|  |  |
| --- | --- |
| **Information Delivery Specifications**  **IDS** | |
| **IDS Bingo Berlin**  **This IDS contains specifications for every facet. Every facet has specifications that are simple to verify basic checkers as well as specifications containing regular expressions.**  **The results can be checked on the Bingo scorecard.** | |
| **Version:** | 1 |
| **Date:** | 24/09/2025 |
| **Author Email :** | menno@aronsengelauff.nl |
| **Milestone:** | Technical Review |
| **Purpose:** | IDS written for 'ILSOenE\_PoC-model.ifc' to test IDS tools for creating and validating specifications. |
|  |  |
| Copyright: Arons en Gelauff architecten / BIMvoice | |

Created with usBIM.IDSeditor by ACCA software S.p.A.

# **Specifications**

## **ATTRIBUTE: IfcProjectedCRS must be a valid EPSG code - ATT01**

Checks if the model is placed onto the correct coordinate system <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcProjectedCRS.htm>

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

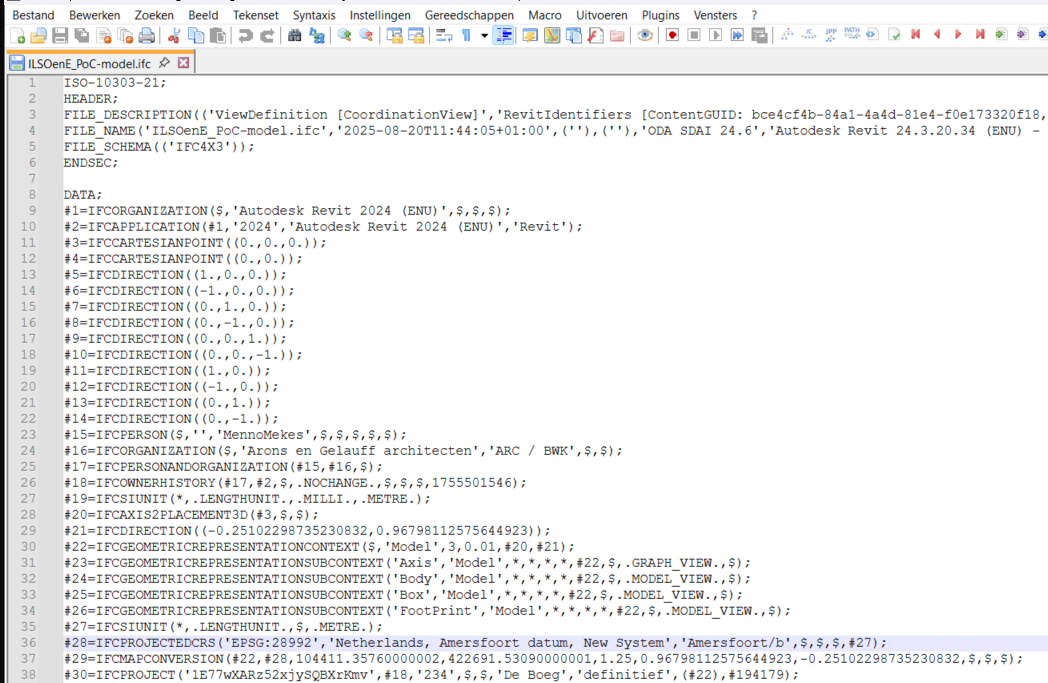
- IFC class **IFCPROJECTEDCRS**

**that MEET the following requirements**

- MUST HAVE attribute **Name** **= EPSG:28992** (Instructions: Choose the EPSG-code for your model here: https://epsg.io/28992Change it to something else to see if the specification works.)

**Instructions**

If your tool does not support this entity, skip this specification.



**Result**

The specification should pass. Try to change the EPSG to something else to find out if you get a mistake.

## **ATTRIBUTE: Attributes in IfcMapConversion must be present - ATT02**

Checks if the model is placed on coordinates <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcMapConversion.htm>

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

- IFC class **IFCMAPCONVERSION**

**that MEET the following requirements**

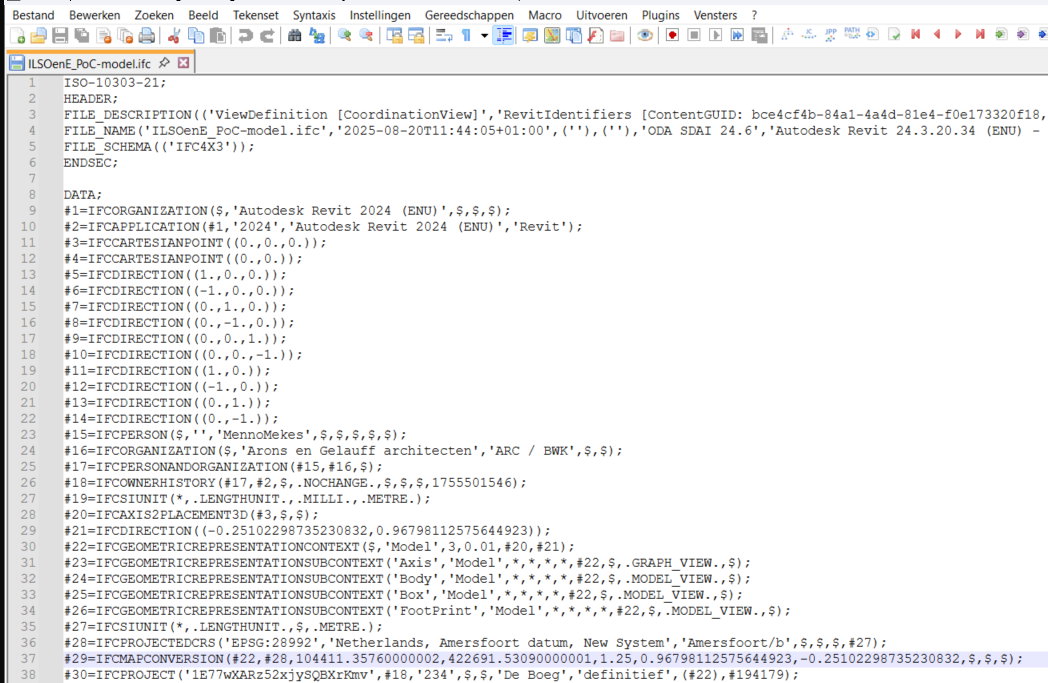
- MUST HAVE attribute **Eastings** (Instructions: Attribute must exist)

- MUST HAVE attribute **Northings** **= 422691.5309** (Instructions: Attribute must exist)

- MUST HAVE attribute **OrthogonalHeight** (Instructions: Attribute must exist)

**Instructions**

If your tool does not support this entity, skip this specification.



**Result**

The specification should pass. When it fails the Northings you can check the notation. Did you enter meters, a point or a comma. Your tool either does not support this or do you need to change something in the notation.

## **ATTRIBUTE: Attributes in IfcMapConversion cannot be 0 - ATT03**

Checks if the model is placed on coordinates <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcMapConversion.htm>

