

Migrating From IFC R2.0 to IFC2X

Draft 2

Authors: Jiri Hietanen (jiri.hietanen@solibri.com)
 Patrick Houbaux (patrick.houbaux@solibri.com)
 Reviewers Yoshinobu Adachi
 Kari Karstila
 Juha Hyvärinen
 János Maros
 Robin Drogmuller

Table of contents

Overview	2
Changes.....	2
Use of EXPRESS	2
Project structure	2
Units	3
Owner history / versioning	4
Geometry System.....	4
Property Set System.....	7
Property Objects.....	9
Calc -Attributes (Removed)	9
Grouping / Zones / Systems	10
Material Layers and Materials.....	10
Relationships	11
Building Elements.....	13
Conclusions.....	22
APPENDICES: Implementation diagrams	23
Appendix A: IFC2X Project Structure.....	23
Appendix B: Shared Geometry in IFC2X.....	24
Appendix C: Quantity Objects in IFC2X.....	25

Overview

This document describes changes between IFC R2.0 and IFC2X. The focus is on changes in the parts of the IFC model that are included in the BLIS views of IFC R2.0.

This document is not intended to be a very detailed description of the changes, but rather an overview. The purpose is to give an idea of the work involved in migrating from IFC R2.0 implementations to IFC2X implementations. The organization of the document follows roughly the concept blocks defined for IFC R2.0.

Changes

Use of EXPRESS

Nested SELECT

IfcValue is a select type that contains 3 other select types;

```
?? IfcMeasureValue
?? IfcSimpleValue
?? IfcDerivedMeasureValue
```

Note: check that your toolbox is supporting this kind of select.

Rules

There are 3 'stand-alone' rules in the schema, e.g.

```
RULE IfcSingleProjectInstance FOR
  (IfcProject);
WHERE
  WR1 : SIZEOF(IfcProject) <= 1;
END_RULE;
```

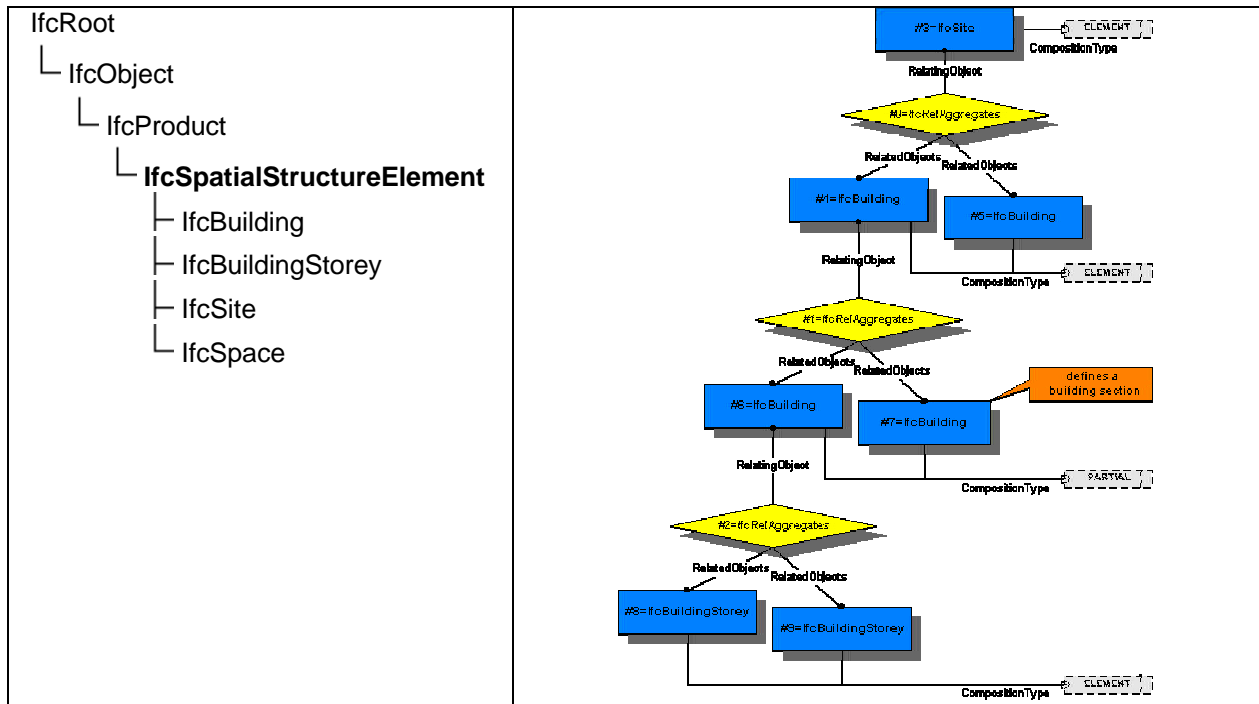
Project structure

In IFC R2.0 the project structure was fixed to: Project -> Site -> Building -> Building Storey. (This was not required by the model, it was an implementers agreement). See IfcContainmentEnum in IFC R2.0

IfcRelContains is replaced by IfcRelAggregates in which the type of the 'containment' relationship is stored in the name attribute; IfcContainmentEnum has been removed.

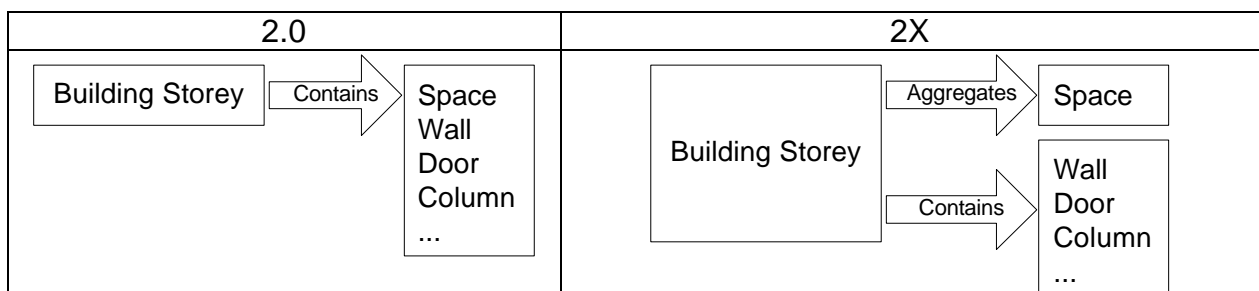
```
#1894 = IFCRELAGGREGATES ('2LeiPTHlH8nPawOIY57mA8', #6, 'BuildingContainer',
'BuildingContainer for BuildigStories', #25, (#29, #263, #927, #1595));
#1895 = IFCRELAGGREGATES ('0xZhKB0dr7wBTsw0LqA0NE', #6, 'SiteContainer',
'SiteContainer For Buildings', #23, (#25));
#1896 = IFCRELAGGREGATES ('0QP9dDvkH7_vdcz_ZpfVK$', #6, 'ProjectContainer',
'ProjectContainer for Sites', #16, (#23));
```

System in IFC2X



The `IfcSpatialStructureElement.CompositionType` defines the type of the spatial element, the values are: Complex, Element, Partial. Per IFC2X implementers agreement only Element is used.

As an end result the project structure in the IFC2X implementations is fixed to be just like it was in the IFC R2.0 implementations. The relationship names have changed and the containment relationship is different when spatial elements contain each other and when spatial elements contain building elements. See Appendix A for details.



Units

Unit Assignment

Added `IfcMonetaryUnit` but this is irrelevant for the current BLIS views.

Unit Definitions

Metric (Named Units)

No relevant changes

US Imperial (Conversion Based Units)

No relevant changes

Owner history / versioning

Owner History Assignment

No changes

Owner History

The ModifiedFlag has been removed and the ChangeAction enumeration is used instead.

