Webinar on OGC IndoorGML

Ki-Joune Li, Pusan National University

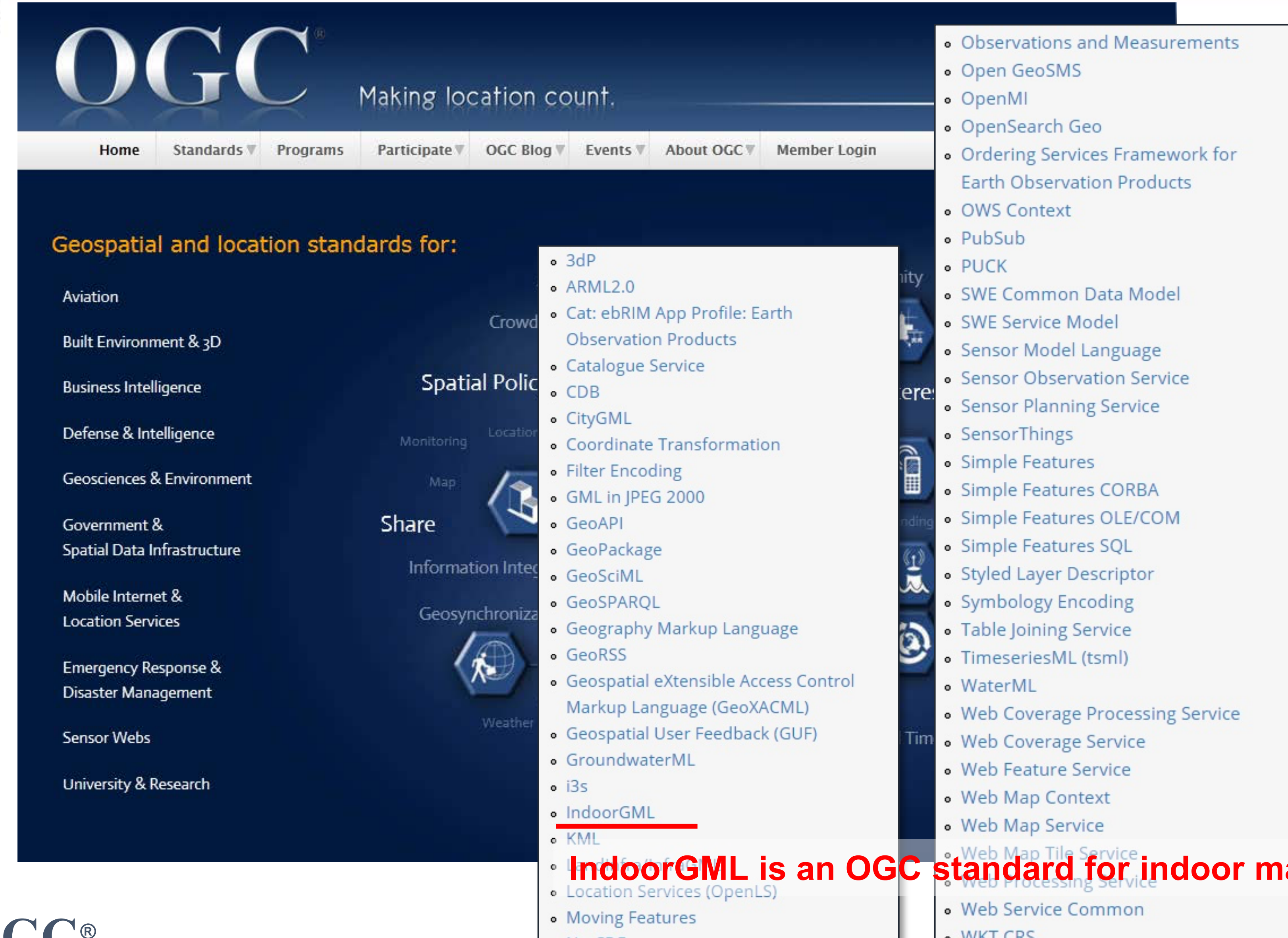
May 28, 2020

Agenda

1. Basic Concepts
2. Making IndoorGML data
3. Extending IndoorGML for your applications
4. Use-Cases
5. Summary

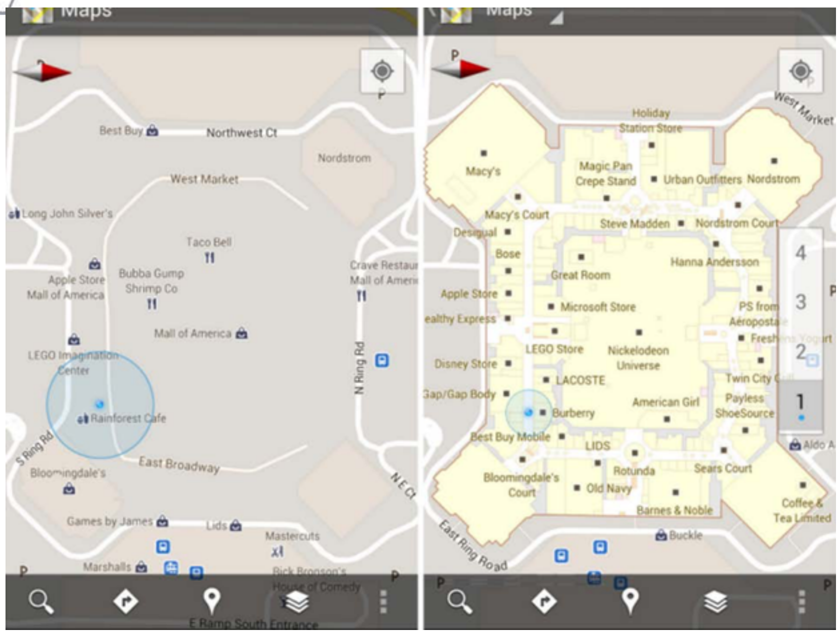
1. Basic Concepts

OGC IndoorGML



IndoorGML is an OGC standard for indoor maps

Indoor Maps



a Google indoor map

an indoor OSM

Indoor maps include
1. Indoor space structure: rooms, hallway, elevator, etc.
2. Features in indoor space: furniture, kiosks, etc.
3. Connections between rooms: doors

OGC®

OGC IndoorGML

Scope of IndoorGML

Indoor maps include

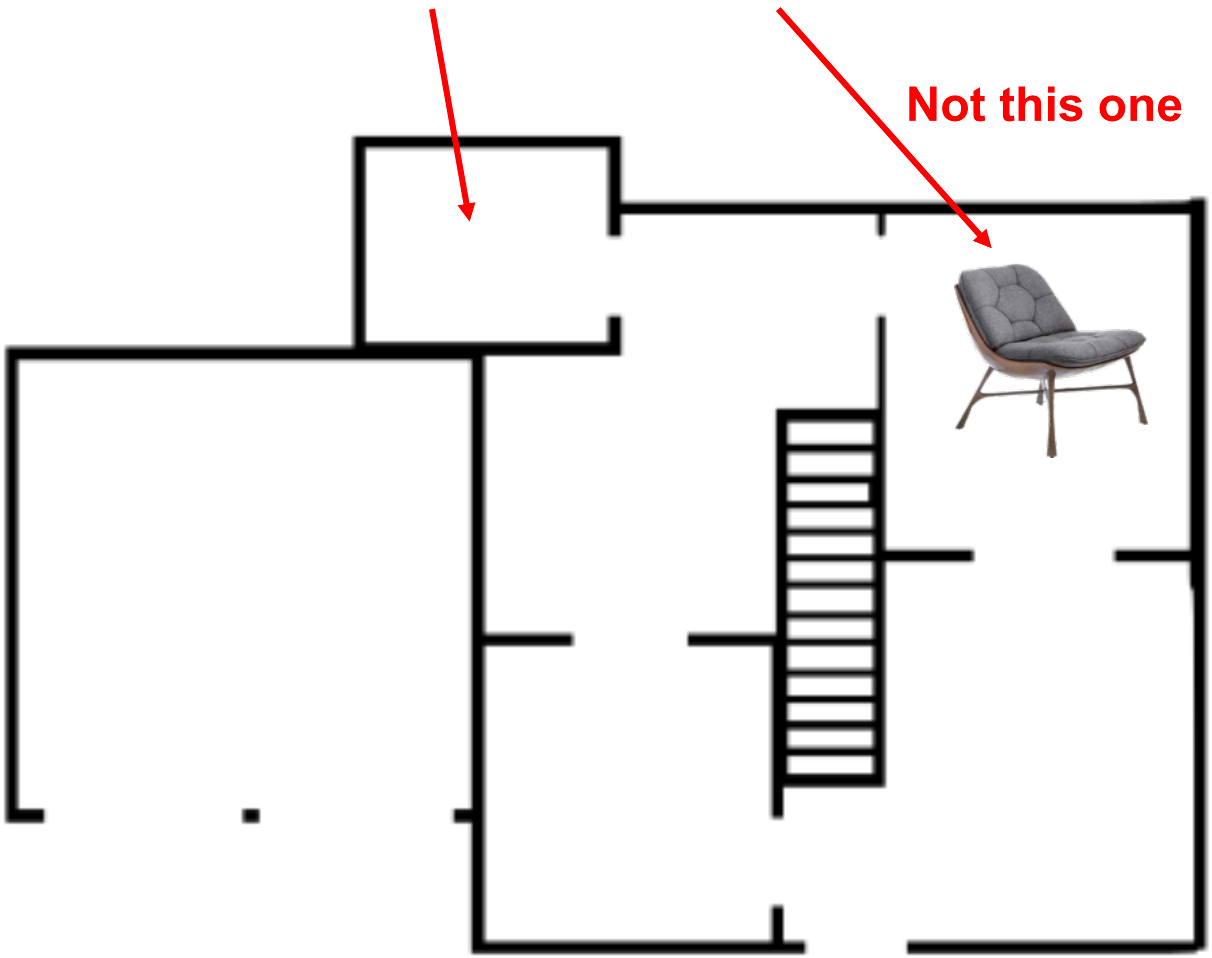
1. Indoor space structure: rooms, hallway, elevator, etc.
2. Features in indoor space: furniture, kiosks, etc.
3. Connections between rooms: doors

+ semantics

Scope of IndoorGML

IndoorGML – Space Model

- We are modeling **SPACE** not *objects* in a space



OGC®

Cellular Space Model

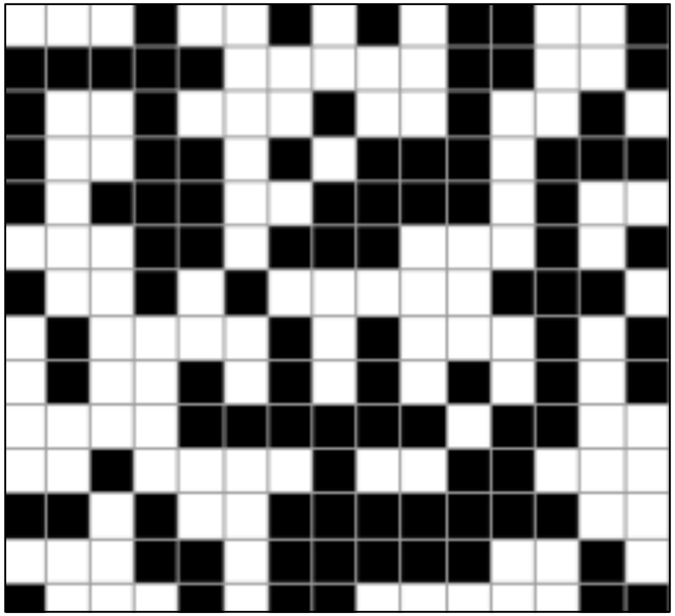
Cellular Space is

a space defined as a set of cells

- $C_i \cap C_i = \emptyset$ (non-overlapping cells)
- $\cup\, C_i \subseteq U$, where U is the entire space (2D or 3D)
- Each cell has cell ID.