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

- IFC class **IFCMAPCONVERSION**

**that MEET the following requirements**

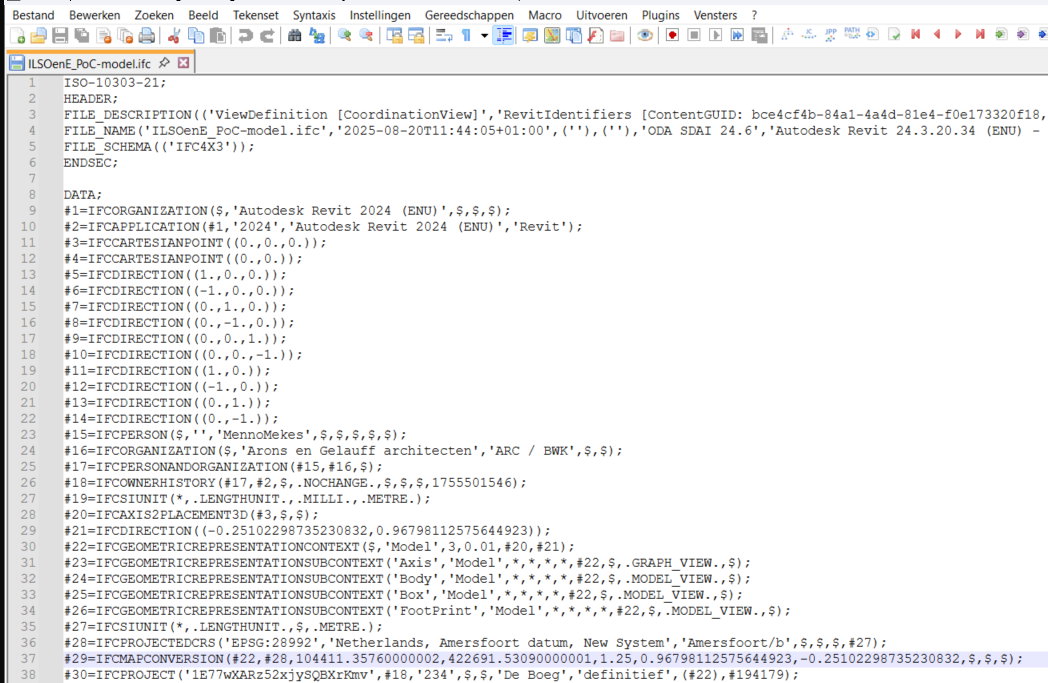
- MUST HAVE attribute **Eastings** **< 0** (Instructions: Attribute must be greater than 0(the coordinates must be set but cannot be at 0,0,0 for the model to be at the correct location)(This test is less than 0 in order to generate a mistake))

- MUST HAVE attribute **Northings** **> 0** (Instructions: Attribute must be greater than 0(the coordinates must be set but cannot be at 0,0,0 for the model to be at the correct location))

- MUST HAVE attribute **OrthogonalHeight** **≥ 1.25** (Instructions: Attribute must be greater than or equal to 1,25m(Actually it should just be 1.25m but this is to test the different bounds.))

**Instructions**

When bounds do not work, skip this specification.



**result**

the specification should pass.

## **ATTRIBUTE: list of correct naming of building storeys - ATT04**

validates the naming of building storeys in a list <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcBuildingStorey.htm>

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

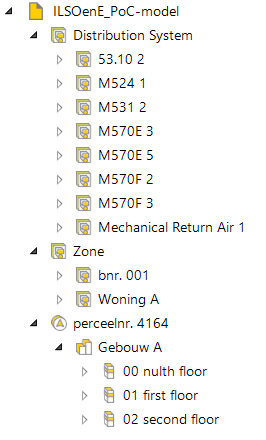
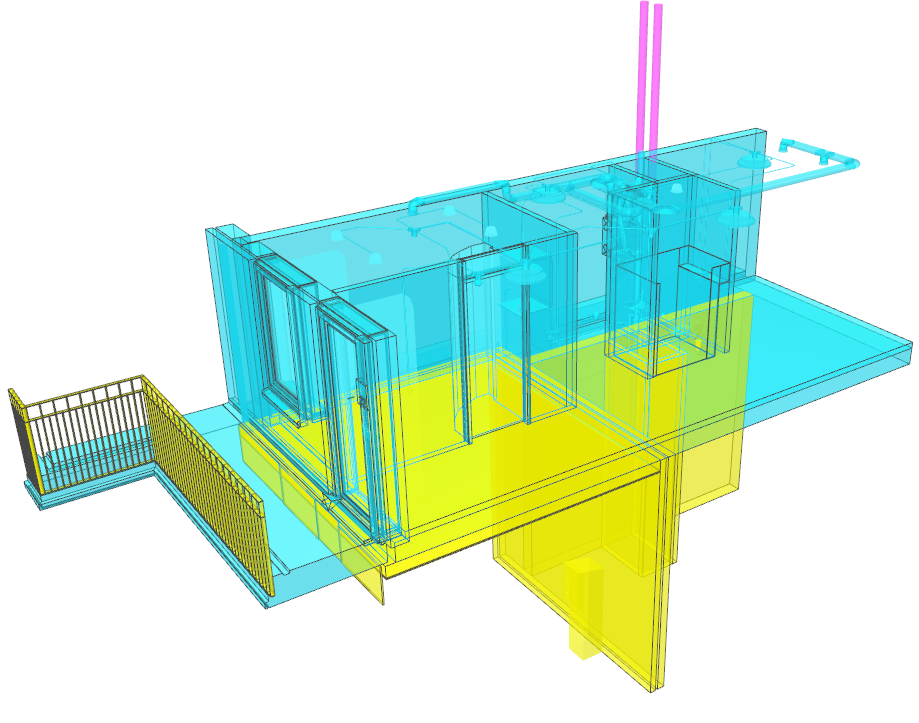
- IFC class **IFCBUILDINGSTOREY**

**that MEET the following requirements**

- MUST HAVE attribute **Name** with value **= 00 ground floor (or: = 01 first floor; = 02 second floor)**

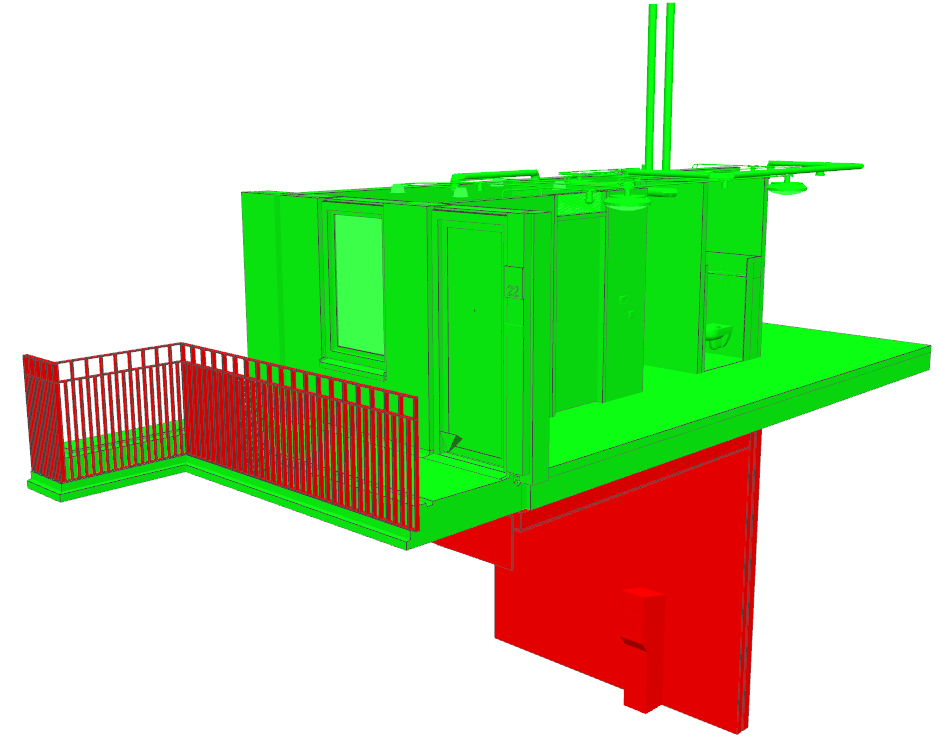
**Instructions**

Add a list of names in the Requirements



**Results**

The model contains 3 building storeys: 00 nulth floor, 01 first floor and 02 second floor. 00 nulth floor is obviously wrong. The specification should give you this IfcBuildingStorey as a result.



