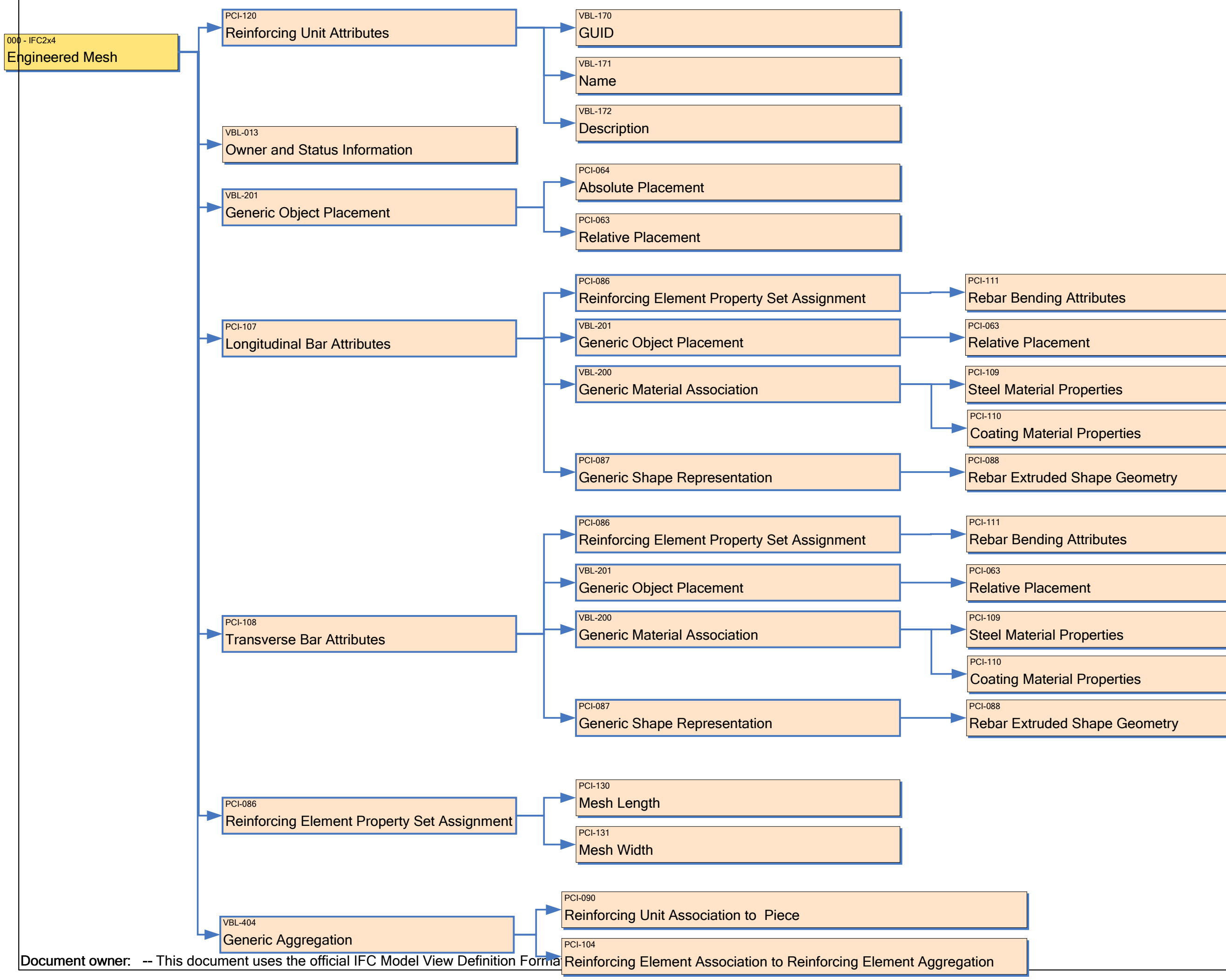
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
		Generic		Generic	Sample		8/25/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Engineered Mesh IFC2x4