IndoorGML with Cellular Space Model

- Several aspects of cellular space
  - Geometry
  - Topology
  - Multi-Layered Structure
  - Semantics
  - Indoor Partitioning
  - Hierarchical Structure
  - Integration with Other Data Set



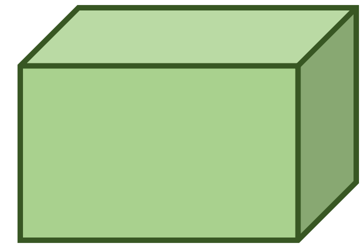
OGC

IndoorGML

Cellular Space Model – Geometry

Three options

Option 1 – 3D Solid for 3D indoor map

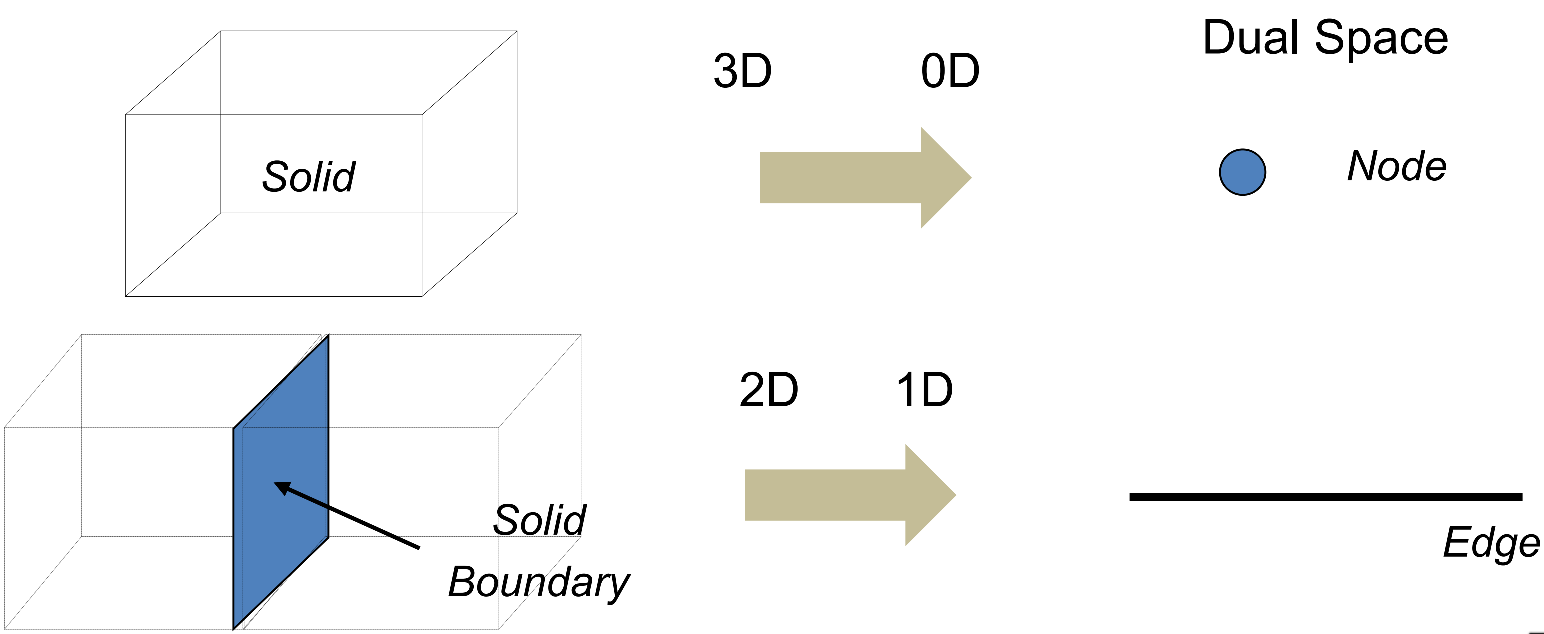


Option 2 – 2D Surface for 2D indoor map

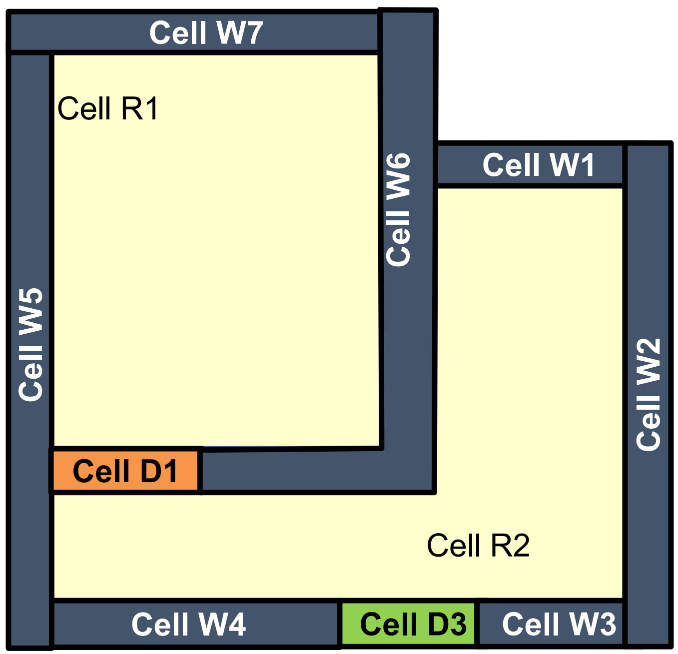
Option 3 – No Geometry for cell without boundary

Cellular Space Model – Topology

- Poincare Duality
  - Conversion from original (primal space) to dual space
  - Given a *N*-D (e.g. 3D) space,
    conversion from *k* D object → *N-k* (e.g. *3 – k*) D objects