## **ATTRIBUTE: simple RegEx list of correct naming of building storeys - ATT05**

validates the naming of building storeys in with a RegEx pattern <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcBuildingStorey.htm>

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

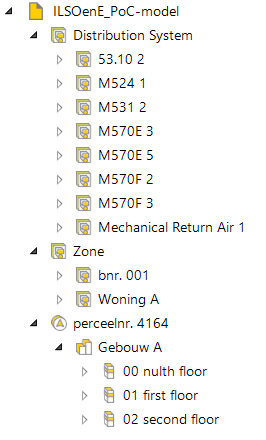
**The model MUST contain entities that have**

- IFC class **IFCBUILDINGSTOREY**

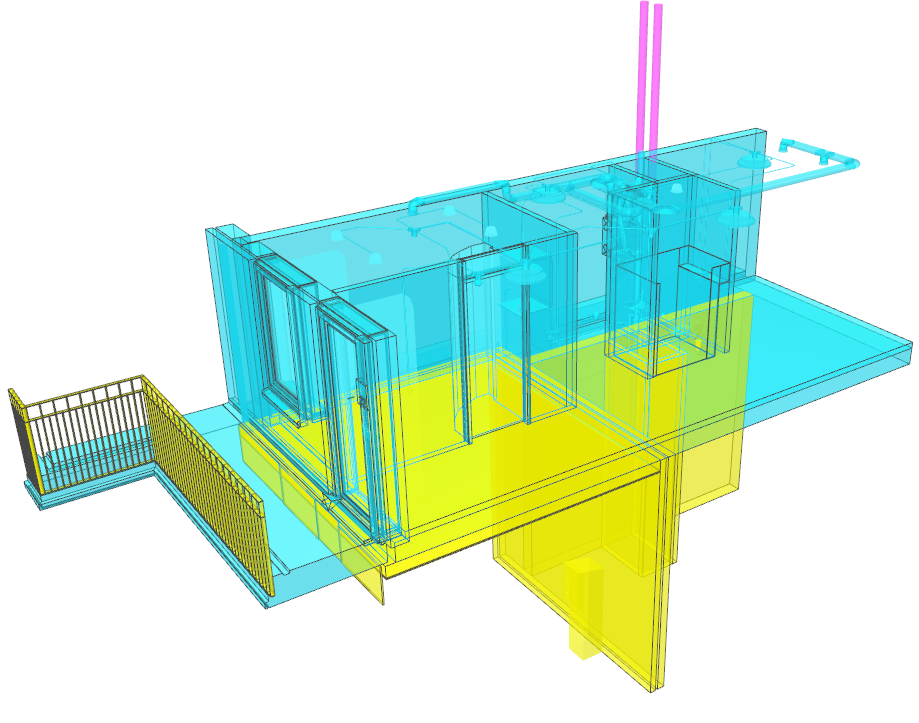
**that MEET the following requirements**

- MUST HAVE attribute **Name** **matching the pattern 00 ground floor|01 first floor|02 second floor**

**Instructions**

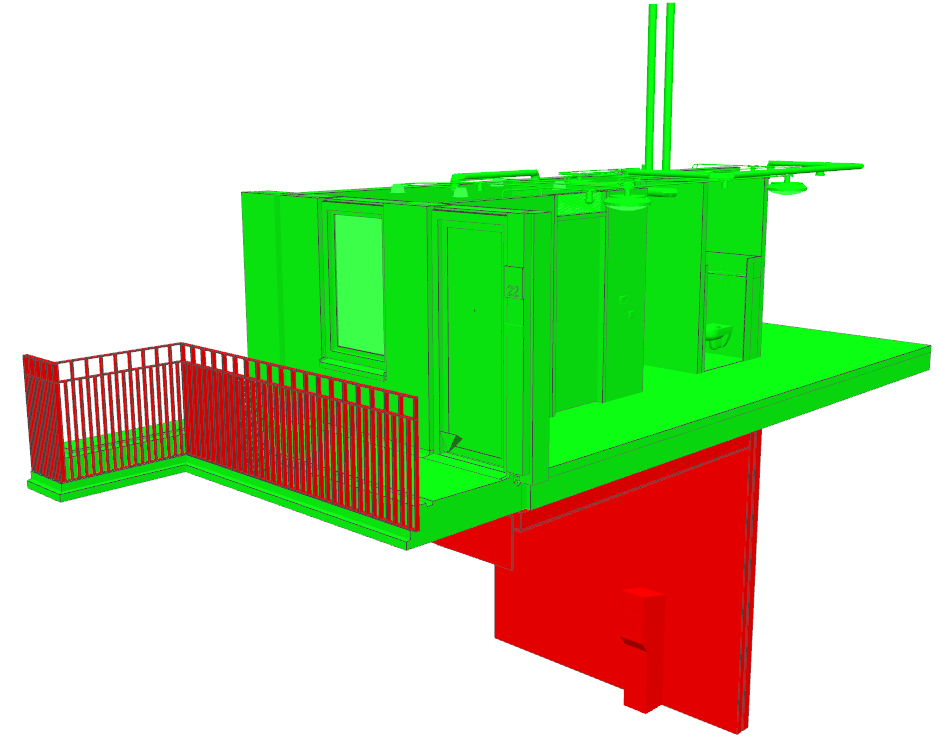


Add a list of names in the Requirements by use of regular expression



**Results**

The model contains 3 building storeys: 00 nulth floor, 01 first floor and 02 second floor. 00 nulth floor is obviously wrong. The specification should give you this IfcBuildingStorey as a result.



## **ATTRIBUTE: complex RegEx of correct naming of building storeys - ATT06**

validates the naming of building storeys in with a RegEx pattern <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcBuildingStorey.htm>

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

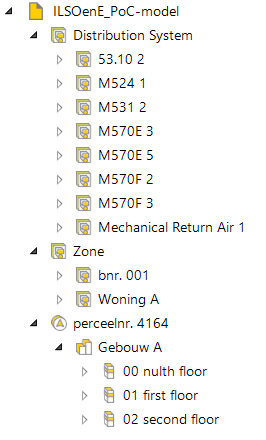
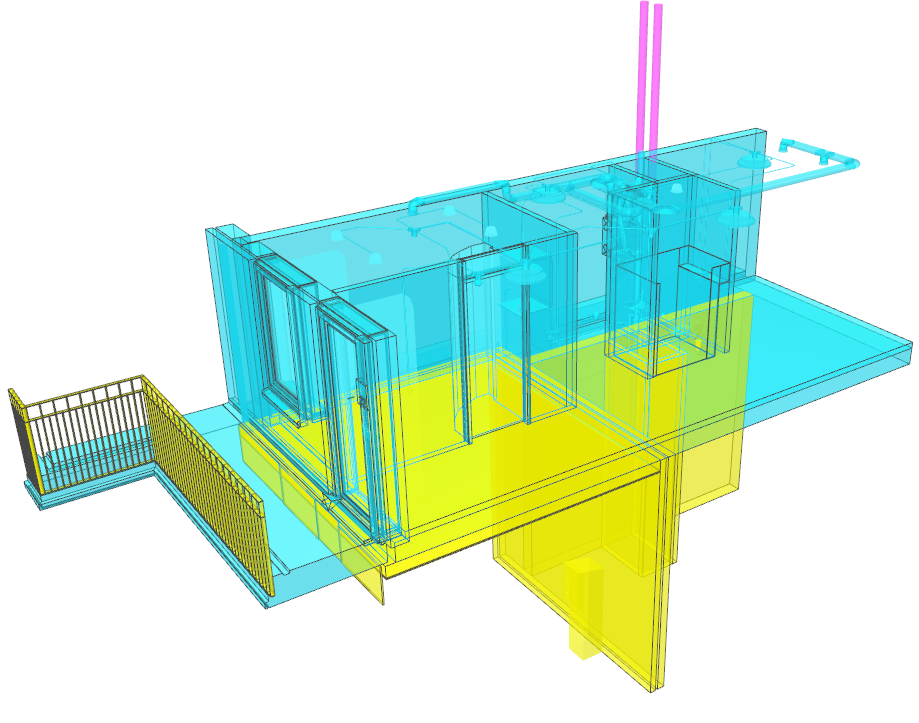
- IFC class **IFCBUILDINGSTOREY**