Audit Trail

IfcAuditTrail has been removed from the model. All information for versioning has been re-engineered and is taking place in the IfcOwnerHistory object. E.g. the CreationDate is now on IfcOwnerHistory and is mandatory.

In the IFC R2.0 implementations the audit trail was only used for tagging deleted objects. In IFC2X deleted objects are tagged using IfcOwnerHistory.ChangeAction = DELETED.

Geometry System

Assigning Geometry to Building Elements

No relevant changes. Only some attribute names have changed.

R2.0: IfcProduct.Representations

? IfcProductDefinitionShape.**Shape**Representations

? IfcShapeRepresentation.Items

2X: IfcProduct.Representation

? IfcProductDefinitionShape.Representations

? IfcShapeRepresentation.Items

Identifying Geometry

No change in system, but IfcShapeRepresentation.RepresentationType is now more closely controlled by WHERE rules. Also the WHERE rules define a different system than the BLIS agreements for IFC R2.0

Swept Solid Geometry

Instead of using IfcAttDrivenExtrudedSegment 2X uses IfcExtrudedAreaSolid.

R2.0		2X	
<u>IfcAttDrivenExtrudedSegment</u>		<u>IfcExtrudedAreaSolid</u>	
1: * SweptArea	<i>IfcCurveBoundedPlane</i>	1: SweptArea	IfcProfileDef
2: * ExtrudedDirection	<i>IfcDirection</i>	2: Position	IfcAxis2Placement3D
3: Depth	IfcPositiveLengthMeasure	3: ExtrudedDirection	IfcDirection
4: Position	IfcAxis2Placement3D	4: Depth	IfcPositiveLengthMeasure
5: ProfileDef	IfcAttDrivenProfileDef		

All 'AttDriven' objects have been removed and their attributes moved up in the hierarchy; the functionality has remained the same.

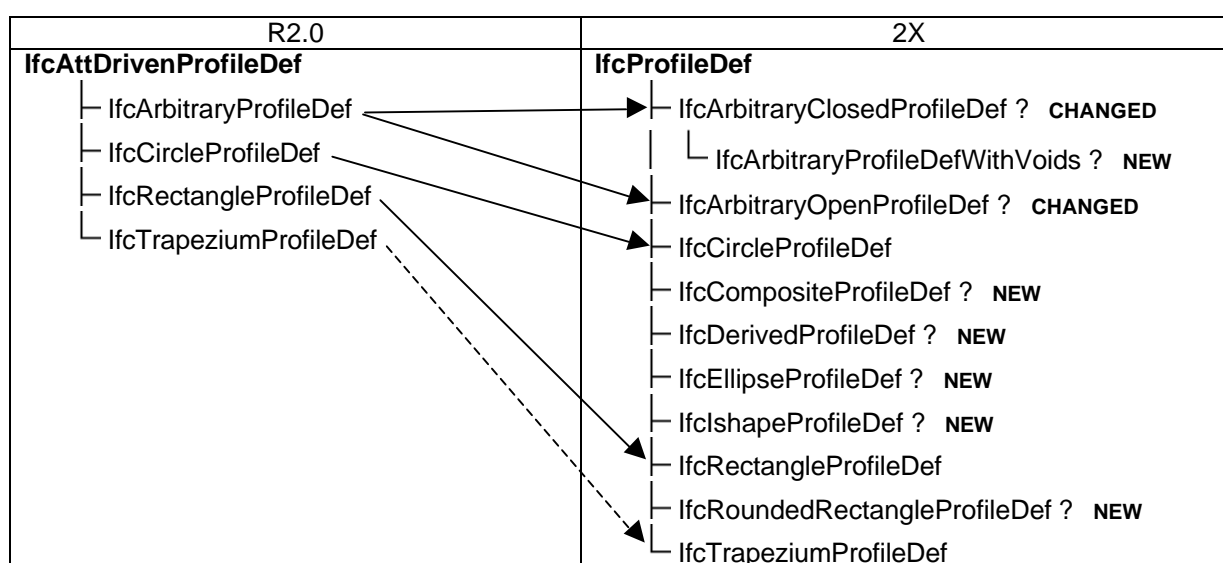
Clipped Geometry

Can be done now using `IfcPolygonalBoundedHalfSpace` in addition to the clipping plane (`IfcHalfSpaceSolid`)

RECHECK: Elaborate more, make diagram

Profile Definitions

All 'AttDriven' objects have been removed and their attributes moved up in the hierarchy; the functionality has remained the same.



NOTE: `IfcTrapeziumProfileDef` was not used in the IFC R2.0 implementations

IfcRectangleProfileDef

R2.0	2X
1: Position <code>IfcAxis2Placement2D</code>	1: ProfileType <code>IfcProfileTypeEnum</code>
2: ProfileType <code>IfcProfileTypeEnum</code>	2: OPT ProfileName <code>IfcLabel</code>
3: XDim <code>IfcPositiveLengthMeasure</code>	3: Position <code>IfcAxis2Placement2D</code>
4: YDim <code>IfcPositiveLengthMeasure</code>	4: XDim <code>IfcPositiveLengthMeasure</code>
	5: YDim <code>IfcPositiveLengthMeasure</code>

IfcCircleProfileDef

R2.0	2X
1: Position <code>IfcAxis2Placement2D</code>	1: ProfileType <code>IfcProfileTypeEnum</code>
2: ProfileType <code>IfcProfileTypeEnum</code>	2: OPT ProfileName <code>IfcLabel</code>
3: Radius <code>IfcPositiveLengthMeasure</code>	3: Position <code>IfcAxis2Placement2D</code>
	4: Radius <code>IfcPositiveLengthMeasure</code>

Arbitrary profiles

R2.0	2X
<u>IfcArbitraryProfileDef</u> 1: <i>Position</i> <i>IfcAxis2Placement2D</i> 2: ProfileType IfcProfileTypeEnum 3: CurveForSurface <i>IfcBoundedCurve</i>	<u>IfcArbitraryClosedProfileDef</u> 1: ProfileType IfcProfileTypeEnum 2: OPT ProfileName IfcLabel 3: OuterCurve <i>IfcCurve</i> <u>IfcArbitraryProfileDefWithVoids</u> 1: ProfileType IfcProfileTypeEnum 2: OPT ProfileName IfcLabel 3: OuterCurve <i>IfcCurve</i> 4: InnerCurves SET [1:?] OF IfcCurve <u>IfcArbitraryOpenProfileDef</u> 1: ProfileType IfcProfileTypeEnum 2: OPT ProfileName IfcLabel 3: Curve <i>IfcBoundedCurve</i>

Brep Geometry

No relevant changes.

R2.0	2x
IfcGeometricRepresentationItem └─ IfcSolidModel └─ IfcManifoldSolidBrep └─ IfcFacetedBrep	IfcRepresentationItem └─ IfcGeometricRepresentationItem └─ IfcSolidModel └─ IfcManifoldSolidBrep └─ IfcFacetedBrep

Shared Geometry (NEW)

Shared geometry is a new feature in IFC2X. In IFC R2.0 it was possible to share geometry, but there was no official way for it.

New object *IfcMappedItem* is used for sharing geometry. The mapped item uses subtypes of *IfcCartesianTransformationOperator [Abstract]* for defining the scale and rotation for the object instances sharing the representation map. See Appendix B for details.