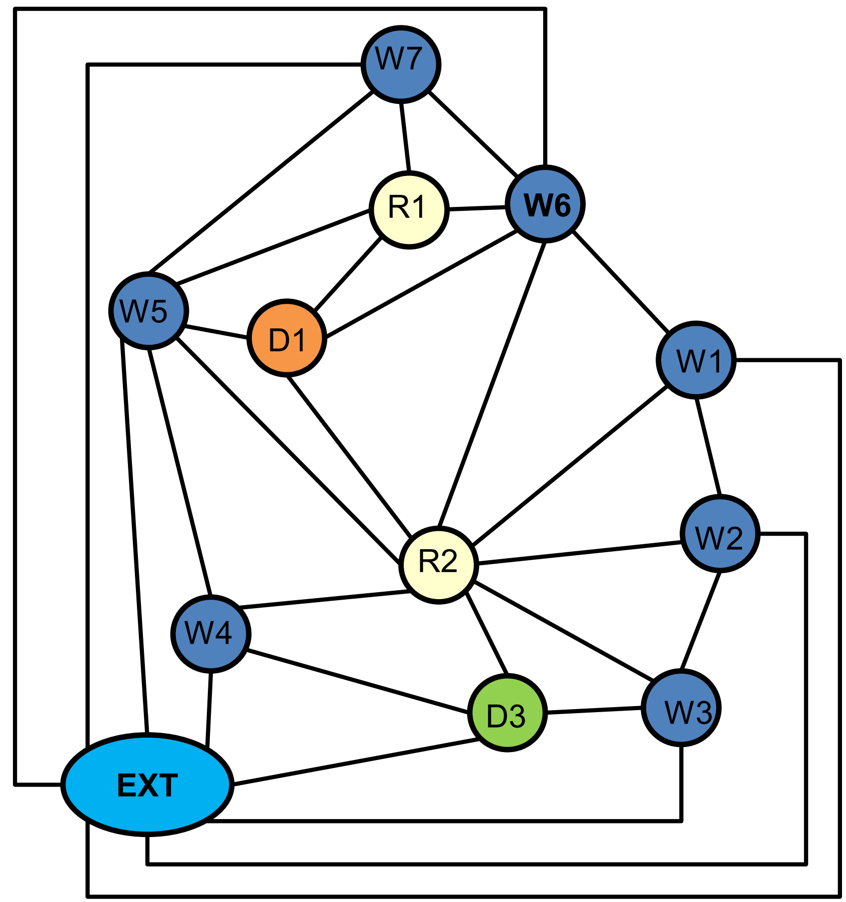
Cellular Space Model – Topology



Ext

Primal Space

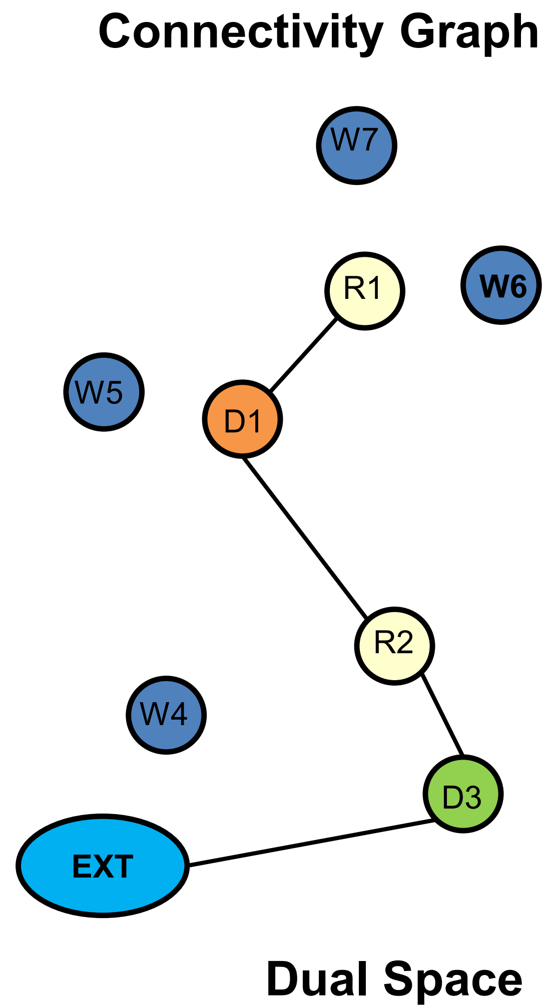
Adjacency Graph



Dual Space

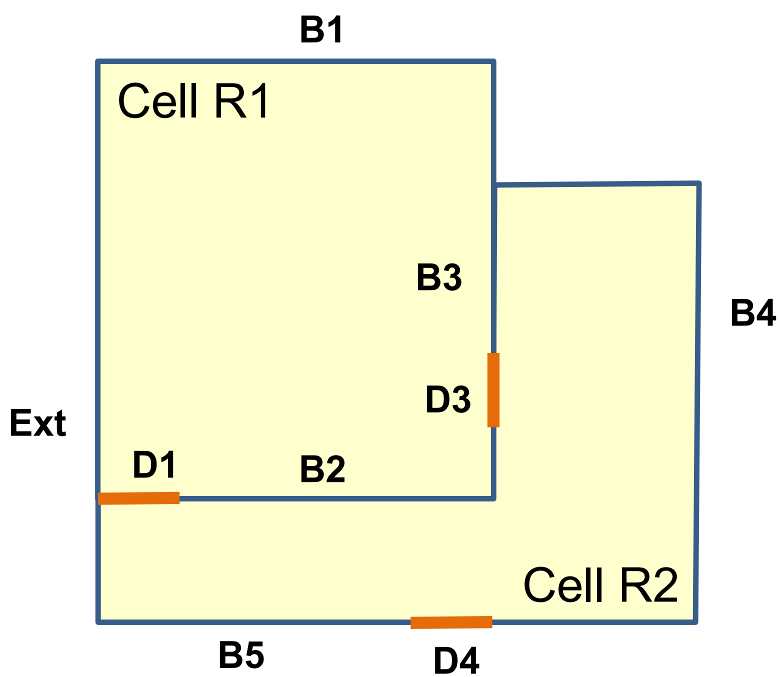
OGC IndoorGML

Cellular Space Model – Topology

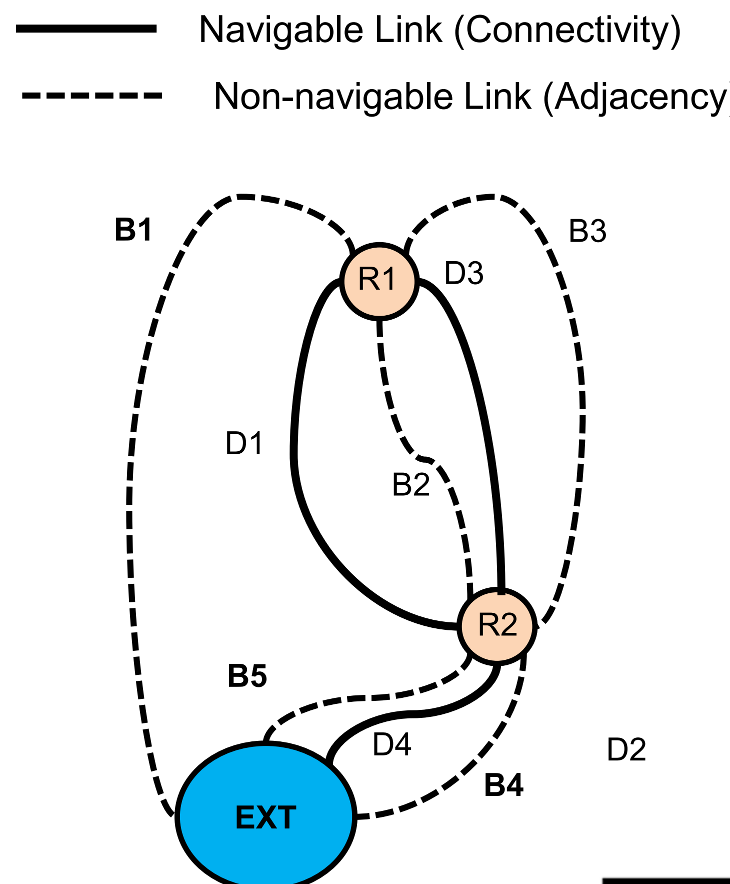


Cellular Space Model – Topology

Example: Wall and Door as Space Boundary