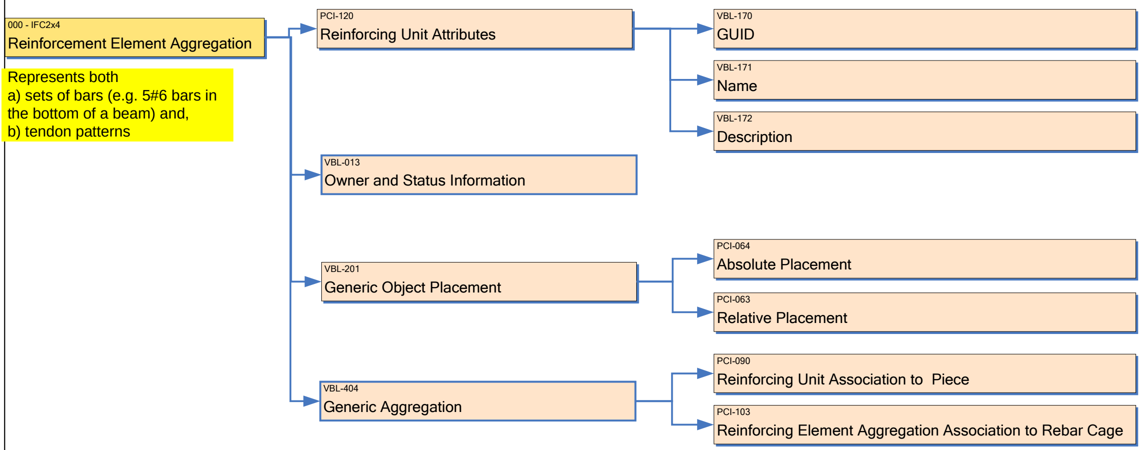
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
		Generic		Generic	Sample		8/25/2009	



Document owner: -- This document uses the official IFC Model View Definition Forma

IFC Model View Definition Diagram : Reinforcement Element Aggregation IFC2x4

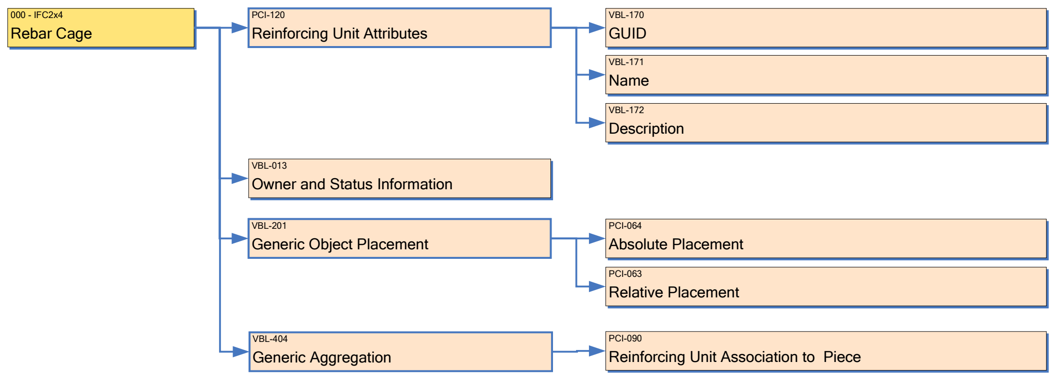
VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
		Generic		Generic	Sample		8/26/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

IFC Model View Definition Diagram : Rebar Cage IFC2x4

VIEW ID	VIEW NAME	APPLICATION NAME	APP. VERSION	EXCHANGE TYPE	DIAGRAM STATUS	DIAGRAM VERSION	DIAGRAM DATE	DIAGRAM AUTHORS
		Generic		Generic	Sample		8/26/2009	



Document owner: -- This document uses the official IFC Model View Definition Format version 1.1. of the IAI (www.iai-international.org).