Placement

IFC R2.0: *IfcProduct.LocalPlacement* ? *IfcLocalPlacement*IFC2X: *IfcProduct.ObjectPlacement* ? *IfcObjectPlacement*

R2.0	2X
IfcRoot └─ IfcModelingAid └─ IfcLocalPlacement └─ IfcConstrainedPlacement	IfcObjectPlacement └─ IfcGridPlacement └─ IfcLocalPlacement

IfcLocalPlacement

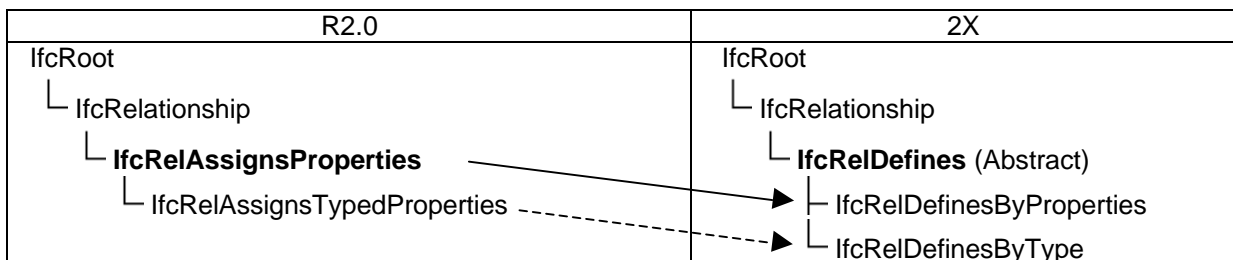
R2.0	2X
1: GlobalId IfcGloballyUniqueId 2: OwnerHistory IfcOwnerHistory 3: OPT Label STRING 4: OPT PlacementRelTo IfcObjectWithPlacementSelect 5: RelativePlacement IfcAxis2Placement	1: OPT PlacementRelTo IfcObjectPlacement 2: RelativePlacement IfcAxis2Placement

Basically no relevant changes when using IfcLocalPlacement. Any use of the "ConstrainedPlacement" has to be reworked with the "GridPlacement".

Property Set System

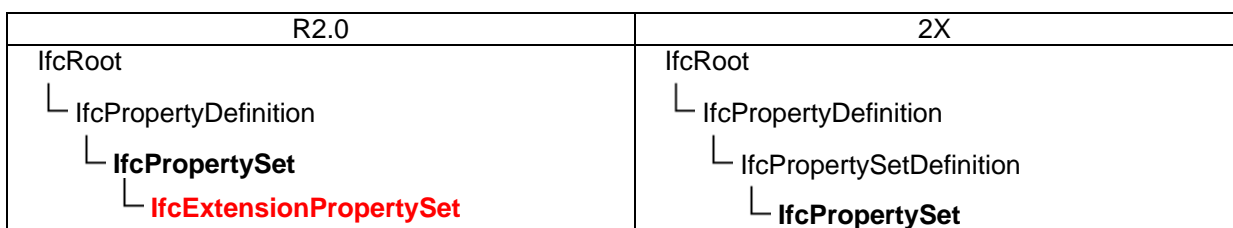
Property Set Assignment

Is still done through inverse IsDefinedBy attribute, but referred relationship objects has changed from IfcRelAssignsProperties to IfcRelDefines



Property Set Objects

No major changes. IfcExtensionPropertySet has been removed.

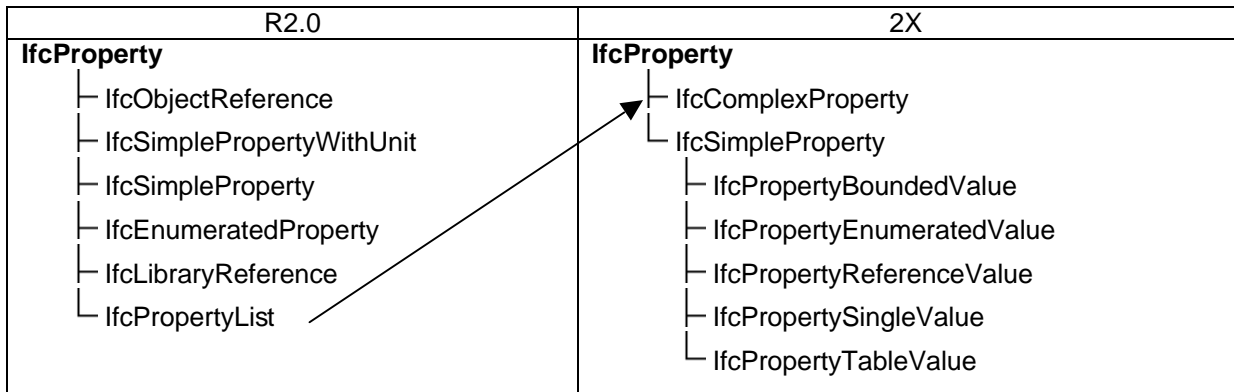


IfcPropertySet

	R2.0	2X
1:	GlobalId IfcGloballyUniqueId	GlobalId IfcGloballyUniqueId
2:	OwnerHistory IfcOwnerHistory	OwnerHistory IfcOwnerHistory
3:	OPT Label STRING	OPT Name IfcLabel
4:	Name STRING	OPT Description IfcText
5:	HasProperties LIST [1:?] OF IfcProperty	HasProperties SET [1:?] OF IfcProperty

The Name of the property set is optional in IFC2X, but there is a where rule on IfcPropertySet that says the name has to be provided.

IfcComplexProperty has replaced IfcPropertyList. Other properties have been moved under IfcSimpleProperty which is now an abstract supertype. Everything that was possible with IFC R2.0 is still possible in IFC 2X.



RECHECK: Which property types are really used by 2x implementers?

Shared Property Sets / Properties (NEW)

In IFC R2.0 all property sets were by instance, i.e. building elements didn't share property definitions. In IFC 2x it is possible to define 'types' that can be shared by several instances. There is no big difference, in 2x it is just possible to define properties by type, which is semantically more correct. Properties could also have been shared with the R2.0 model, but there has not been any agreement about this.

RECHECK: How is sharing of properties supported by 2x implementers?

Property Objects

R2.0	2X
<pre> IfcRoot ├── IfcPropertyDefinition │ ├── IfcOccupancyNumber │ ├── IfcSpaceUseCase │ ├── IfcManufactureInformation │ ├── IfcPropertySet │ ├── IfcMetricValue │ └── IfcElectricalCharacteristics ├── Pset_DoorLiningCommon ├── Pset_DoorPanelCommon ├── Pset_WindowLiningCommon ├── Pset_WindowPanelCommon ├── Calc -attributes (now done with IfcElementQuantity) ├── Various HVAC property sets ├── IfcPermeableCovering └── </pre> <p>NOTE: Objects with bold text were used in the BLIS views of IFC R2.0</p>	<pre> IfcRoot ├── IfcPropertyDefinition │ ├── IfcPropertySetDefinition │ │ ├── IfcDoorLiningProperties │ │ ├── IfcDoorPanelProperties │ │ ├── IfcElectricalBaseProperties │ │ │ └── IfcElectricalExtendedProperties │ │ ├── IfcElementQuantity │ │ ├── IfcFluidFlowProperties │ │ ├── IfcManufacturerInformation │ │ ├── IfcPermeableCoveringProperties │ │ ├── IfcPropertySet │ │ ├── IfcSpaceThermalLoad │ │ ├── IfcWindowLiningProperties │ │ └── IfcWindowPanelProperties │ └── IfcTypeObject │ ├── IfcTypeProduct │ │ ├── IfcDoorStyle │ │ └── IfcWindowStyle </pre>

Calc -Attributes (Removed)