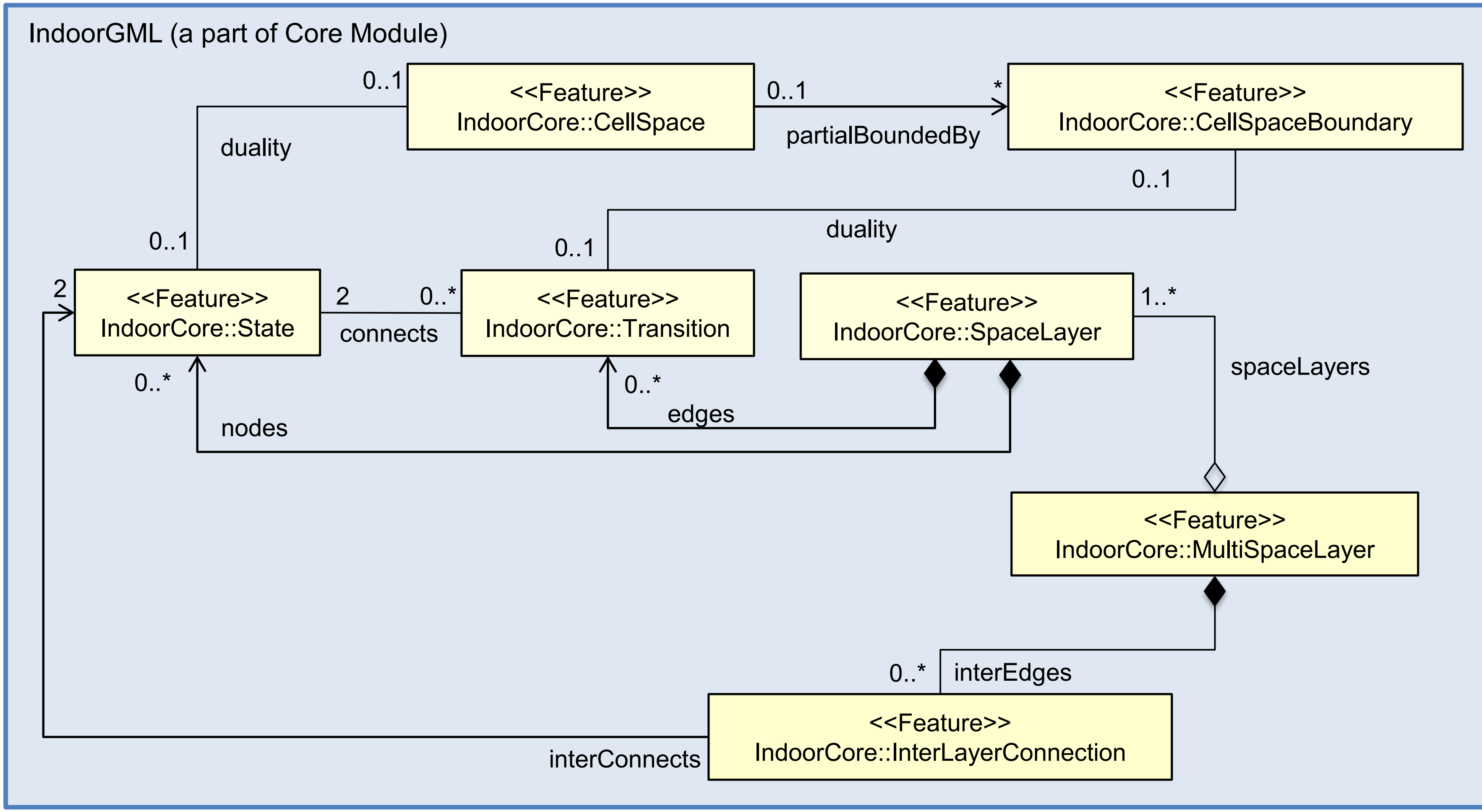


Topographic Space

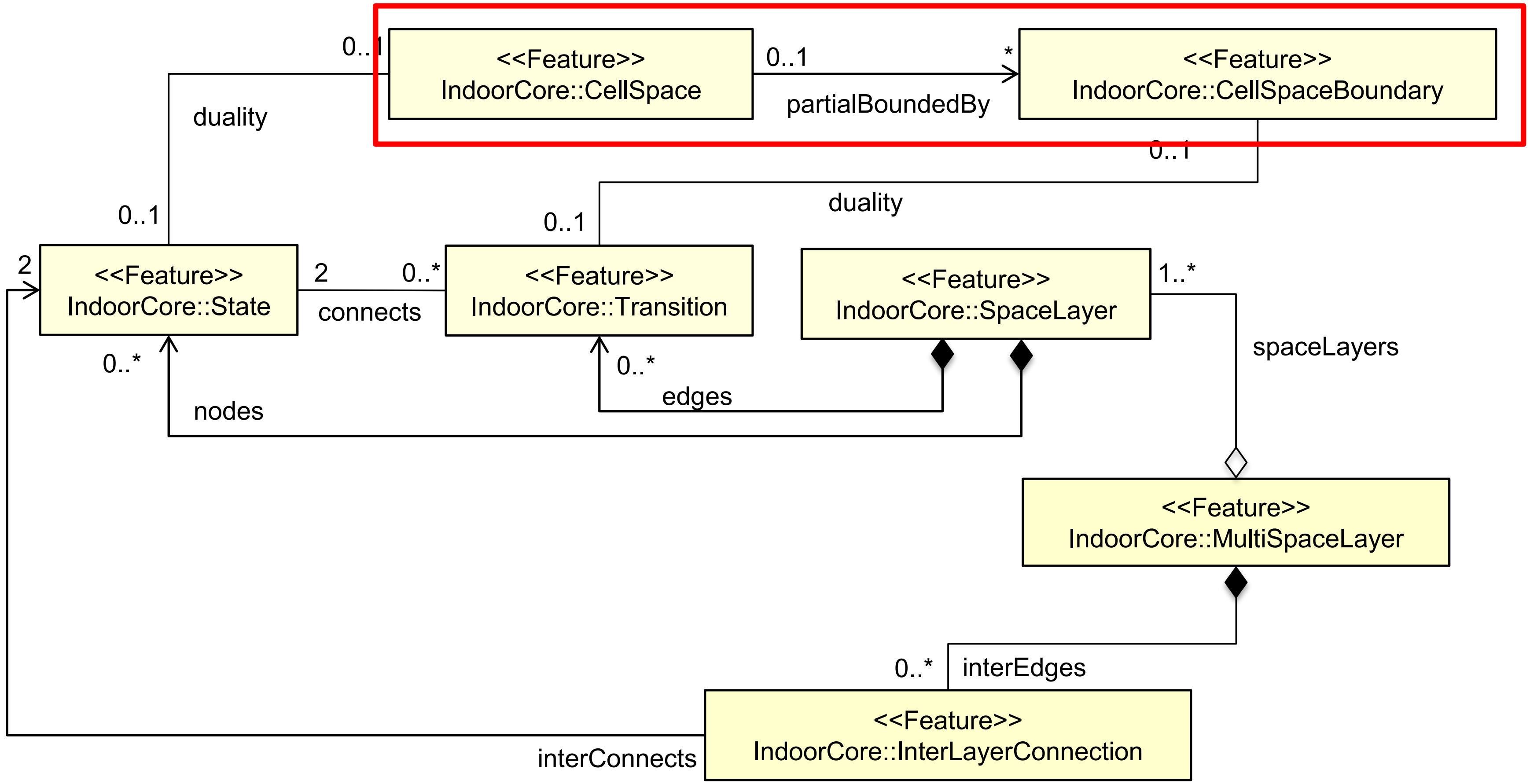


Dual Space

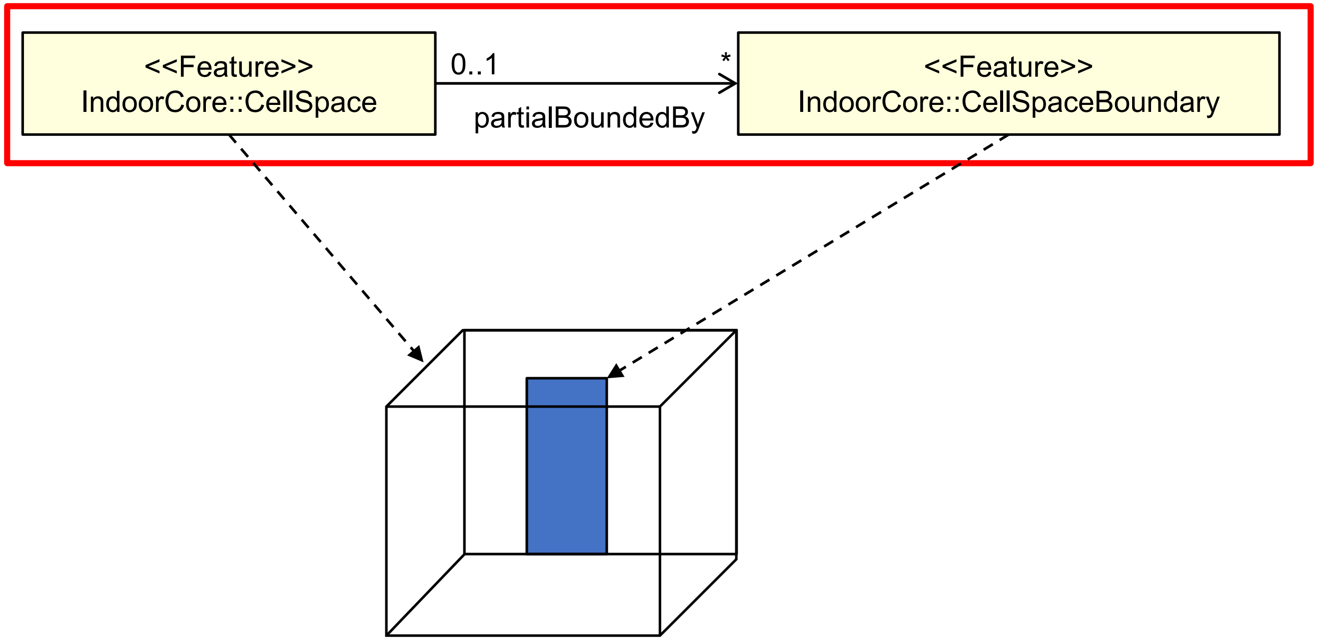
Cellular Space Model – Data Model



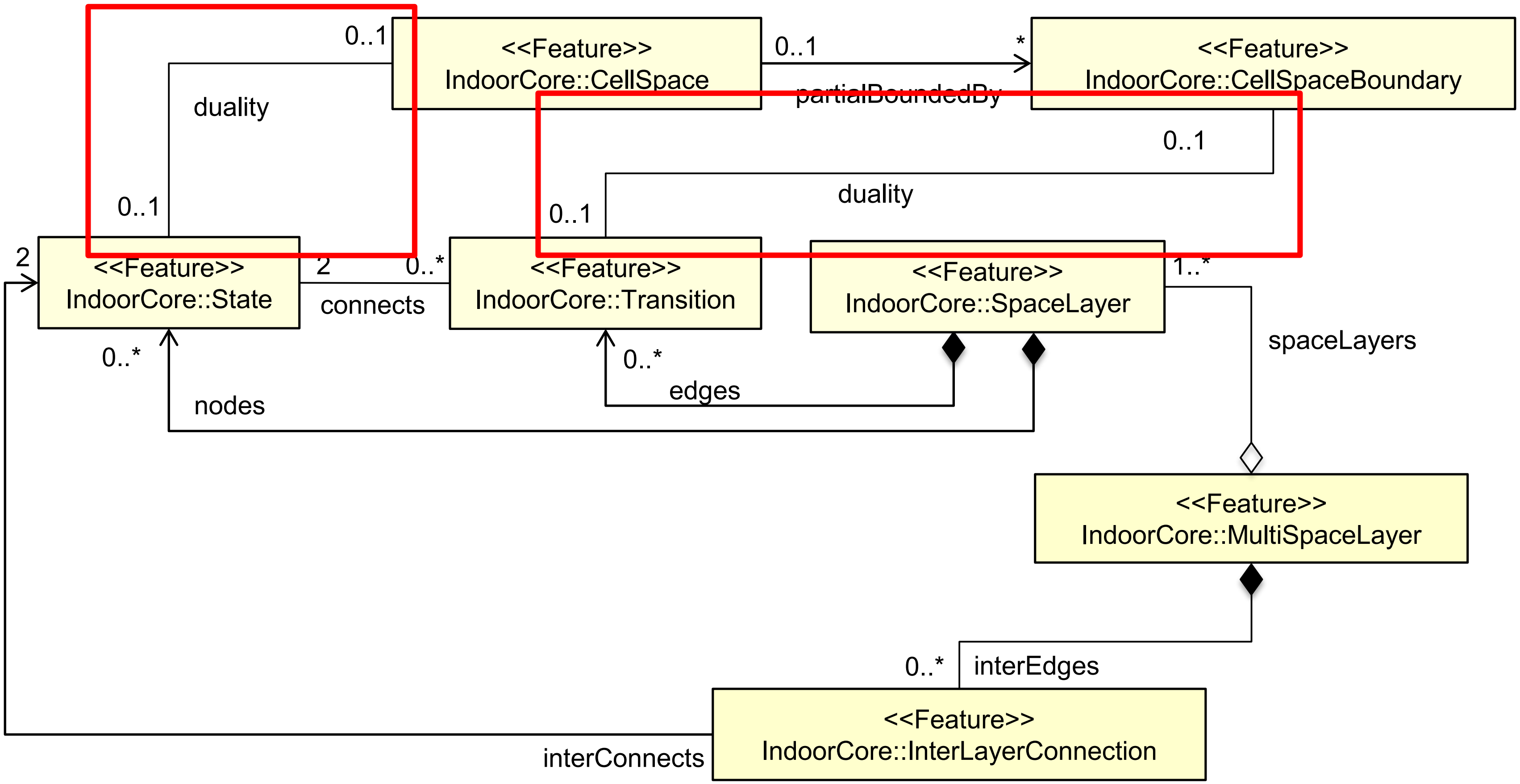
Cellular Space Model – Data Model



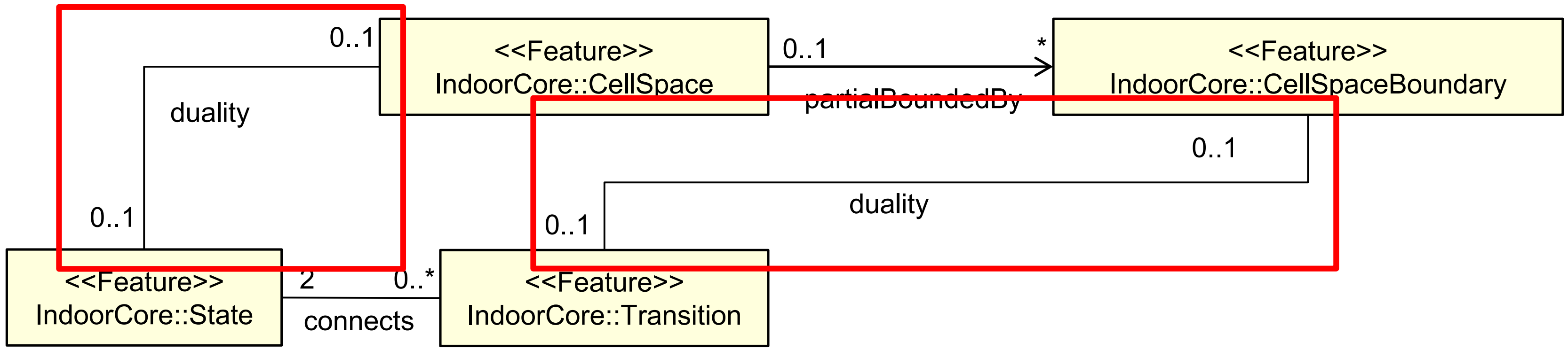
Cellular Space Model – Data Model



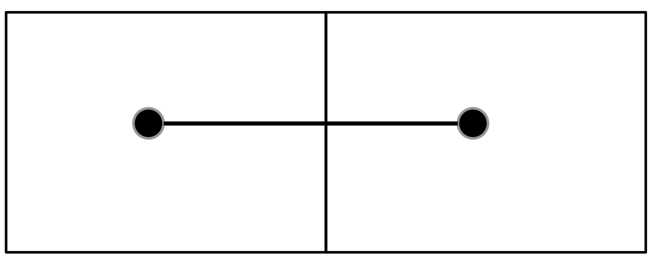