**that MEET the following requirements**

- MUST HAVE attribute **Name** **matching the pattern [0-2][0-2].\*floor**

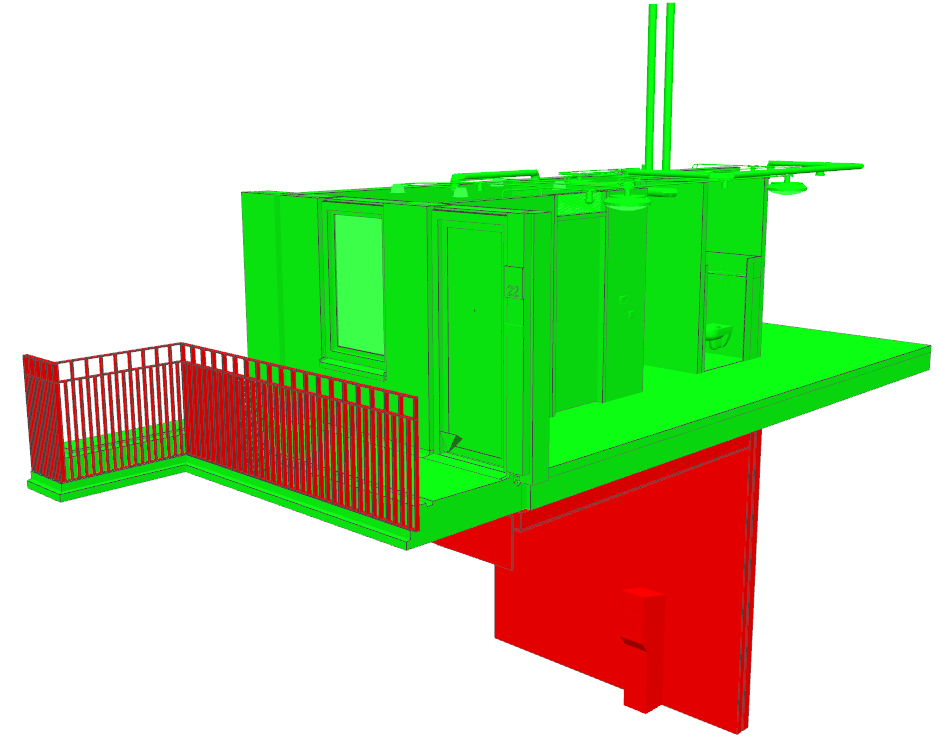
**Instructions**

Add a list of names in the Requirements by use of regular expression



**Results**

The model contains 3 building storeys: 00 nulth floor, 01 first floor and 02 second floor. 00 nulth floor is obviously wrong. However, with this regular expression you will not find it.



## **ENTITY: Walls must have allowed Predefined Types - ENT01**

These are the Predefined Types. Be aware of not allowed types. <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcWallTypeEnum.htm>

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MAY contain entities that have**

- IFC class **IFCWALL**

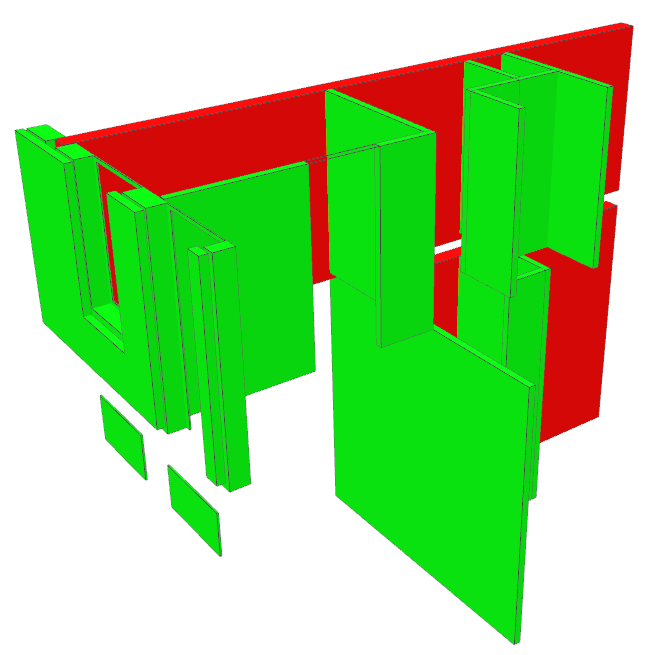
**that MEET the following requirements**

- MUST HAVE IFC class **IFCWALL** with predefined type **ELEMENTEDWALL (or: MOVABLE; PARAPET; PARTITIONING; PLUMBINGWALL; RETAININGWALL; SHEAR; SOLIDWALL; WAVEWALL; USERDEFINED; NOTDEFINED)** (Instructions: Do not allow for POLYGONAL and STANDARD as they are deprecated in the current Schema.)

**Instructions**

Find Walls and check for all of the Predefined Types that are allowed in the Schema.

Use the required cardinality: minOccurs="1" maxOccurs="unbounded".



**Results**

The red walls are an IfcWall with predefined type: STANDARD. This enumeration is deprecated in IFC4x3 so this is not in the list and should be the result of this specification.

## **ENTITY: Walls may have allowed Predefined Types - ENT02**

These are the Predefined Types. Be aware of not allowed types. <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcWallTypeEnum.htm>

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

- IFC class **IFCWALL**

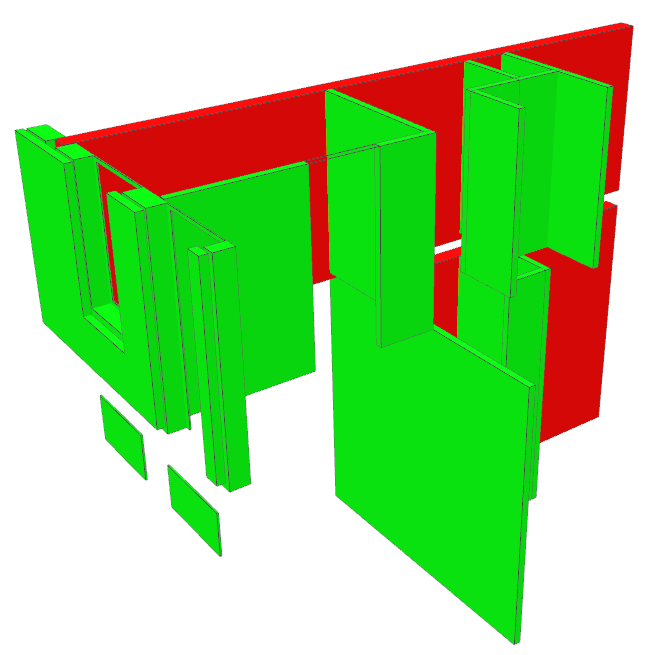
**that MEET the following requirements**

- MUST HAVE IFC class **IFCWALL** with predefined type **ELEMENTEDWALL (or: MOVABLE; PARAPET; PARTITIONING; PLUMBINGWALL; RETAININGWALL; SHEAR; SOLIDWALL; WAVEWALL; USERDEFINED; NOTDEFINED)** (Instructions: Do not allow for POLYGONAL and STANDARD as they are deprecated in the current Schema.)

**Instructions**

Find Walls and check for all of the Predefined Types that are allowed in the Schema.

Use the optional cardinality: minOccurs="0,"maxOccurs="unbounded".



**Results**

The red walls are an IfcWall with predefined type: STANDARD. This enumeration is deprecated in IFC4x3 so this is not in the list and should be the result of this specification.

## **ENTITY: Walls must not have Predefined Types that are not allowed - ENT03**

* Walls may not have a Predefined Type that is deprecated. <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcWallTypeEnum.htm>

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST NOT contain entities that have**

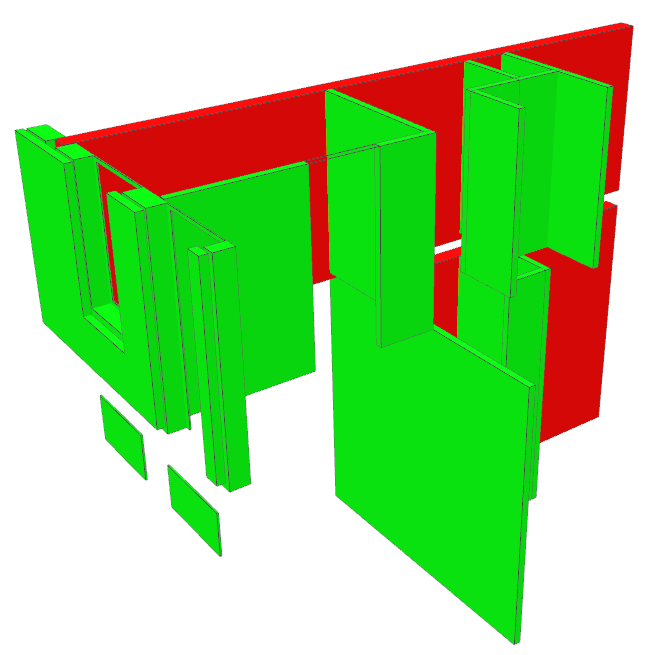
- IFC class **IFCWALL** with predefined type **POLYGONAL (or: STANDARD)**

**that MEET the following requirements**

- ...