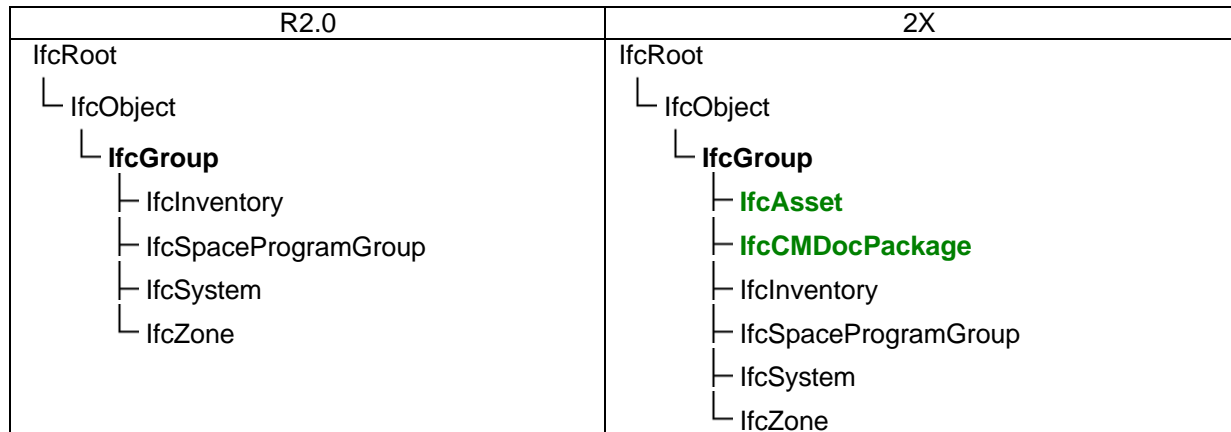
Calc attributes are replaced by IfcElementQuantity. The name of the quantity is no longer 'hard coded' but has to be agreed on. Some names are defined in the documentation for each building element, e.g. IfcWall has WallGroundArea, WallSideArea and WallVolume. Other names can be used, but they should be based on an implementers agreement. See Appendix C for details.

RECHECK: Are any other names agreed by 2x implementers?

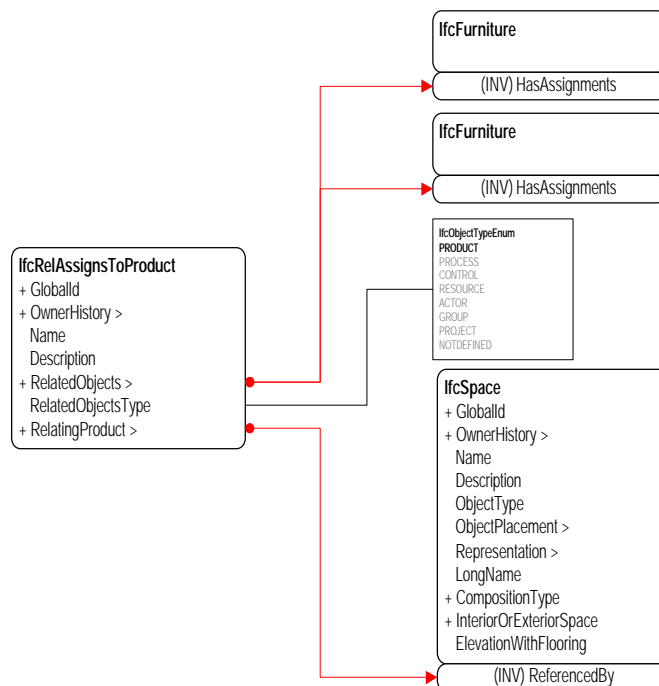
Grouping / Zones / Systems

R2.0: IfcObject.PartOfGroups ? IfcRelGroups.RelatingGroup ? IfcGroup

2X: IfcObject.HasAssignments ? IfcRelAssignsToGroup.RelatingGroup ? IfcGroup



In IFC2X it is possible to assign objects to a product without using a group in between. This is done using the IfcRelAssignsToProduct relationship



RECHECK: Is it a bug that a group has only an optional name?

Material Layers and Materials

Assigning material and material layers to a building element now happens though:

IfcObject.HasAssociations

? IfcRelAssociatesMaterial.RelatingMaterial

? IfcMaterialSelect.

The mechanism for defining material layers has remained the same.

IfcMaterial has changed a lot. Basically it only has a name and a possibility for classification. It is no longer possible to assign properties to a material. Now the mechanism is such that you assign a material to a material property and there is no way to navigate the relationship in the other direction (inverse)

Attribute / Relation	R2x	Change from R2.0 Final
Name	IfcLabel	New attribute in R2x
ClassifiedAs	INVERSE SET [0:1] OF IfcMaterialClassificationRelationship	New attribute in R2x
MaterialClassification	OPTIONAL IfcClassificationList	Deleted
Properties	SET [0:?] OF IfcMaterialPropertySelect	Deleted
MaterialFinishes	SET [0:?] OF IfcMaterialFinish	Deleted
MaterialName	STRING	Deleted

RECHECK: Is the missing inverse attribute from material to material property a bug?

Relationships

Containment

R2.0: IfcObject.Contains ? IfcRelContains.RelatedObjects/RelatingObject

2X: IfcObject.Decomposes ? IfcRelAggregates.RelatedObjects/RelatingObject

Wall to Wall Connection

No change is logical wall to wall connections connection

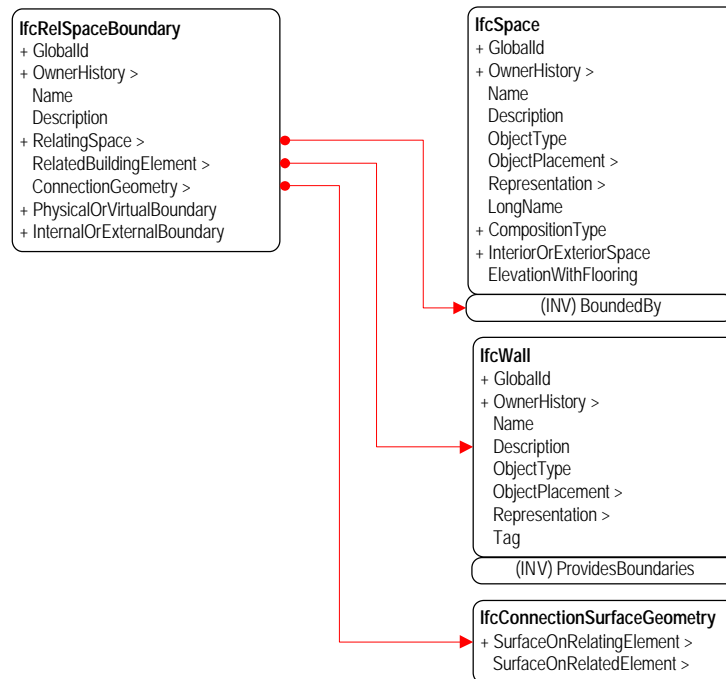
?? IfcConnectionEnum has changed to IfcConnectionTypeEnum

?? IfcConnectionEnum.START has changed to IfcConnectionTypeEnum

ATSTART

RECHECK: What are the rules if the connection has geometry?

Space Boundaries

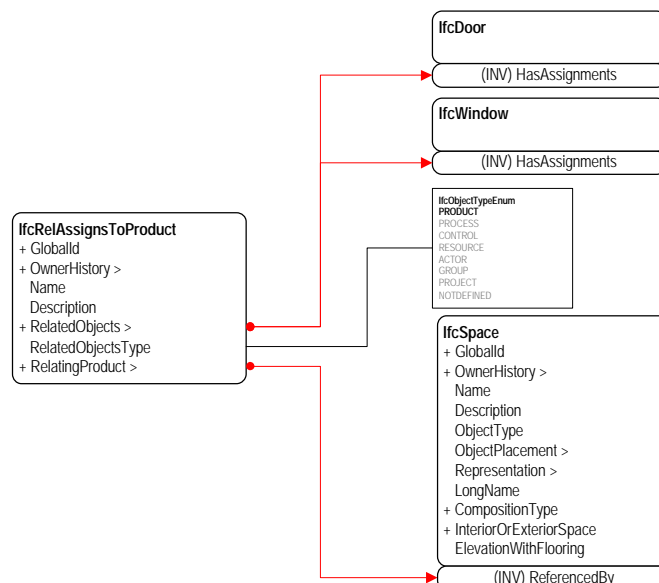


IfcSpaceBoundary has been replaced by IfcRelSpaceBoundary, which uses IfcConnectionSurfaceGeometry for the geometry.