Cellular Space Model – Data Model



Cellular Space Model – Data Model



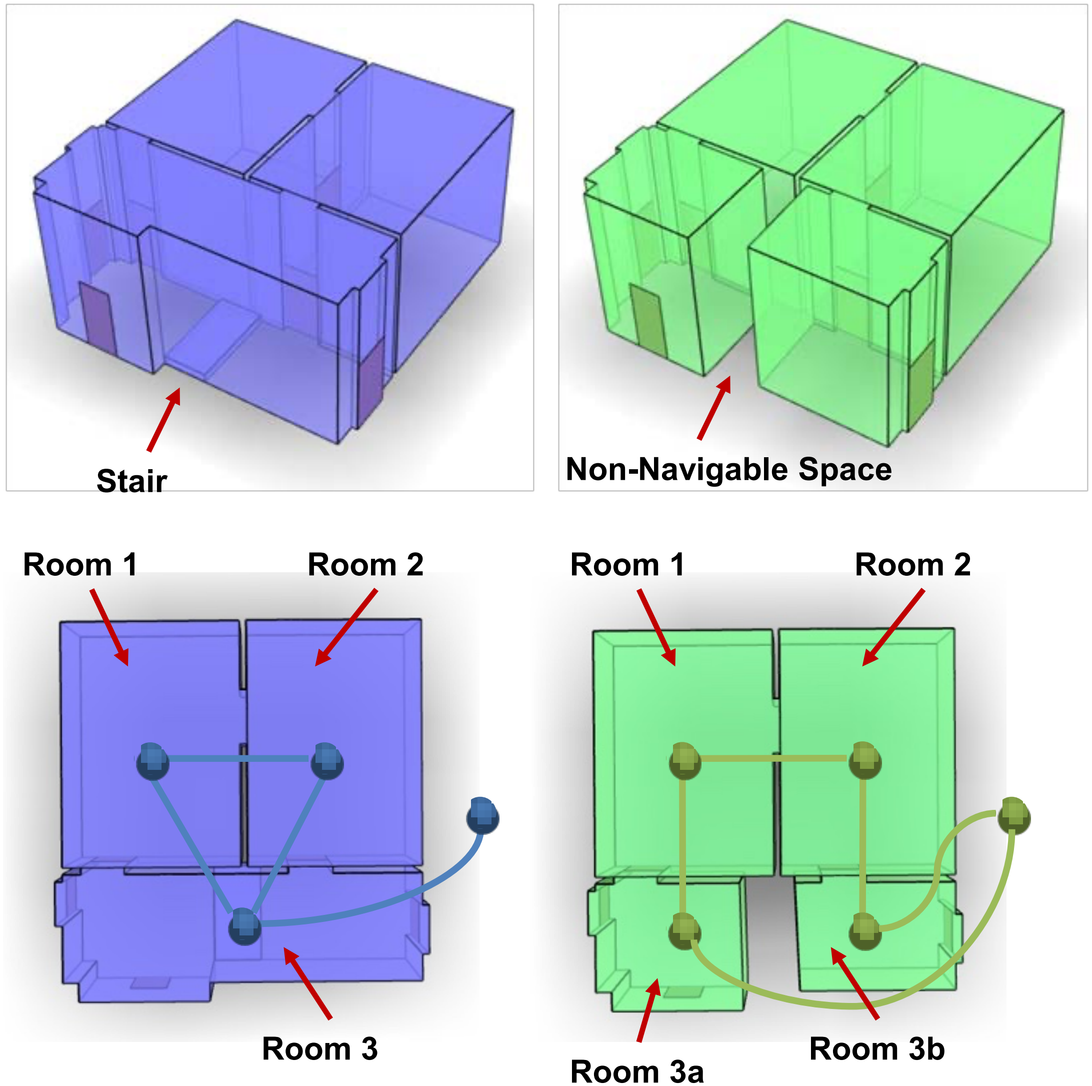
→ Duality is optional



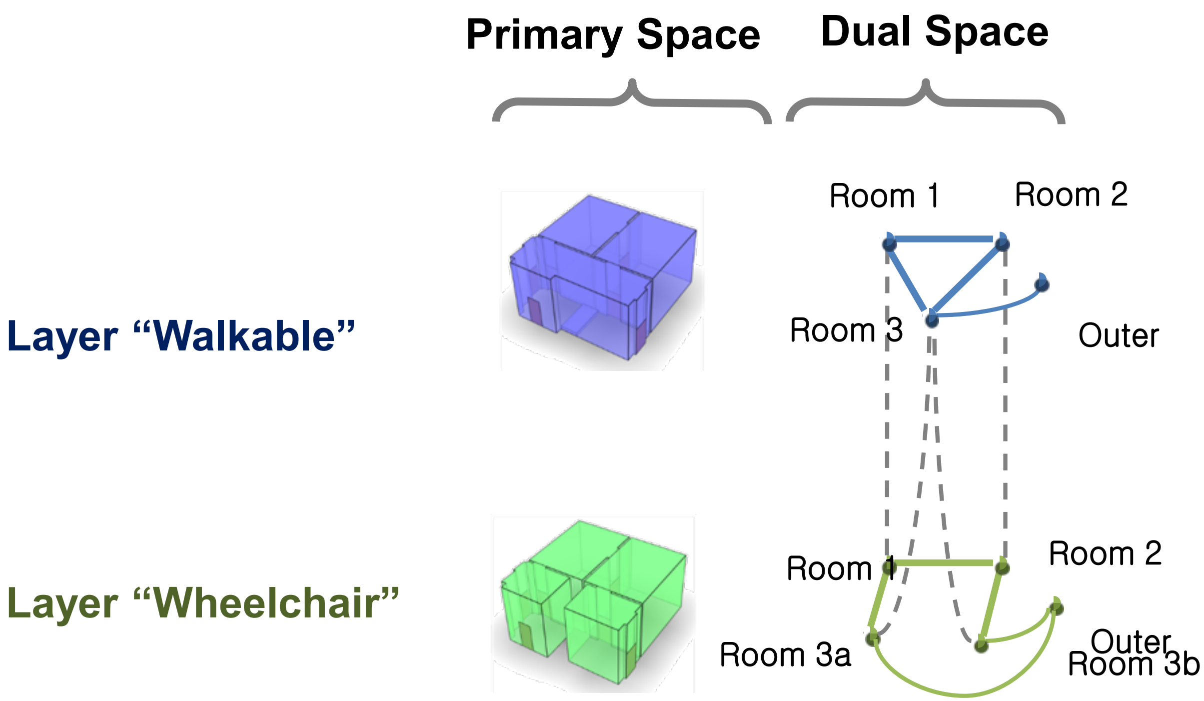
Explicit duality

State and transition without cell space and cell space boundary

Cellular Space Model – Multi-Layer Space Model



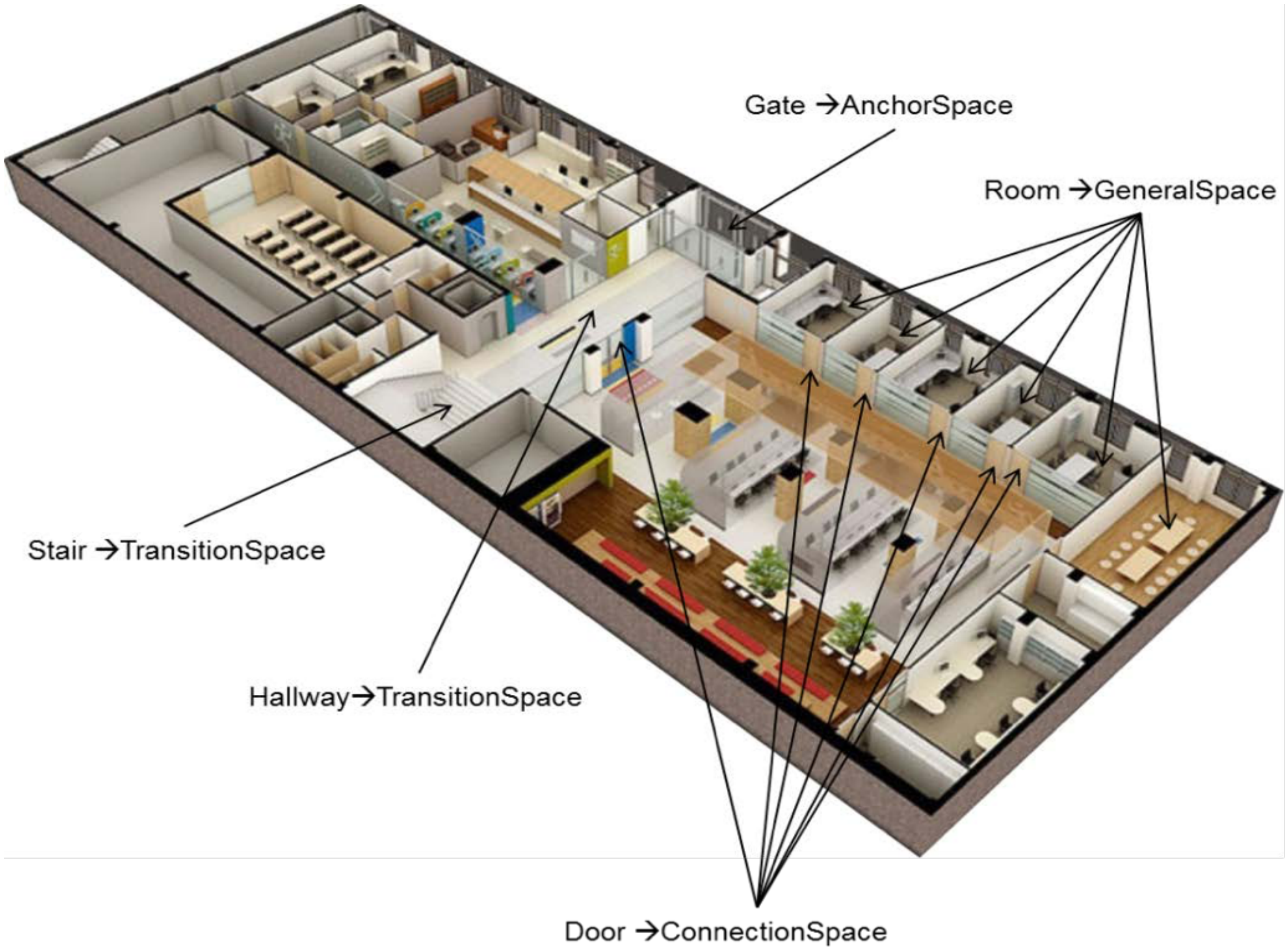
Cellular Space Model – Multi-Layer Space Model