Appendix A: Building Entity Subtypes for Precast Pieces and Precast Piece Types

The table below defines the ways in which all precast pieces are to be modeled using IFC 2x4.

The identities of specific precast piece types are set using a combination of the IFC entity type and the value of the *PredefinedType* property of the Building Element Type entity associated with the Building Element itself.

Precast Pieces	**IFC Entity to be used** (child entities of IfcBuildingElement)	**'Entity Type' property value or enumerated type values**
BEAMS Box Beam Rectangular Beam Transfer Beam Bridge Segment	IfcBeam IfcBeamStandardCase	Must refer to an instance of IfcBeamType using IfcRelDefinesByType which must have the property `PredefinedType = BEAM`
Hollow Core plank Double Tee	IfcBeam IfcBeamStandardCase	Must refer to an instance of IfcBeamType using IfcRelDefinesByType which must have the property `PredefinedType = SLAB_PART` (IFC 2x4 Beta) `PredefinedType = JOIST` (IFC 2x3)
Inverted Tee Beam L Beam	IfcBeam IfcBeamStandardCase	Must refer to an instance of IfcBeamType using IfcRelDefinesByType which must have the property `PredefinedType = SLAB_SUPPORT_BEAM` (IFC 2x4 Beta) `PredefinedType = T_BEAM` (IFC 2x3)
Structural Spandrels (Pocket, Ledge, Button Haunch, Deep Beams)	IfcBeam IfcBeamStandardCase	Must refer to an instance of IfcBeamType using IfcRelDefinesByType which must have the property `PredefinedType = SPANDREL` (IFC 2x4 Beta)
DECORATIVE PIECES Decorative Panel Column Cover Mullions Head Sill Cap (pieces with no structural function)	IfcCovering	Must refer to an instance of IfcCoveringType using IfcRelDefinesByType which must have the property `PredefinedType = CLADDING` Must be related to a child entity of IfcBuildingElement using IfcRelCoversBldgElements
Column Drilled Pier	IfcColumn	-
Core wall Pilasters Insulated wall pieces Stem Wall	IfcWall	Must refer to an instance of IfcWallType using IfcRelDefinesByType which must have the property `PredefinedType = STANDARD`
Lite-wall Shear wall K-Frames Pilasters	IfcWall	Must refer to an instance of IfcWallType using IfcRelDefinesByType which must have the property `PredefinedType = SHEAR`

Precast Pieces	**IFC Entity to be used** (child entities of IfcBuildingElement)	**'Entity Type' property value or enumerated type values**
Hollow-core walls Retaining Wall	IfcWall	Must refer to an instance of IfcWallType using IfcRelDefinesByType which must have the property `PredefinedType = ELEMENTEDWALL`
Flat slab piece Box slab	IfcSlab IfcSlabStandardCase	Must refer to an instance of IfcSlabType using IfcRelDefinesByType which must have the property `PredefinedType = FLOOR` or `PredefinedType = ROOF`
Stair landing piece	IfcSlab IfcSlabStandardCase	Must refer to an instance of IfcSlabType using IfcRelDefinesByType which must have the property `PredefinedType = LANDING`
Slab on grade piece	IfcSlab IfcSlabStandardCase	Must refer to an instance of IfcSlabType using IfcRelDefinesByType which must have the property `PredefinedType = BASESLAB`
Stair piece with/without integral landing	IfcStairFlight	Must be aggregated in an IfcStair using IfcRelAggregates
Precast Pile	IfcPile	Must have the property `PredefinedType = FRICTION`
Caisson Unit	IfcPile	Must have the property `PredefinedType = USERDEFINED` Must have the property `ObjectType = 'CAISSON UNIT'`
Grade Beam	IfcFooting	Must have the property `PredefinedType = FOOTING_BEAM`
Spread (pad) Footing	IfcFooting	Must have the property `PredefinedType = PAD_FOOTING`
Pile Cap	IfcFooting	Must have the property `PredefinedType = PILE_CAP`
Strip footing	IfcFooting	Must have the property `PredefinedType = STRIP_FOOTING`
Pier Cap	IfcPlate IfcPlateStandardCase	Must refer to an instance of IfcPlateType using IfcRelDefinesByType which must have the property `PredefinedType = USERDEFINED` and must have the property `ObjectType = 'PIER_CAP'`
Hyperbolic parabolic shells Other roof shell pieces	IfcRoof	-
MODULES (Cell/Room, Core, Building)	IfcElementAssembly	Must refer to an instance of IfcMemberType using IfcRelDefinesByType which must have the property `PredefinedType = USERDEFINED` `ObjectType = 'PRECAST_MODULE'`