RECHECK: WR1 does not apply, check how the physical/virtual system should be. Create similar diagram as was done for R2.0.

Space to Opening/Door/Window Connection

IfcRelContains.ContainedOrReferenced ? IfcContainedOrReferencedEnum has been removed. This was used in R2.0 for associating openings/doors/windows to spaces. In 2x some other way has to be used, e.g. IfcObject.HasAssignments -> IfcRelAssignsToProduct.RelatingProduct. The same system was described already in 'Groups / Systems / Zones' above.



RECHECK: Is this the way to do it in 2x? If yes, are there many IfcRelAssignsToProduct instances per space or just one

Fills Element

No relevant changes

Voids Element

No relevant changes

Building Elements

Wall

Geometry Use Definitions

R2.0	2X
?? Extruded or revolved with rectangular profiles	?? IfcWall can be used like in 2.0
?? Profile of the wall extruded horizontally	?? IfcWallStandardCase is new
?? The wall base line was derived from the geometry.	○ WallAxis: basically the definition of the baseline of the wall. Has to be always given.
?? Complex cases using Brep	○ Body: the solid definition either CSG or swept
	○ For swept (extruded): foot print of the wall extruded vertically

Properties

R2.0	2X
Pset_WallCommon - Reference - Description - ExtendToStructure - ExternalWall - FireRating - ThermalRating - AcousticRating IfcWall.calcWallArea IfcWall.calcWallVolume NOTE: Bold properties used in BLIS views	Pset_WallCommon - Reference - Description - ExtendToStructure - ExternalWall - FireRating - AcousticRating - LoadBearing - ThermalTransmittanceCoefficient IfcQuantityArea.Name = WallGroundArea IfcQuantityArea.Name = WallSideArea IfcQuantityVolume.Name = WallVolume

Relationships

Relationship	Use	System
Logical connections between walls	No changes	No changes
Space boundaries	No changes	Has changed
Relation to opening element	No changes	No changes
Relation to material	No changes	Has changed
Relation to building storey	No changes	Has changed
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

Door

Geometry Use Definitions

R2.0	2X
<p>B-Rep geometry was agreed to be used but the model allowed any kind of geometry through <code>IfcProductDefinitionShape</code> and additional parameters for the geometry were given in Property Sets.</p> <p>In IFC R2.0 files doors usually didn't have their own geometry, instead the geometry was taken from the opening element.</p>	<p>Still any kind of geometry can be used, but now the geometry of the door can also be defined by the object <code>IfcDoorStyle</code> related to the door through the relation <code>IfcRelDefinesByType</code> that provides a block definition for the geometry of the door.</p> <pre> IfcRoot ├── IfcPropertyDefinition │ └── IfcTypeObject │ └── IfcTypeProduct │ └── IfcDoorStyle </pre> <p>It is possible to define a shape aspect for the geometry that defines the door lining and the door panel.</p>

RECHECK: How is the geometry of the door defined by the IFC2X implementers?
Any agreements?

Properties

R2.0	2X
<p><code>IfcDoor.UserDefinedType</code></p> <p>Pset_DoorCommon</p> <ul style="list-style-type: none"> - Reference - Description - NominalHeight - NominalWidth - HardwareGroup - Shading - IsExterior - ParameterTakesPrecedence - ArbitraryShapeRepresentation - OrientationToExterior - Infiltration - ThermalTransmittanceCoefficient - FireRating - AcousticRating - SecurityRating <p>Pset_DoorLiningCommon</p> <ul style="list-style-type: none"> - LiningDepth - LiningThickness - ThresholdDepth - ThresholdThickness <p>Pset_DoorPanelCommon</p> <ul style="list-style-type: none"> - PanelThickness - PanelToLiningOffset - PanelHeight - PanelWidth - CrackLenght - InfiltrationCoefficient - StandardPanelType - PanelHasOpenings - GlazingAreaFraction - Glazing 	<p><code>IfcDoor.OverallHeight</code> <code>IfcDoor.OverallWidth</code></p> <p>Pset_DoorCommon</p> <ul style="list-style-type: none"> - Reference - Description - IsExterior - Infiltration - ThermalTransmittanceCoefficient - FireRating - AcousticRating - SecurityRating <p>Pset_DoorPanelCommon</p> <ul style="list-style-type: none"> - PanelToLiningOffset - CrackLength - InfiltrationCoefficient - PanelHasOpenings - GlazingAreaFraction - Finish - Color <p>IfcDoorLiningProperties.LiningDepth IfcDoorLiningProperties.LiningThickness <code>IfcDoorLiningProperties.ThresholdDepth</code> <code>IfcDoorLiningProperties.ThresholdThickness</code> <code>IfcDoorLiningProperties.TransomThickness</code> <code>IfcDoorLiningProperties.TransomOffset</code> <code>IfcDoorLiningProperties.LiningOffset</code> <code>IfcDoorLiningProperties.ThresholdOffset</code> <code>IfcDoorLiningProperties.CasingThickness</code> <code>IfcDoorLiningProperties.CasingDepth</code></p> <p><code>IfcDoorPanelProperties.PanelDepth</code> <code>IfcDoorPanelProperties.PanelOperation</code> <code>IfcDoorPanelProperties.PanelWidth</code></p>

<ul style="list-style-type: none"> - Finish - Color <p>Pset_DoorPanelSwinging</p> <ul style="list-style-type: none"> - CommonDoorPanelProperties - LeftNotRightSwing - SwingStartAngle - IncludedSwingAngle <p>Pset_HardwareGroup</p> <ul style="list-style-type: none"> - Reference - Description - Manufacturer - ModelLabel - ModelDescription - Finish <p>NOTE: Bold properties used in BLIS views</p>	<p>IfcDoorPanelProperties.PanelPosition</p> <p>IfcDoorStyle.OperationType</p> <p>IfcDoorStyle.ConstructionType</p> <p>IfcDoorStyle.ParameterTakesPrecedence</p> <p>IfcDoorStyle.Sizeable</p> <p>Pset_HardwareGroup</p> <ul style="list-style-type: none"> - Reference - Description - Manufacturer - ModelLabel - ModelDescription - Finish
--	---

Relationships

Relationship	Use	System
Relation to building storey	No changes	Has changed
Relation to space	No changes	Has changed
Relation to opening element	No changes	No changes
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

Window

Geometry

R2.0	2X
<p>B-Rep geometry was agreed to be used but the model allowed any kind of geometry through IfcProductDefinitionShape and additional parameters for the geometry were given in Property Sets.</p> <p>In IFC R2.0 files windows usually didn't have their own geometry, instead the geometry was taken from the opening element.</p>	<p>Still any kind of geometry can be used, but now the geometry of the window can also be defined by the object IfcWindowStyle related to the window through the relation IfcRelDefinesByType that provides a block definition for the geometry of the window.</p> <pre> IfcRoot ├── IfcPropertyDefinition │ └── IfcTypeObject │ └── IfcTypeProduct │ └── IfcWindowStyle </pre> <p>It is possible to define a shape aspect for the geometry that defines the window lining and the window panel.</p>

RECHECK: How is the geometry of the window defined by the IFC2X implementers? Any agreements?