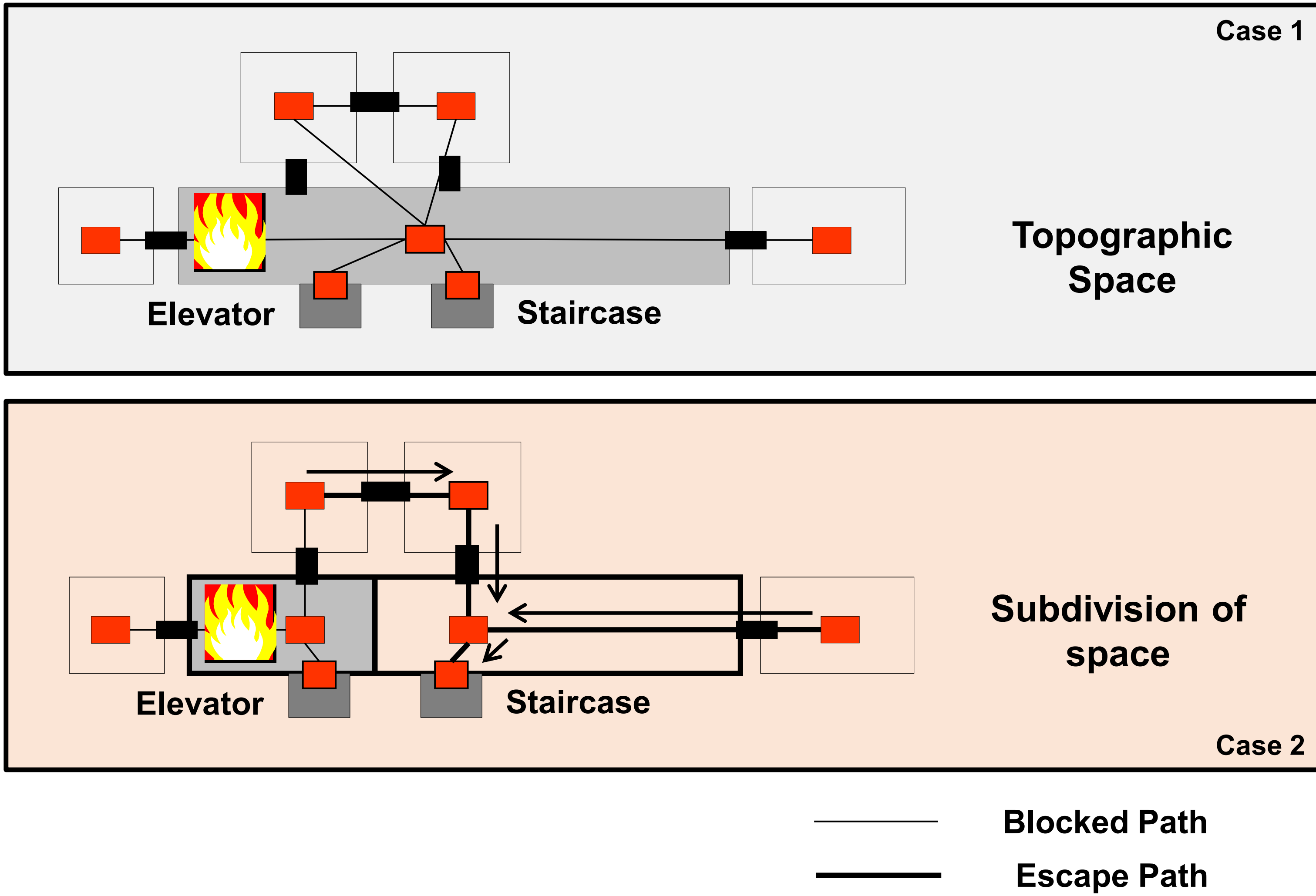
Cellular Space Model – Semantics

- Semantic Interpretation of Indoor Space
  - Classification of Indoor Space
    - Example – Room, Door, Corridor, Stair Space, Elevator Shaft, Gate
  - OmniClass (http://omniclass.org)
    - A standard for organizing all construction information
    - Provides classification system for the construction
      - By function: Kitchen, Elevator Shaft, Office Space, etc.
- Definition of Attributes
  - Names, Usage, Functions, etc..
  - Directions
  - Accessibility

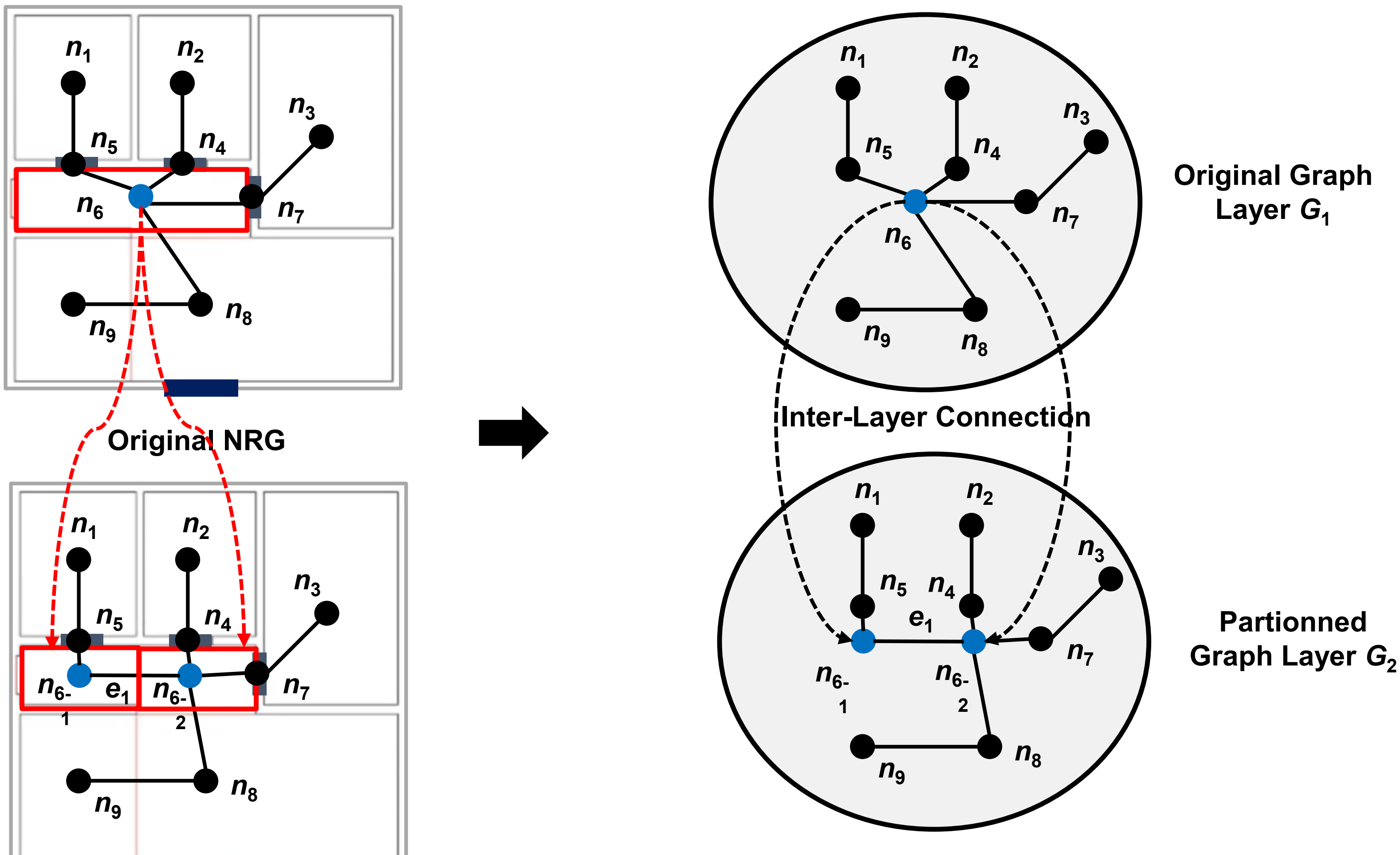
Cellular Space Model – Classification of Space