**Instructions**

Find Walls and check for all of the Predefined Types that are not allowed in the Schema.Use the prohibited cardinality: minOccurs="0" maxOccurs="0".IDS documentation reads: requirements are ignored in the verification of models for the prohibited cardinality.



**Results**

The red walls are an IfcWall with predefined type: STANDARD. This enumeration is deprecated in IFC4x3 so this is not in the list and should be the result of this specification.

## **CLASSIFICATION: walls must have CCI classification - CL01**

Checks if the classification is present.This specification has an Entity as applicability.

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

- IFC class **IFCWALL**

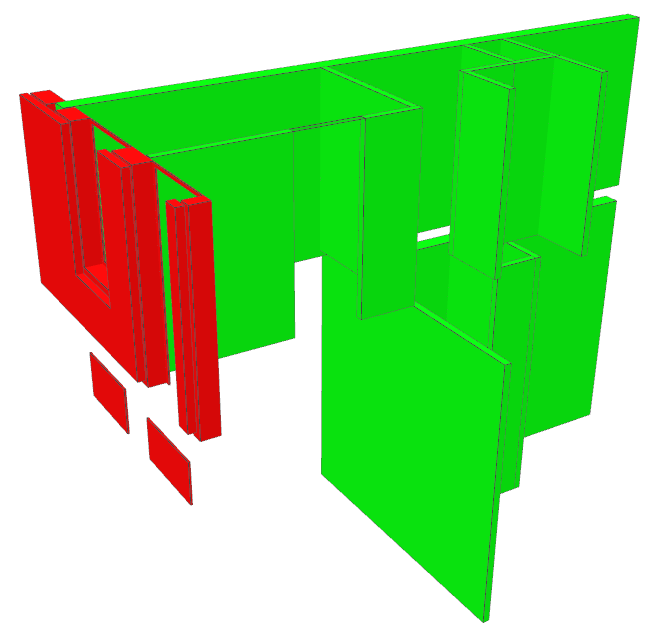
**that MEET the following requirements**

- MUST HAVE classification **CCI Construction**

- URI: <https://identifier.buildingsmart.org/uri/molio/cciconstruction/1.0/class/L-AD>

**Instructions**

Find all walls, they need to have a CCI classification



**Results**

The red walls do not have a CCI Classification code.

## **CLASSIFICATION: concrete walls must have CCI classification - CL02**

Checks if the classification is correct. This specification has an attribue as applicability as well as an Entity.

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

- IFC class **IFCWALL**

- attribute **Name** **matching the pattern .\*beton.\***

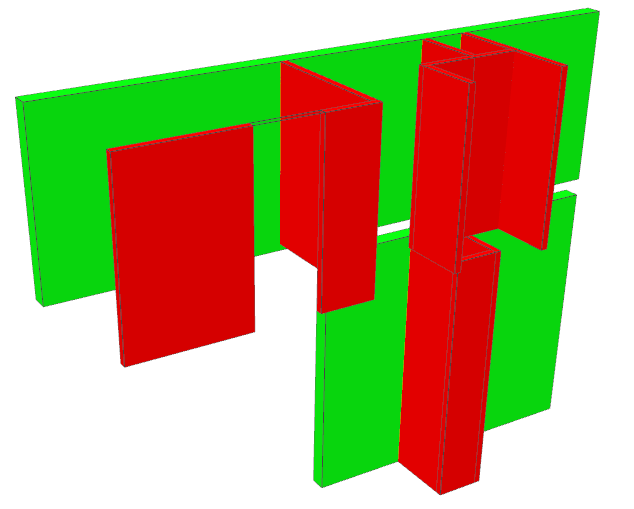
**that MEET the following requirements**

- MUST HAVE classification **CCI Construction.L-AD**

- URI: <https://identifier.buildingsmart.org/uri/molio/cciconstruction/1.0/class/L-AD>

**Instructions**

Find all elements containing the word 'beton' (Dutch for concrete) in the Attribute 'Name', they need to have a CCI classification LA-D.



**Result**

The red walls have a name, 22\_cellen**beton**\_100, that contains the word ‘beton’. The CCI classification is L-B instead of L-AD.

## **CLASSIFICATION: CCI classification L-Ad must be a wall. - CL03**

Reversed specification that wlil find a Slab that has a Wall-code.

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

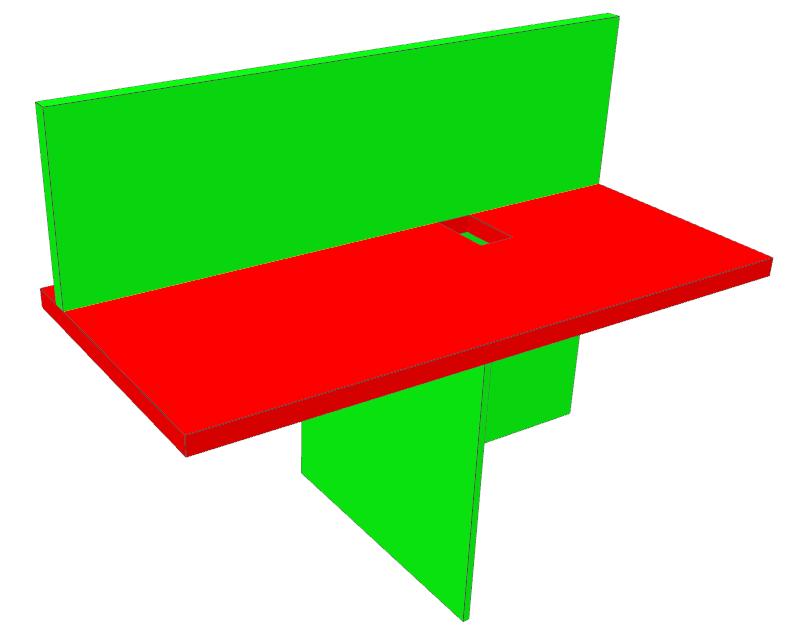
- classification **CCI Construction.L-AD**

**that MEET the following requirements**

- MUST HAVE IFC class **IFCWALL**

**Instructions**

Find all elements with classification L-AD, they must be walls.



**Result**

The red slab has a CCI classification of L-AD but that’s wrong. L-AD is walls.

## **PROPERTY: IsExternal in Requirement - PROP01**

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/property/IsExternal.htm>

- If IsExternal in the Pset, this entity should have it defined.

- IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

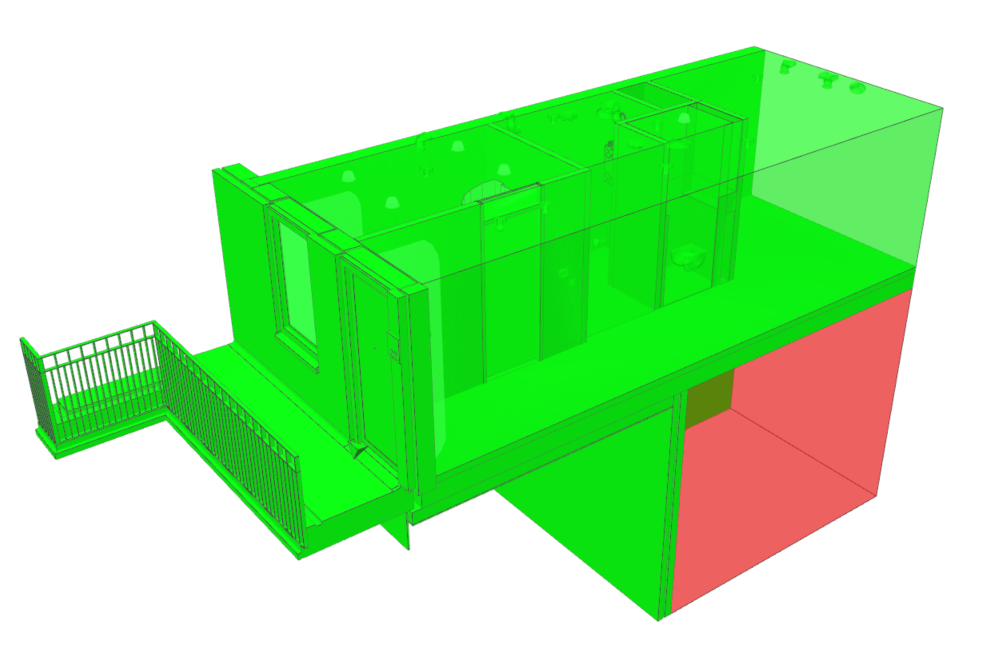
- IFC class **IFCWALL (or: IFCSLAB; IFCRAILING; IFCCOVERING; IFCROOF; IFCSPATIALZONE; IFCSTAIR; IFCPLATE; IFCBEAM; IFCCHIMNEY; IFCRAMP; IFCJUNCTIONBOX; IFCCOLUMN; IFCMEMBER; IFCSPACE; IFCBUILDINGELEMENTPROXY; IFCDOOR; IFCWINDOW; IFCSHADINGDEVICE; IFCSIGN; IFCCURTAINWALL)**

**that MEET the following requirements**

- MUST HAVE property **IsExternal** of PSet matching the pattern **Pset\_.\*Common** **(IFCBOOLEAN)**

**Instructions**

Find out if all entities that have IsExternal in their Common Psets actually have this Property defined.



**Result**

One space does not have IsExternal set in the Pset\_SpaceCommon.

## **PROPERTY: IsExternal in Applicability - PROP02**

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/property/IsExternal.htm>

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

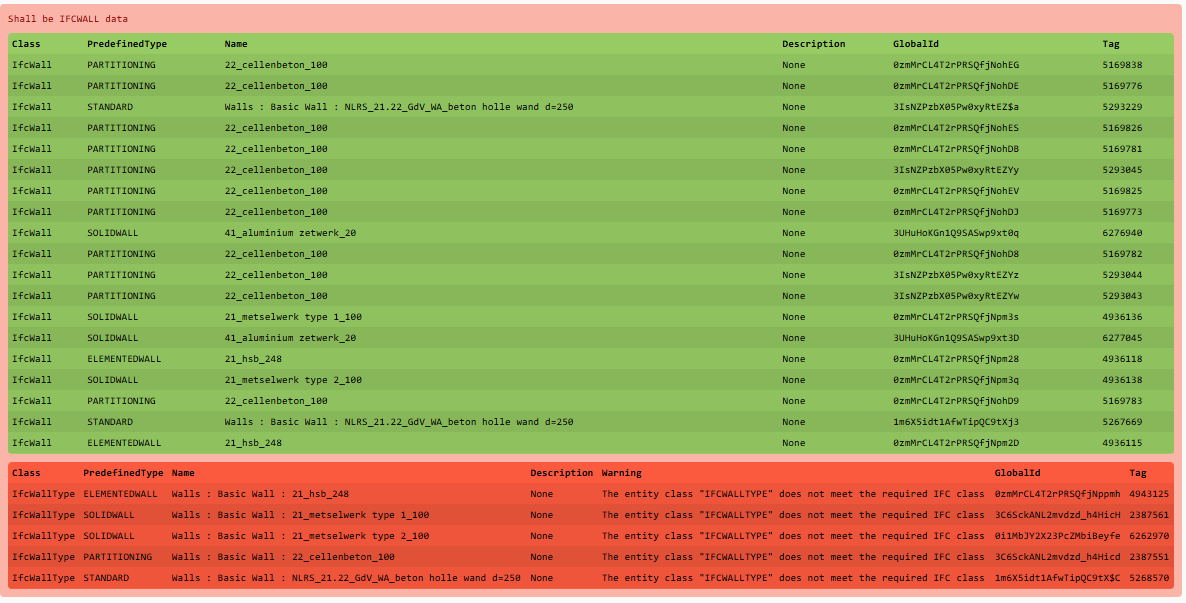
- property **IsExternal** of PSet **Pset\_WallCommon** **(IFCBOOLEAN)**

**that MEET the following requirements**

- MUST HAVE IFC class **IFCWALL** (Instructions: For this specification, IfcWallType has been left out to generate some failures.)

**Instructions**

IsExternal in Pset\_WallCommon should be IfcWall



**Result**

This should always be correct right? If your tool supports the validation of Types, you will get mistakes. If not, the specification will pass.

## **PROPERTY: External Railing should have IsExternal = True - PROP03**

<https://data.ketenstandaard.nl/publications/nlsfb/2021/781eb5a2-6424-483d-b050-73a7bbff5775>

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MAY contain entities that have**

- IFC class **IFCRAILING**

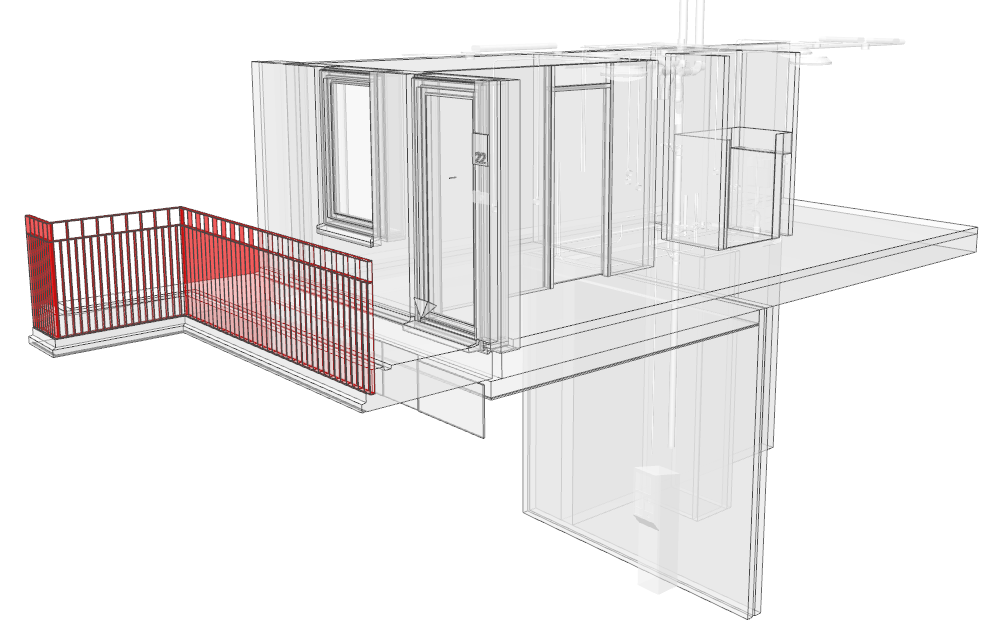
- classification **NL-SfB Tabel 1.34.12**

**that MEET the following requirements**

- MUST HAVE property **IsExternal** of PSet **Pset\_RailingCommon** **(IFCBOOLEAN) = true** (Instructions: DataType: IFCBOOLEAN, URI: <https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/IsExternal>

**Instructions**

Find the Railing with NL-SfB ClassificationCode: 34.12: balustrades, handrails; balustrades, outer balustrades'. Outer balustrades should have the property IsExternal = True.



**Result**

IsExternal in Pset\_RailingCommon in the Railing is set to False, it should be True.