Properties

R2.0	2X
<p>IfcWindow.UserDefinedType</p> <p>Pset_WindowCommon</p> <ul style="list-style-type: none"> - Reference - Description 	<p>IfcWindow.OverallHeight</p> <p>IfcWindow.OverallWidth</p> <p>Pset_WindowCommon</p> <ul style="list-style-type: none"> - Reference

<ul style="list-style-type: none"> - NominalHeight - NominalWidth - HardwareGroup - Shading - IsExterior - ParameterTakesPrecedence - ArbitraryShapeRepresentation - OrientationToExterior - Infiltration - ThermalTransmittanceCoefficient - FireRating - AcousticRating - SecurityRating <p>Pset_WindowLiningCommon</p> <ul style="list-style-type: none"> - calcLiningDepth - calcLiningThickness <p>Pset_WindowPanelCommon</p> <ul style="list-style-type: none"> - FrameWidth - FrameDepth - FrameToLiningOffset - PanelHeight - PanelWidth - StileDepth - StileThickness - CrackLenght - InfiltrationCoefficient - GlazingAreaFraction - StandardPanelType - Glazing <p>Pset_WindowPanelPivoting</p> <ul style="list-style-type: none"> - PivotsVertically - PivotsHorizontally <p>Pset_WindowPanelSliding</p> <ul style="list-style-type: none"> - PanelFixed - HorizontalNotVerticalSliding - CounterBalanced <p>Pset_WindowPanelSwinging</p> <ul style="list-style-type: none"> - LeftNotRightHinged - PanelSwingAngle <p>Pset_HardwareGroup</p> <ul style="list-style-type: none"> - Reference - Description - Manufacturer - ModelLabel - ModelDescription - Finish <p>NOTE: Bold properties used in BLIS views</p>	<ul style="list-style-type: none"> - Description - IsExterior - Infiltration - ThermalTransmittanceCoefficient - FireRating - AcousticRating - SecurityRating <p>Pset_WindowPanelCommon</p> <ul style="list-style-type: none"> - FrameToLiningOffset - StileDepth - StileThickness - CrackLength - InfiltrationCoefficient - GlazingAreaFraction <p>Pset_WindowCleaning</p> <ul style="list-style-type: none"> - WindowCleaningElementTypeEnum <p>IfcWindowLiningProperties.LiningDepth</p> <p>IfcWindowLiningProperties.LiningThickness</p> <p>IfcWindowLiningProperties.TransomThickness</p> <p>IfcWindowLiningProperties.MullionThickness</p> <p>IfcWindowLiningProperties.FirstTransomOffset</p> <p>IfcWindowLiningProperties.SecondTransomOffset</p> <p>IfcWindowLiningProperties.FirstMullionOffset</p> <p>IfcWindowLiningProperties.SecondMullionOffset</p> <p>IfcWindowPanelProperties.OperationType</p> <p>IfcWindowPanelProperties.PanelPosition</p> <p>IfcWindowPanelProperties.FrameDepth</p> <p>IfcWindowPanelProperties.FrameThickness</p> <p>IfcWindowStyle.ConstructionType</p> <p>IfcWindowStyle.OperationType</p> <p>IfcWindowStyle.ParameterTakesPrecedence</p> <p>IfcWindowStyle.Sizeable</p> <p>Pset_HardwareGroup</p> <ul style="list-style-type: none"> - Reference - Description - Manufacturer - ModelLabel - ModelDescription - Finish
--	--

Relationships

Relationship	Use	System
Relation to building storey	No changes	Has changed
Relation to space	No changes	Has changed
Relation to opening element	No changes	No changes
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

Opening Element

Geometry Use Definitions

Uses the same geometry types as in R2.0, but with a few additional agreements that are different than what was used in the IFC R2.0 implementations.

R2.0	2X
<p>?? IfcRectangleProfileDef.XDim interpreted as opening width</p> <p>?? IfcRectangleProfileDef.YDim interpreted as opening height</p> <p>?? Extrusion depth interpreted as thickness</p>	<p>The following interpretation of dimension parameter applies for rectangular openings:</p> <p>?? IfcRectangleProfileDef.YDim interpreted as opening width</p> <p>?? IfcRectangleProfileDef.XDim interpreted as opening height</p> <p>NOTE: Rectangles are now defined centric, the placement location has to be set:</p> <p>?? IfcCartesianPoint(XDim/2,YDim/2)</p>
<p>In the IFC R2.0 implementations voids in curved walls were made using extruded openings that were large enough to 'punch' the opening into the curved wall.</p>	<p>Special agreement for defining openings in round building elements, e.g., in round walls. The opening width, in case of a rectangular opening equal with the IfcRectangleProfileDef.XDim, is defined as the straight line distance between two parallel jambs. If the jambs are defined radial (to the center of the arc used to define the round wall) then the opening width is defined to be the outer arc length.</p> <p>Pset_OpeningCommon.ParallelJambs is used for indicating the case.</p>

Properties

R2.0	2X
IfcOpeningElement.calcOpeningArea	IfcQuantityArea.Name = OpeningArea
NOTE: Bold properties used in BLIS views	

Relationships

Relationship	Use	System
Relation to building storey	Has changed ⁽¹⁾	Has changed
Relation to space	No changes	Has changed
Relation to voided element	No changes	No changes
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

⁽¹⁾ In the IFC R2.0 implementations a opening element wasn't allowed to be contained by a building storey. In IFC2X it has to be contained by a spatial structure, e.g. a building storey.

Column

Geometry Use Definitions

No relevant changes. In IFC2X the new IfcIshapeProfileDef can be used for column profiles.

Properties

R2.0	2X
IfcColumn.calcColumnSectionArea IfcColumn.calcColumnVolume Pset_ColumnCommon - Reference - Description (Missing in IFC2X) - Length - Width - Height - SlendernessRatio NOTE: Bold properties used in BLIS views	IfcQuantityArea.Name = CrossSectionArea IfcQuantityVolume.Name = ColumnVolume Pset_ColumnCommon - Reference - Length - Width - Height - SlendernessRatio

Relationships

Relationship	Use	System
Relation to building storey	No changes	Has changed
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

Beam

Geometry Use Definitions

No relevant changes. In IFC2X the new IfcIshapeProfileDef can be used for beam profiles.

Properties

R2.0	2X
IfcBeam.calcBeamSectionArea IfcBeam.calcBeamVolume Pset_BeamCommon - Reference - Description (Missing in IFC2X) - Depth - Width - Span - Camber - SectionModulus - Slope NOTE: Bold properties used in BLIS views	IfcQuantityArea.Name = CrossSectionArea IfcQuantityVolume.Name = BeamVolume Pset_BeamCommon - Reference - Depth - Width - Span - Camber - SectionModulus - Slope

Relationships

Relationship	Use	System
Relation to roof	Has Changed ⁽¹⁾	New
Relation to building storey	No changes	Has changed
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

⁽¹⁾ Beams can be used as part of a roof that consists of slabs and beams

Slab

Geometry Use Definitions

No relevant changes in geometry

Properties

R2.0	2X
IfcSlab.PredefinedType IfcSlab.calcSlabArea IfcSlab.calcSlabVolume Pset_SlabCommon - Reference - Description - FireRating - ThermalRating - AcousticRating Pset_SlabRoof - RequiredSlope NOTE: Bold properties used in BLIS views	IfcSlab.PredefinedType IfcQuantityArea.Name = SlabArea IfcQuantityVolume.Name =SlabVolume Pset_SlabCommon - Reference - Description - FireRating - ThermalTransmittanceCoefficient - AcousticRating Pset_SlabRoof - RequiredSlope

Relationships

Relationship	Use	System
Space boundaries	No changes	Has changed
Relation to opening element	No changes	No changes
Relation to material	No changes	Has changed
Relation to building storey	No changes	Has changed
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

Roof

IfcRoof is not part of the IFC R2.0 view definitions, only IfcSlab of type roof.