Appendix B: IFC2x4 Property Set Definition for Precast Concrete Connection Hardware and Embeds
This property set is used to identify the way that the hardware that is used in connections and joints, such as plates, shims, etc. are delivered and applied. There are five possibilities: 1) the components are cast into the cast-in-place concrete structure or 2) provided as part of or welded onto the steel structural frame; 3) they are provided loose to the site and assembled when the precast piece is connected to the structure; 4) they are attached to the precast piece in the plant and delivered with it; and 5) they are cast into the precast piece. All of these should be made explicit. This is not unique to precast concrete, but relevant for construction of steel frame structures, wood structures and even parts of CIP structures.
In addition, there are other properties that must be defined for each of the hardware pieces, such as their treatment, which may be galvanized, stainless, painted or none. This is relevant not only for discrete accessories, but also for rebars and mesh in many applications.
Both of these are achieved by adding a property set with two enumerated property types for the *IfcDiscreteAccessory* entity, as follows:

PropertySet Definition:

PropertySet Name	Pset_ComponentProductionRequirements
Applicable Entities	IfcElementComponent IfcElementComponentType
Applicable Type value	
Definition	Definition from IAI: Set of common properties of component elements (especially discrete accessories, but also fasteners, reinforcement elements, or other types of components) to express design requirements for determining production methods and tasks.

Property Definitions:

Name	Property Type	Data Type	Definition
DeliveryType	IfcPropertyEnumeratedValue	PEnum_ComponentDeliveryType • CAST_IN_PLACE • WELDED_TO_STRUCTURE • LOOSE • ATTACHED_FOR_DELIVERY • PRECAST • NOTDEFINED	Determines how the accessory will be delivered to the site.
CorrosionTreatment	IfcPropertyEnumeratedValue	PEnum_ComponentCorrosionTreatment • PAINTED • EPOXYCOATED • GALVANISED • STAINLESS • NONE • NOTDEFINED	Determines corrosion treatment for metal components. This property is provided if the requirement nees to be expressed (a) independently of a material specification and (b) as a mere requirements statement rather than a workshop design/ processing feature.

The object types are defined using the following property sets.

- Pset_DiscreteAccessoryAnchorBolt
- Pset_DiscreteAccessoryColumnShoe
- Pset_DiscreteAccessoryCornerFixingPlate
- Pset_DiscreteAccessoryDiagonalTrussConnector
- Pset_DiscreteAccessoryEdgeFixingPlate
- Pset_DiscreteAccessoryFixingSocket
- Pset_DiscreteAccessoryLadderTrussConnector
- Pset_DiscreteAccessoryStandardFixingPlate
- Pset_DiscreteAccessoryWireLoop