Cellular Space Model – Subspacing



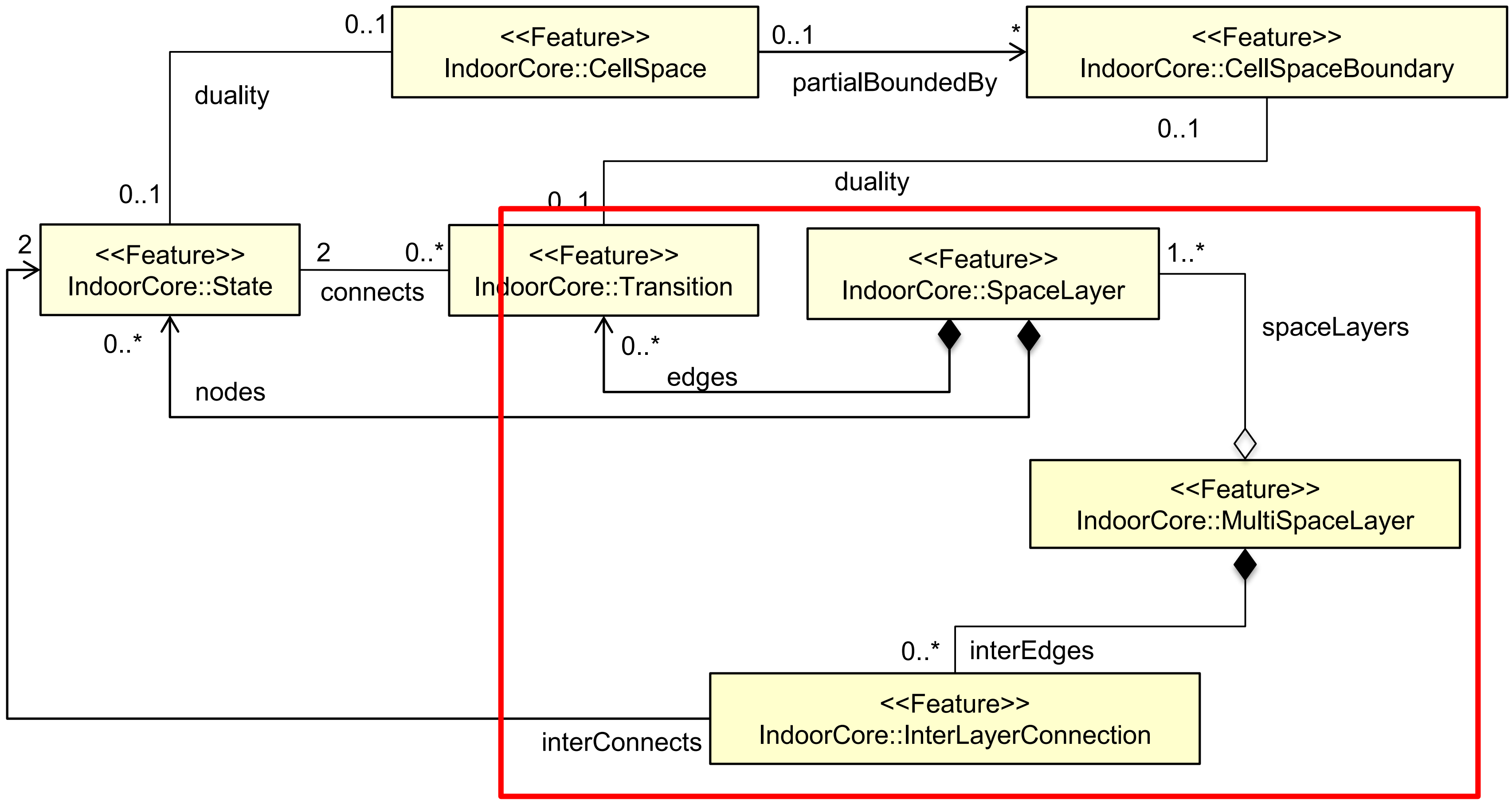
Cellular Space Model – Subspacing



Partitioned Spaces

{($G_1.n_6$, $G_2.n_{6\text{-}1}$,contains), ($G_1.n_k$, $G_2.n_k$, contains)} : *Inter-Layer Connection for* **subspacing**

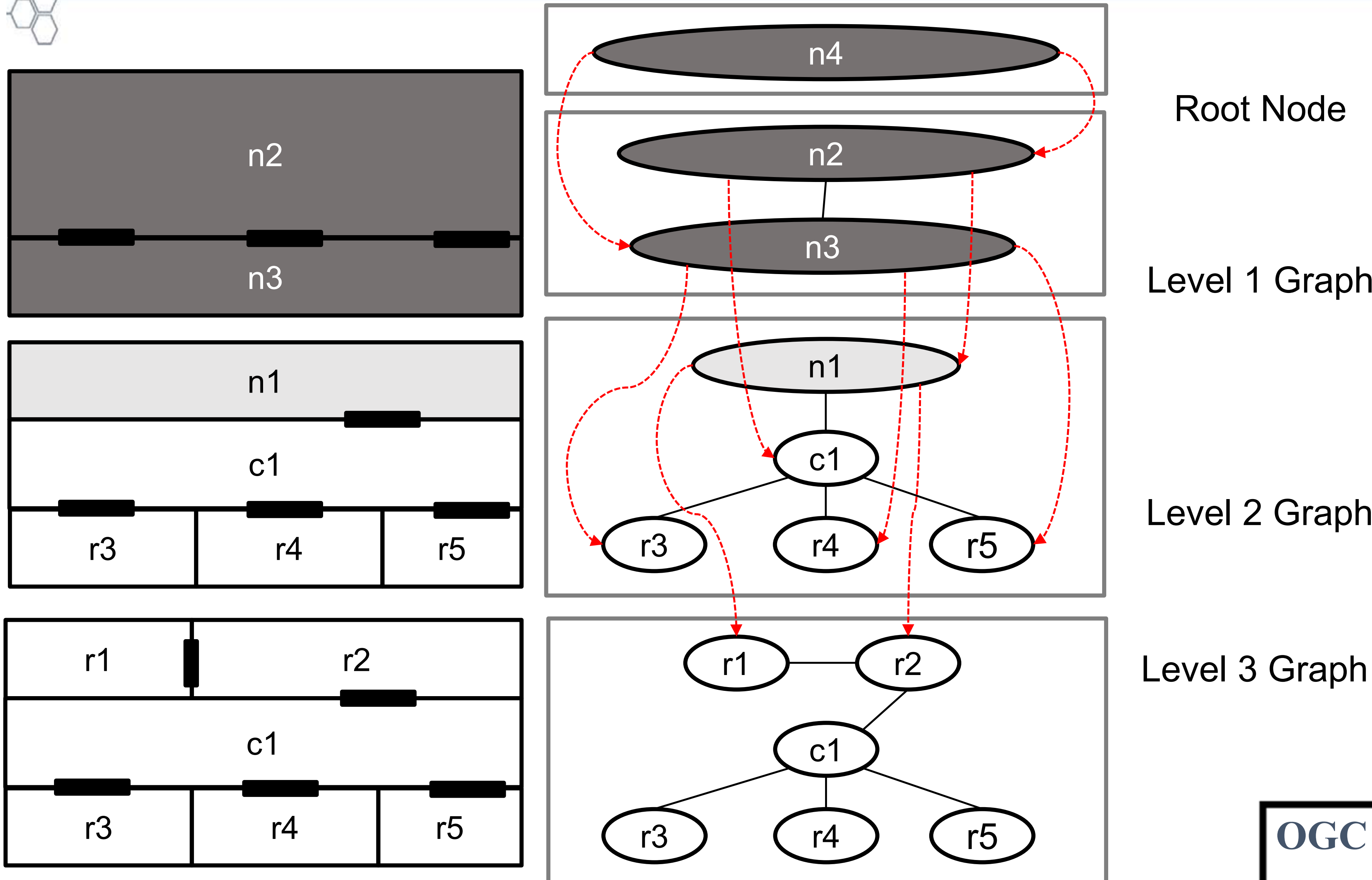
Cellular Space Model – Data Model



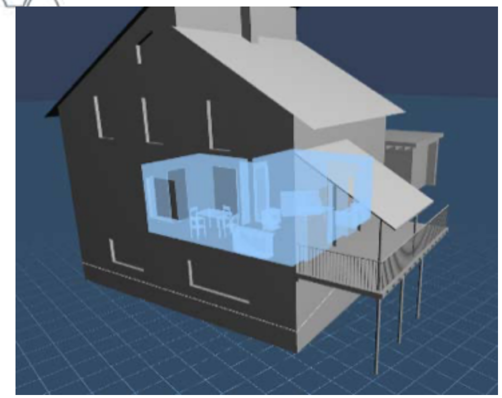
OGC®

OGC IndoorGML

Cellular Space Model – Hierarchical Subspacing

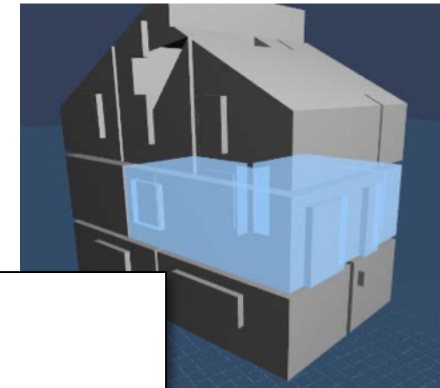


Cellular Space Model – Integration with Others



Room object in
CityGML data set
FJK-Haus-LoD4-V3.gml

Cell object in
IndoorGML



```
<bldg:interiorRoom>
    <bldg:Room
gml:id="GMLID_BUI198667">
    <gml:description>
        Wohnen / Essen
    </gml: description >
    <gml:name>102</gml:name>
    …
    </bldg:Room>
</bldg:interiorRoom>
```

```
<indoorcore:State gml:id="R15">
    <gml:name>102</gml:name>
    <duality gml:id="SR15">
        <gml:name>102</gml:name>
        <Geometry3D>
            <gml:Solid gml:id="sl15">
    </gml:Solid>
        </Geometry3D>
        <externalReference>
            <informationSystem>
              FJK-Haus-LoD4-V3.gml
            </informationSystem>
            <externalObject>

    <name>GMLID_BUI198667</name>
            </externalObject>
        </externalReference>
    …
    </indoorcore:State>
```