Geometry Use Definitions

Just like in IFC R2.0 the roof can be an aggregate of other components or it can have a geometry of its own. In IFC R2.0 the types of objects that can form a roof was not defined, but IfcSlab or type Roof was a logical assumption. In IFC2X the roof consists of slabs and beams. If the roof is not an aggregate then it uses Brep geometry, just like in IFC R2.0

Properties

R2.0	2X
IfcRoof.calcTotalRoofSurfaceArea Pset_RoofCommon - Reference - Description - FireRating	IfcQuantityArea.Name = TotalRoofArea Pset_RoofCommon - Reference - Description - FireRating

Relationships

Relationship	Use	System
Relation to components	Has Changed ⁽¹⁾	New
Relation to building storey	No changes	Has changed
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

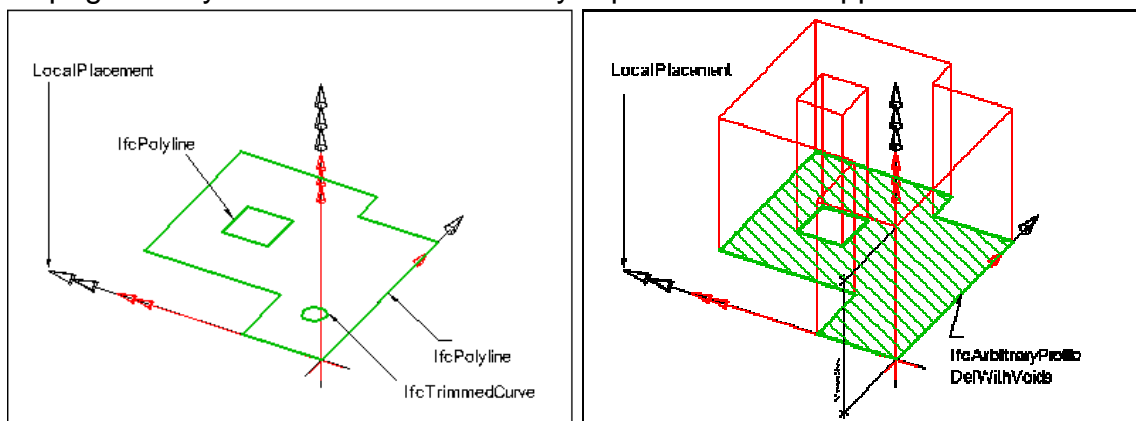
⁽¹⁾ Roofs can consist of slabs and beams

Space

Geometry Use Definitions

Nothing has basically changed, but because of the changes in the profile system it is now possible to define spaces with holes using the `IfcArbitraryProfileDefWithVoids` object in the profile. If voids are allowed in spaces or not is still under debate by the 2X implementers.

Spaces in IFC2X have a separate 2D geometry for defining the foot print, The swept geometry is still extruded vertically. Spaces can be clipped like before.



Properties

R2.0	2X
<code>IfcSpace.SpaceName</code> (Missing in IFC2X)	<code>IfcQuantityLength.Name = AverageHeight</code>
<code>IfcSpace.SpaceReference</code> (Missing in IFC2X)	<code>IfcQuantityLength.Name = AverageGrossHeight</code>
<code>IfcSpace.calcTotalPerimeter</code>	<code>IfcQuantityLength.Name = AverageClearHeight</code>
<code>IfcSpace.calcTotalArea</code>	<code>IfcQuantityLength.Name = Perimeter</code>
<code>IfcSpace.calcTotalVolume</code>	<code>IfcQuantityArea.Name = Area</code>
<code>IfcSpace.calcAverageHeight</code>	<code>IfcQuantityVolume.Name = Volume</code>
<code>IfcSpace.calcAverageGrossHeight</code>	
<code>IfcSpace.calcAverageClearHeight</code>	
<code>IfcSpace.calcElevationWithFlooring</code>	
Pset_SpaceCommon	Pset_SpaceCommon
- Description	- Description
- CodeUseType	- CodeUseType
- SpaceCatalogue	- SpaceCatalogue
- ReqSommerSpaceTemperature	- ReqSummerSpaceTemperature
- ReqSummerSpaceHumidity	- ReqSummerSpaceHumidity
- ReqWinterSpaceTemperature	- ReqWinterSpaceTemperature
- ReqWinterSpaceHumidity	- ReqWinterSpaceHumidity
- ReqIntermediateSpaceTemperature	- ReqIntermediateSpaceTemperature
- ReqIntermediateSpaceHumidity	- ReqIntermediateSpaceHumidity
- ReqDiscontinuedHeating	- ReqDiscontinuedHeating
- MainFireUse	- MainFireUse
- AncillaryFireUse	- AncillaryFireUse
- FireRiskFactor	- FireRiskFactor
- NaturalVentilation	- NaturalVentilation
- SprinklerProtection	- SprinklerProtection
Pset_SpaceElementInformation	Pset_SpaceHvacInformation
- CoolingDesignAirflow	- CoolingDesignAirflow
- HeatingDesignAirflow	- HeatingDesignAirflow
- TotalSensibleHeatGain	- TotalSensibleHeatGain
- TotalHeatGain	- TotalHeatGain
- TotalHeatLoss	- TotalHeatLoss
- HeatingDryBulb	- HeatingDryBulb
- HeatingRelativeHumidity	- HeatingRelativeHumidity
- CoolingDryBulb	- CoolingDryBulb
- CoolingRelativeHumidity	- CoolingRelativeHumidity
- VentilationAirFlowrate	- VentilationAirFlowrate
	- ExhaustAirFlowrate
	- CeilingRAPlenum
	- BoundaryAreaHeatLoss

<ul style="list-style-type: none"> - ExhaustAirFlowrate - CeilingRAPlenum - BoundaryAreaHeatLoss <p>Pset_SpaceProgramCommon</p> <ul style="list-style-type: none"> - ProgramSpaceDescription - RoomNumber - RoomName - ProgrammedFloorArea (Missing in IFC2X) - RequestedLocations - GeneralLocationDescription - Function - SecurityRequirements - SpecialRequirements - BudgetLimits - InteractWith <p>Pset_SpaceProgramCirculation</p> <ul style="list-style-type: none"> - CommonSpaceProgramProperties - SpacesServed - CirculationLoad - RequiredFFETypes <p>Pset_SpaceProgramOccupied</p> <ul style="list-style-type: none"> - OccupiedSpaceProgramStandard - Occupants - OccupantOrganizataion - OccupancyTargetDate - BldgCodeOccupancyType - ProgrammedOccupantCount - RequiredFFETypes - PrivacyRequirements - WeeklyOccupiedHours <p>Pset_SpaceProgramOccupiedStandard</p> <ul style="list-style-type: none"> - EmployeeType - FurnitureStyle - CostLimit - StandardArea - MinimumArea - MaximumArea - StandardLength - StandardWidth 	<p>Pset_SpaceProgramCommon</p> <ul style="list-style-type: none"> - ProgramSpaceDescription - GeneralLocationDescription - Function - SecurityRequirements - SpecialRequirements <p>Pset_SpaceProgramCirculation</p> <ul style="list-style-type: none"> - CirculationLoad - RequiredFFETypes <p>Pset_SpaceProgramOccupied</p> <ul style="list-style-type: none"> - Occupants - OccupantOrganizataion - OccupancyTargetDate - BldgCodeOccupancyType - ProgrammedOccupantCount - RequiredFFETypes - PrivacyRequirements - WeeklyOccupiedHours <p>Pset_SpaceProgramOccupiedStandard</p> <ul style="list-style-type: none"> - EmployeeType - FurnitureStyle - StandardLength - StandardWidth
--	--

NOTE: Bold properties used in BLIS views

The attributes SpaceName and SpaceReference (number) don't exist on IfcSpace anymore. The use of these properties is dependent on implementers agreements.