Accessory type	Standard type designation	Description
Shading devices:	'Shading device'	Elements specifically designed to provide shading, often fixed externally and sometimes moving (e.g. by rotation)
Corbels as separate components:	'Hidden steel corbel'	Corbel system made from steel components embedded into the master element
	'Visible steel corbel'	Corbel system made from steel components protruding from the master element
	'Visible concrete corbel'	Corbel system made as a separate precast concrete component added to the master element
Anchor bolts:	'Foundation bolt'	Fixture with bolt embedded into concrete structures for securing columns or for machine foundations
	'Ribbed bar bolt'	Fixture with bolt and one or several anchoring ribbed (reinforcement) bars embedded into concrete structures for securing columns or for machine foundations.
	'Extension bolt'	Fixture with bolt for extension joints between usually precast elements.
Connecting accessories, for example for sandwich wall panels:	'Diagonal truss connector'	A fixing device in truss form with diagonal cross bars holding two precast conrete panels together in a sandwich wall panel.
	'Ladder truss connector'	A fixing device in truss form with straight cross bars in ladder form holding two precast conrete panels together in a sandwich wall panel.
	'Panel suspender'	A straight fixing device holding two precast conrete panels together in a sandwich wall panel.
	'Bracket'	A bracket for supporting various components or distribution elements.
Electrical accessories for precast concrete elements:	'Protective plug'	Protective plug used in element for protecting electrical accessories during manufacturing, transportation and assembly.
Fixing parts:	'Standard fixing plate'	Standard fixing plate.
	'Edge fixing plate'	Fixing plate attached to the edge of

Accessory type	Standard type designation	Description
		an element.
	'Corner fixing plate'	Fixing plate attached to the corner of an element.
	'Slab fixing plate'	Fixing plate for slabs.
	'Balcony hinge'	Accessory supporting and fixing balconies.
	'Frame shoe'	Fixing shoe for frames.
	'Thermo frame'	Thermo frame.
	'Column shoe'	Fixing shoe for columns.
	'Wall shoe'	Fixing shoe for walls.
	'Fixing socket'	Fixing socket.
Joint accessories:	'Neoprene bearing plate'	Rubber plate used as a bearing in, for example, joints between column corbels and beams.
	'Working joint reinforcement'	Reinforcement accessory used in working joints.
	'Expansion joint reinforcement'	Reinforcement accessory used in expansion joints.
	'Ribbed steel bar extension'	Extension accessory made of a ribbed (reinforcement) bar used in joints.
	'Steel pin bolt'	Pin bolt used to join together, for example, columns and beams.
	'Concrete dowel'	Dowel pin used in joints.
	'Concrete groove'	A groove made in a joint.
	'Steel plate'	A steel plate used as an accessory in a joint.
	'Wire loop'	A joint connector accessory made from a wire loop.
	'Steel loop'	A joint connector accessory made from a steel bar loop.
	'Sealing strip'	A strip sealing the joint.
	'Sealing compound'	Sealing compound protecting and sealing the joint.
Lifting accessories:	'Wire lifting hook'	A lifting aid in the form of a wire loop.
	'Steel lifting hook'	A lifting aid in the form of a steel bar loop.
	'Lifting socket'	A lifting aid in the form of a socket.
	'Steel lifting anchor'	A lifting aid in the form of a steel lifting anchor.
	'Lifting hole'	A lifting aid in the form of a hole.
Accessories mainly used in the building services domain:	'Antivibration'	An isolating device to prevent other elements to be effected by vibrations.
	'Drop rod'	A length of material providing a hanging support to a bracket. Note that a drop rod is considered to include nuts and washers required for securing.
	'Duct foot'	A base support used to receive a vertical pipe (BS6100 330 3309 - duct foot).

Accessory type	Standard type designation	Description
	'Framing'	A frame placed around a penetration to prevent scraping against the building surface or structure.
	'Grommet'	An element placed within a penetration that seals the penetration for a particular reason.
	'Rack'	A set of shelving for the purposes of storage that may be freestanding or bolted to a structure.
	'Safety part'	A part, typically installed in vertical shafts at each level, to ensure safety from falling when entering the shaft.
	'Sleeve'	A thin barrier placed between a penetration and a penetrating element.
	'Support section'	A section of material that is used as an intermediate support upon which multiple brackets can be mounted.