## **PROPERTY: ticket vending machine is eating my money! - PROP04**

<https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/IfcElectricAppliance.htm>

IFC versions: **IFC4X3\_ADD2**

**The model MAY contain entities that have**

- IFC class **IFCELECTRICAPPLIANCE.VENDINGMACHINE**

**that MEET the following requirements**

- MUST HAVE property **TicketVendingMachineType** of PSet **Pset\_TicketVendingMachine** **(IFCLABEL) = TICKETVENDINGMACHINE** (Instructions: DataType: IFCLABELTicketVendingMachineType is a enumerated property, chose the correct one from this list: <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/PEnum_TicketVendingMachineType.htm>

URI: <https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/TicketVendingMachineType>

- MUST HAVE property **TicketStuckRatio** of PSet **Pset\_TicketVendingMachine** **(IFCRATIOMEASURE) < 10** (Instructions: DataType: IFCRATIOMEASURE <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/Pset_TicketVendingMachine.htm>

URI: <https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/TicketStuckRatio>

- MUST HAVE property **MoneyStuckRatio** of PSet **Pset\_TicketVendingMachine** **(IFCRATIOMEASURE) ≥ 50** (Instructions: DataType: IFCRATIOMEASURE <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/Pset_TicketVendingMachine.htm>

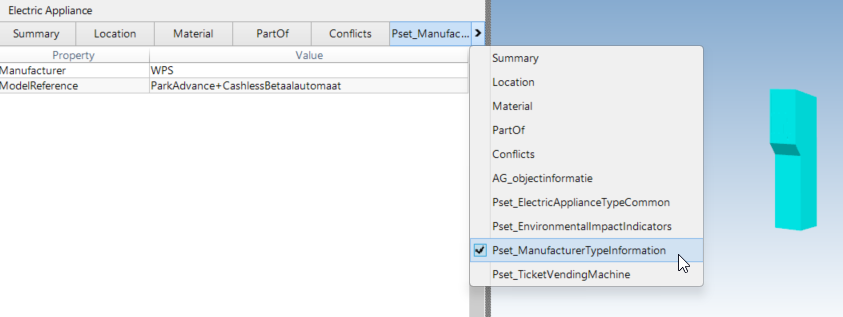
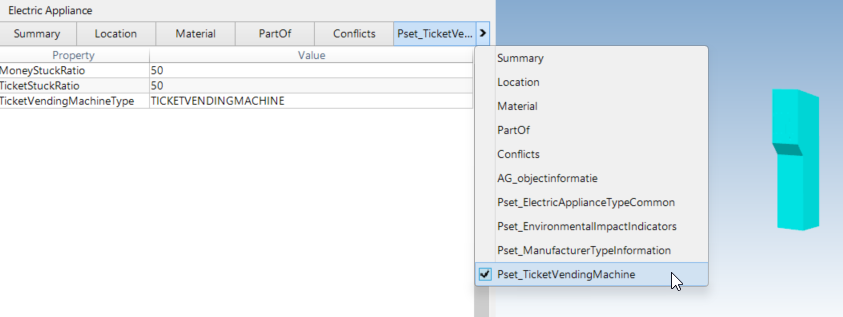
URI: <https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/MoneyStuckRatio>

- MUST HAVE property **Manufacturer** of PSet **Pset\_ManufacturerTypeInformation** **(IFCLABEL) = WPS** (Instructions: DataType: IFCLABEL <https://standards.buildingsmart.org/IFC/RELEASE/IFC4_3/HTML/lexical/Pset_ManufacturerTypeInformation.htm>

URI: <https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/Manufacturer>

**Instructions**

Find the Ticket vending machine and add Properties Manufacturer, TicketVendingMachineType, TicketStuckRatio and MoneyStuckRatio. Set the ratios as bounds.



**Result**

TicketStuckRatio should generate a mistake. Test a few values that are equal to, lesser or greater than 50 to see how your specification works.

## **MATERIAL: specific element must have a specific name - MAT01**

Validating materials is usually never done like this. But if a tool does not support RegEx, this is a simple test.

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

- IFC class **IFCWALL.SOLIDWALL**

- attribute **Name** **= 21\_metselwerk type 1\_100**

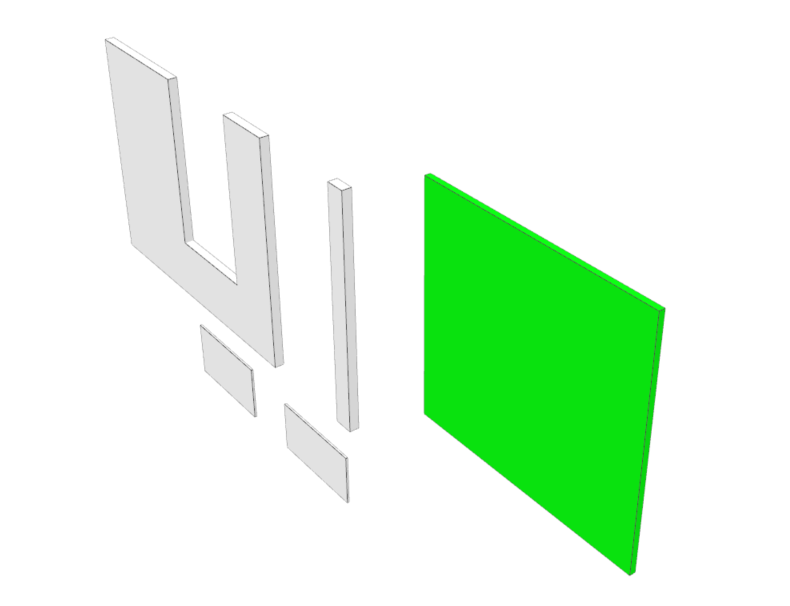
**that MEET the following requirements**

- MUST HAVE material **steenachtig\_baksteen\_steen\_roodroze**

URI: <https://identifier.buildingsmart.org/uri/nkt/naakt/2.4/class/steenachtig_baksteen_steen>

**Instructions**

Find walls with name '21\_metselwerk type 1\_100', the Material should be steenachtig\_baksteen\_steen\_roodroze



**Result**

The correct wall should be found

## **MATERIAL: brickwork should start with steenachtig\_baksteen\_steen - MAT02**

Usually you need to filter elements and then, the materialname must have a specific syntax ore a specific word in the name. Regular expressions are key in these type of specifications.

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

- IFC class **IFCWALL.SOLIDWALL**

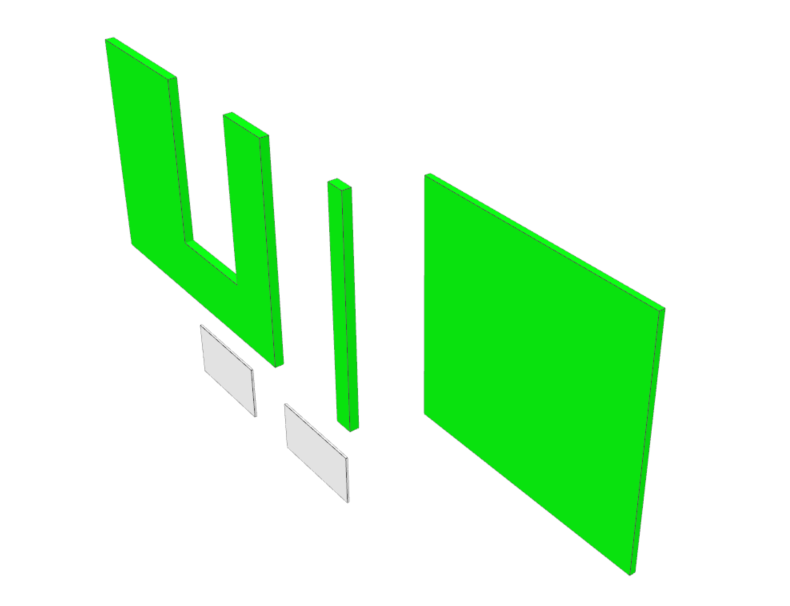
- attribute **Name** **matching the pattern .\*metselwerk.\***

**that MEET the following requirements**

- MUST HAVE material matching the pattern **steenachtig\_baksteen\_steen.\***

**Instructions**

Find walls with a name that contains 'metselwerk', the Material should start with steenachtig\_baksteen\_steen



**Result**

The correct walls will be found.