RECHECK: What is IFC2X implementers agreement about the space name and space number?

Relationships

Relationship	Use	System
Space boundaries	No changes	Has changed
Relation to building storey	No changes	Has changed
Relation to contained elements	No changes	Has changed
Relation to referenced elements	No changes	Has changed
Grouping	No changes	Has changed
Property assignment	No changes	Has changed

Conclusions

Everything that was included in the BLIS views of IFC R2.0 is possible also with IFC2X. Some minor things are missing, e.g. the space number, but this can be fixed with implementers agreements

In general the model has become more versatile and flexible, which brings up the need for more precise implementers agreements than was the case with IFC R2.0. The IAI has provided better textual descriptions than before, e.g. in the area of geometry use definitions which should help.

Provided that the flexibility of the model is constrained adequately with agreements implementing the scope of the BLIS views in IFC2X should technically not be a major task. If the model is not constrained the implementation will be more difficult than it was with IFC R2.0.

NOTE: This document doesn't yet cover the following objects used in the BLIS views

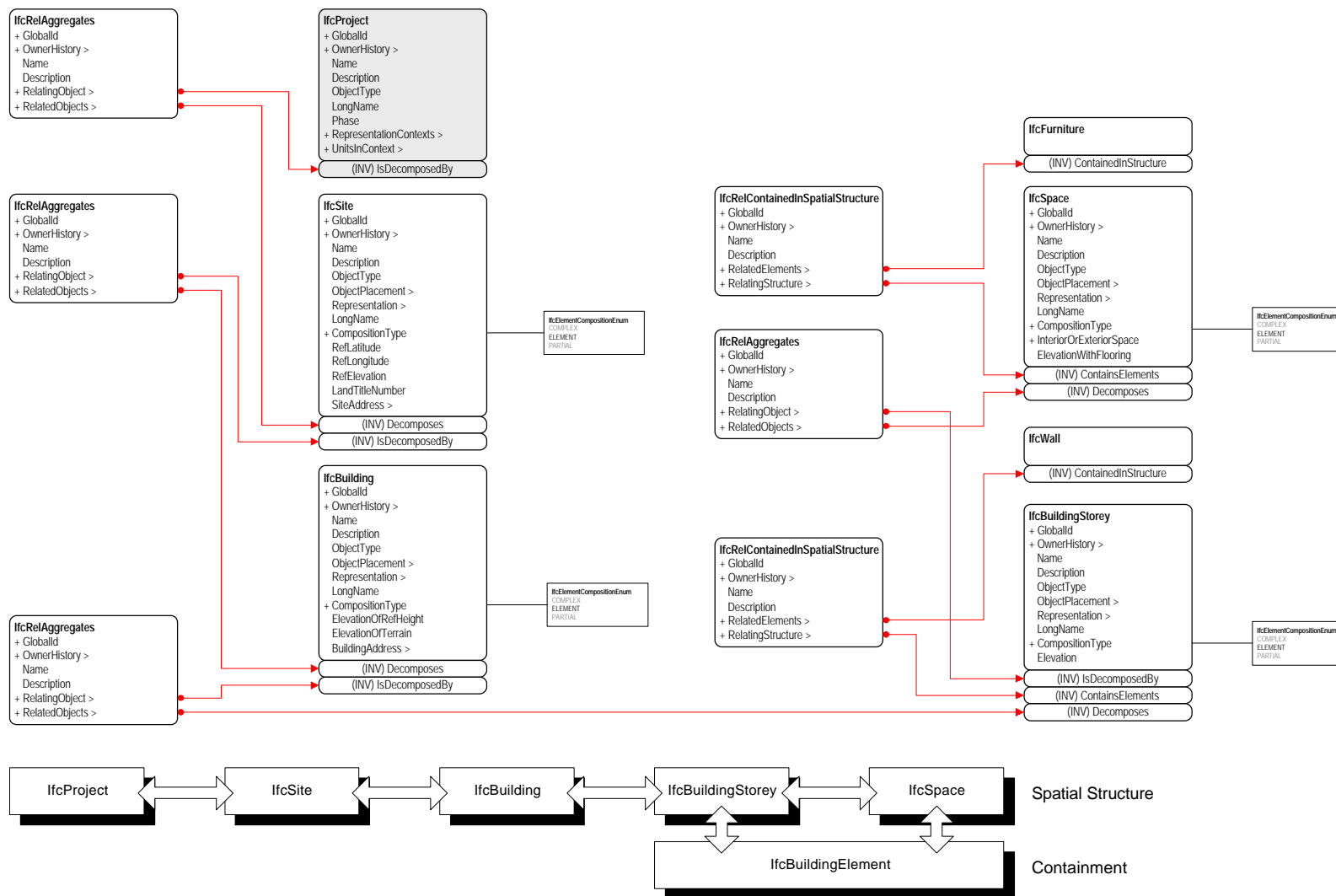
- Cabinet
- Counter or shelf
- Electrical appliance
- Electrical fixture
- Flow controller
- Flow equipment
- Flow fitting
- Flow segment
- Flow terminal
- Furniture
- Plumbing fixture
- Proxy [product with geometry]
- Stair

Also the following concepts are not covered

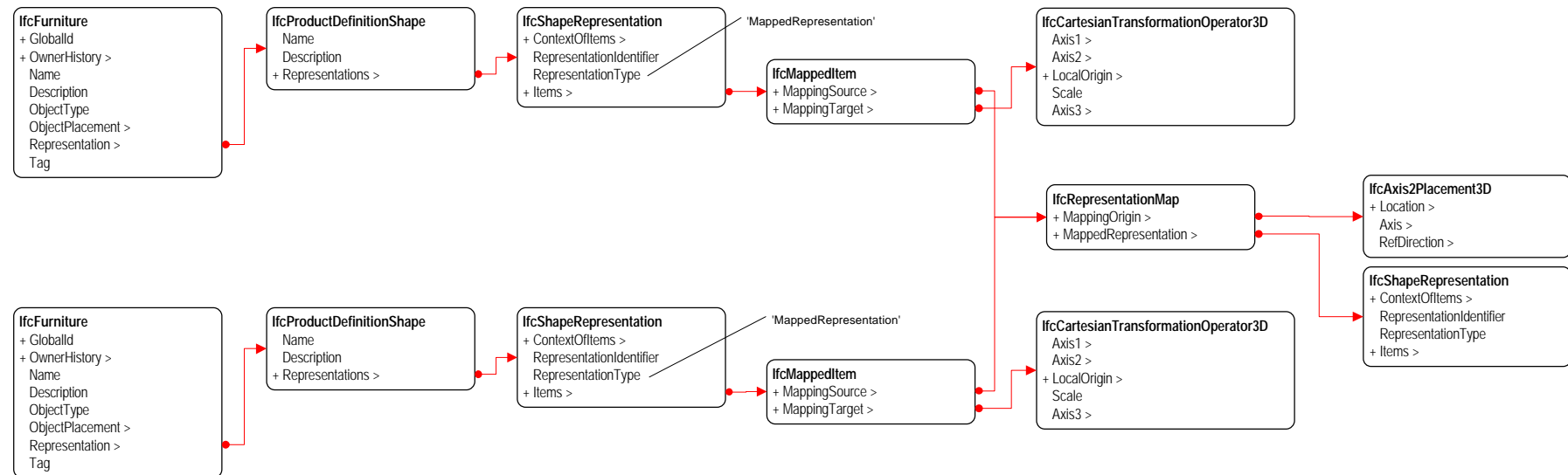
- Manufacture information
- Classification
- Design grid
- Cost information

APPENDICES: Implementation diagrams

Appendix A: IFC2X Project Structure



Appendix B: Shared Geometry in IFC2X



Appendix C: Quantity Objects in IFC2X