## **MATERIAL: a materialname must have a specific property - MAT03**

https://standards.buildingsmart.org/IFC/RELEASE/IFC4\_3/HTML/lexical/PEnum\_AssemblyPlace.htm

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

- material matching the pattern **(beton|gips|hout|ntb|generiek)\_.\*\_prefab-element.\***

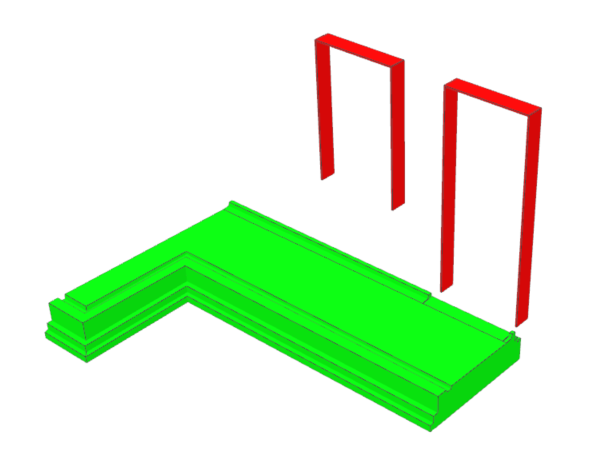
**that MEET the following requirements**

- MUST HAVE property **AssemblyPlace** of PSet **Pset\_ManufacturerTypeInformation** **(IFCLABEL) = FACTORY**

URI: <https://identifier.buildingsmart.org/uri/buildingsmart/ifc/4.3/prop/AssemblyPlace>

**Instructions**

Find elements with a materialname containing 'prefab-element'. They should have the AssemblyPlace property set as FACTORY in Pset\_ManufacturerTypeInformation



**Result**

The concrete slab is a prefab element, assembled in the Factory. The gypsum boards are prefabicated pieces but do not have the property AssemblyPlace set.

## **PARTOF: MEP part of an IfcDistributionsystem - PO01**

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

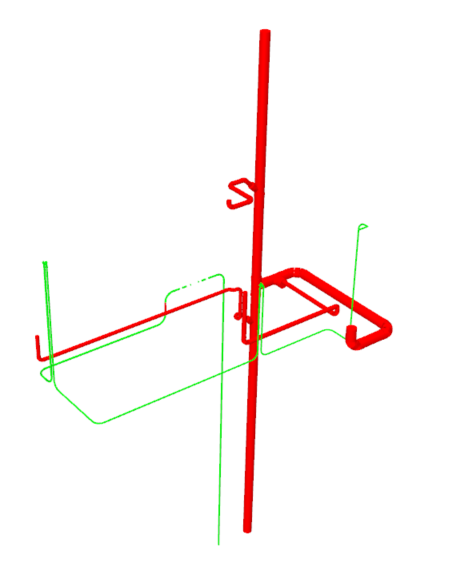
- IFC class **IFCPIPEFITTING (or: IFCPIPESEGMENT)**

**that MEET the following requirements**

- MUST HAVE relationship **IFCRELASSIGNSTOGROUP** with IFC class **IFCDISTRIBUTIONSYSTEM.DOMESTICCOLDWATER**

**Instructions**

Find some pipes, they must be part of an IfcDistributionSystem



**Result:**

Some pipesegments and fittings are not part of the IfcDistributionSystem

## **PARTOF: A smokesensor must be in a space - PO02**

IFC versions: **IFC4X3\_ADD2 IFC4**

**The model MUST contain entities that have**

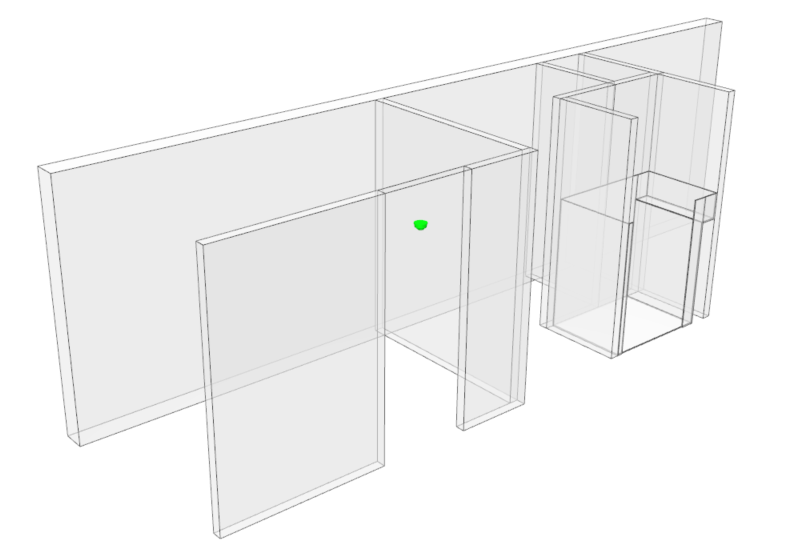
- IFC class **IFCSENSOR.SMOKESENSOR**

**that MEET the following requirements**

- MUST HAVE relationship **IFCRELCONTAINEDINSPATIALSTRUCTURE** with IFC class **IFCSPACE**

**Instructions**

Find the IfcSensor.SMOKESENSOR that must be contained it a spatial structure, an IfcSpace



**Result**

The Smokedetector is part of a space.

(In IDS version 1.0 we cannot further define what space it should be part of.)

## **PARTOF: A building must be build on a site - PO03**

IFC versions: **IFC4X3\_ADD2 IFC4 IFC2X3**

**The model MUST contain entities that have**

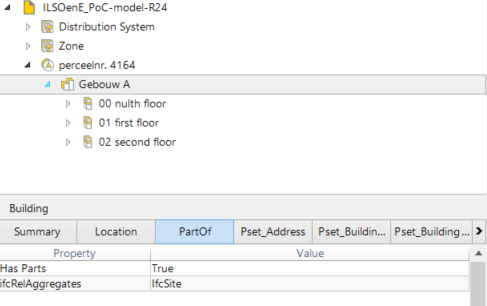
- IFC class **IFCBUILDING**

**that MEET the following requirements**

- MUST HAVE relationship **IFCRELAGGREGATES** with IFC class **IFCSITE**

**Instructions**

Find the IfcBuilding, it should aggregate into an IfcSite



**Result**

The building has a relation to the site it is on.

## **PARTOF: Door fills wall – PO04**

IFC versions: **IFC4X3\_ADD2**

**The model MUST contain entities that have**

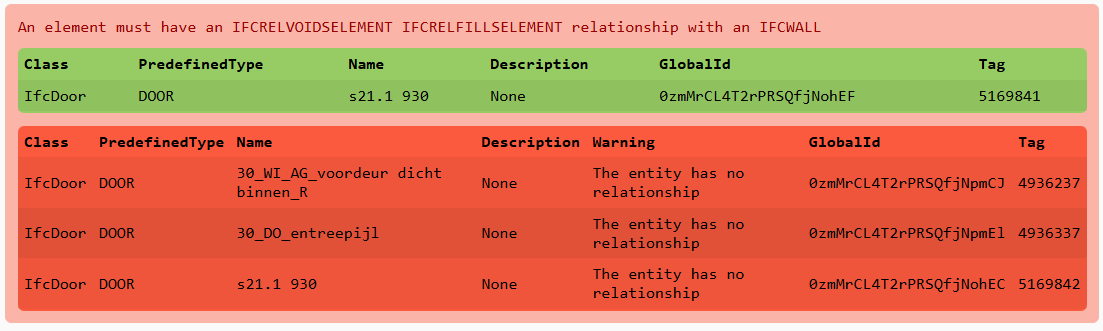
- IFC class **IFCDOOR**

**that MEET the following requirements**

- MUST HAVE relationship **IFCRELVOIDSELEMENT IFCRELFILLSELEMENT** with IFC class **IFCWALL**

**Instructions:**

**IFCRELVOIDSELEMENT IFCRELFILLSELEMENT** is a list as opposed to IFCRELAGGREGATES and the other part of relationships. It let’s you skip the IfcOpening because it’s a combination of the two relations



**Result**

Some elements are nested elements that do not void or fill an element. The one door in the interior wall does.