2. Making IndoorGML Data

Tools for Making IndoorGML Data

- InEditor
- InFactory
- InViewer

InEditor – 2D IndoorGML Editing Tool

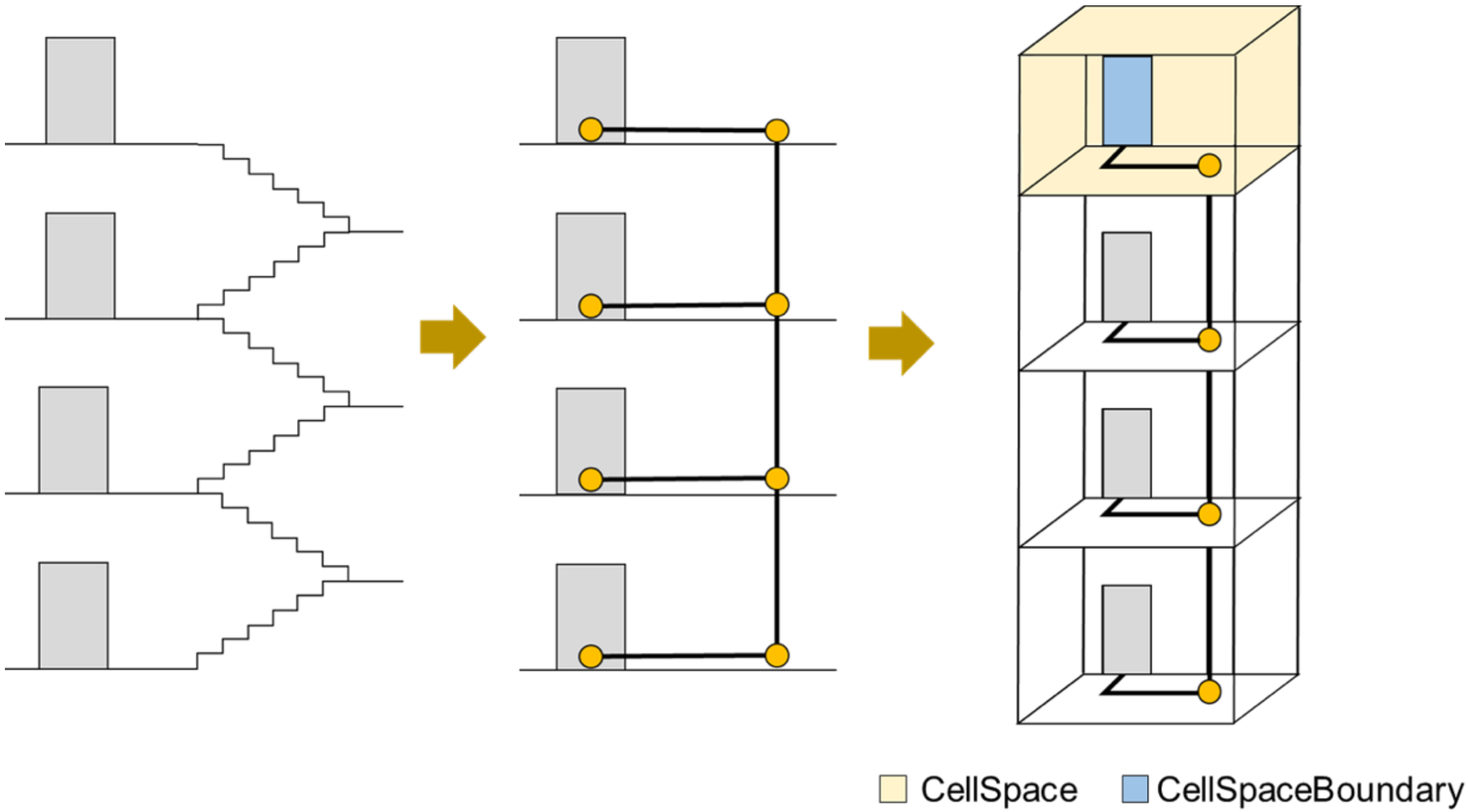
- An Open Source Editor for making IndoorGML
  - Available at https://github.com/STEMLab/InEditor (MIT License)
  - Operate with InFactor (RESTFul server for IndoorGML construction)

- Edit IndoorGML data
  - From 2D Image (Multiple Levels if necessary) and
  - Extrude it to 3D

- Installation of InEditor –
  - https://github.com/STEMLab/InEditor/wiki/2-1.-Enviroment-Setting
  - Install nodejs, maven, and JDK (over 1.8)
  - Install InFactory and InEditor
  - Install InViewer (Web version) or InViewer-Desktop (Unity3D version)

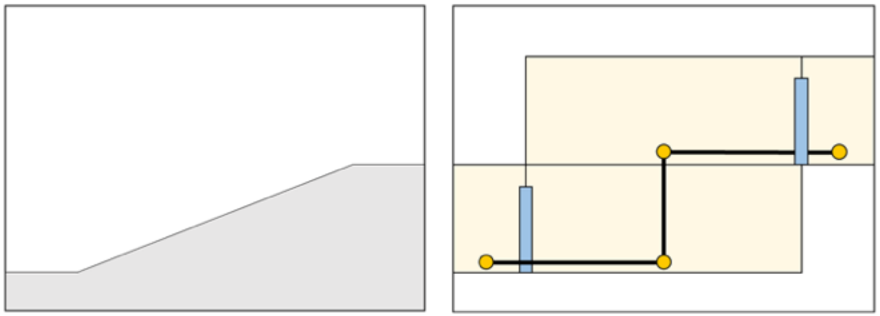
InEditor – Demonstration

Practical tips – Stair case in separate cells

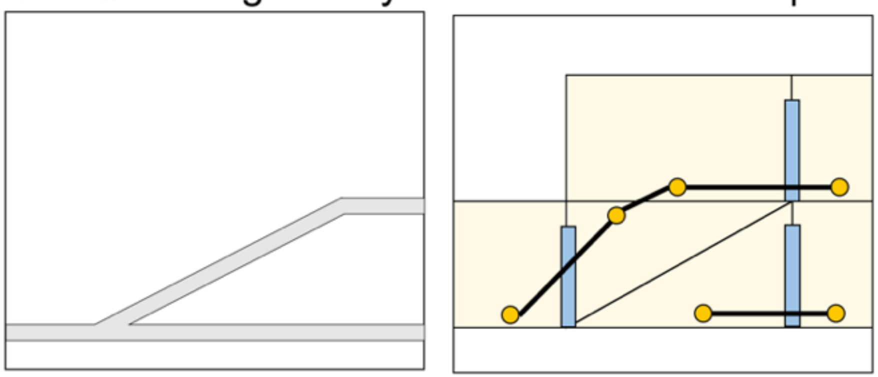


Practical tips – Stairs (complicated cases)

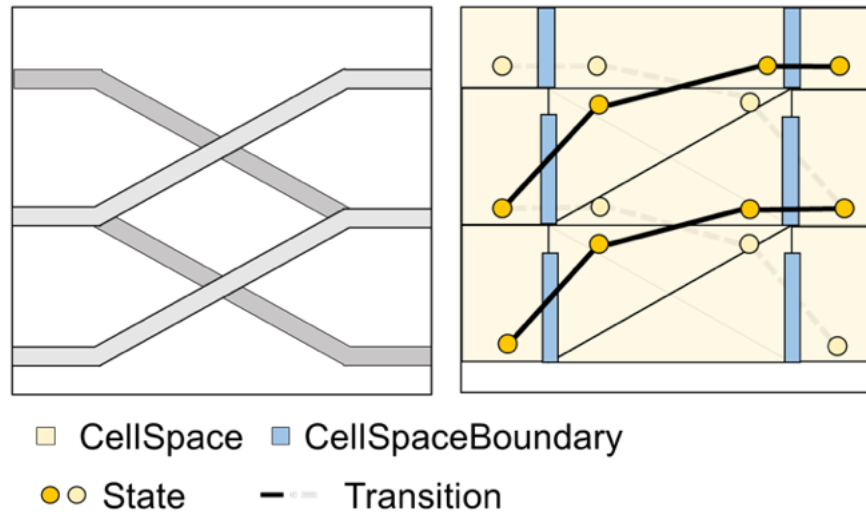
Stair#1 A single-story stair



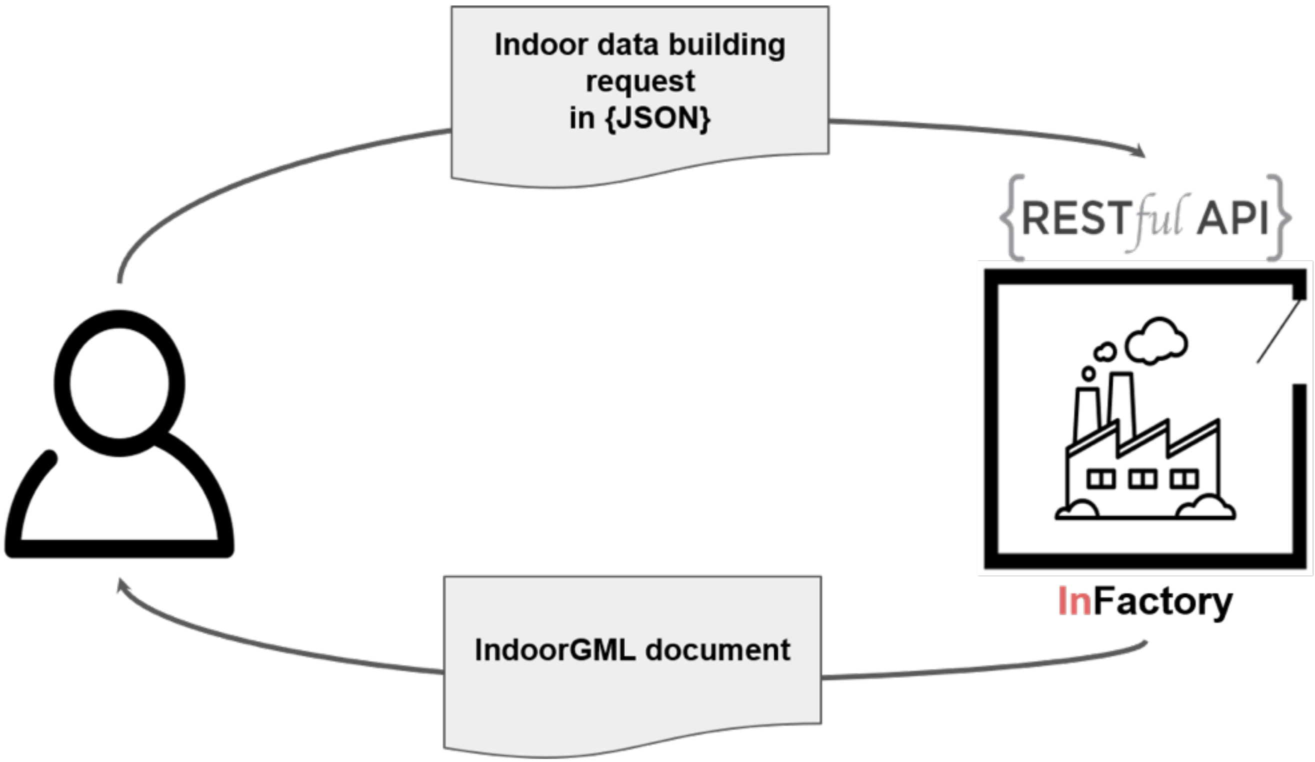
Stair#2 Single-story stair with available space



Stair#3 A continuous stair



InFactory – a RESTful server for making IndoorGML



Open Source Software available at https://github.com/stemlab/infactory

InFactory - Basics

- Creating requests to the InFactory server.
- Steps;
  - Step 1: Creating IndoorGML document (POST request)
  - Step 2: Creating IndoorGML Features (POST request)

```
IndoorFeatures
                |-> PrimalSpaceFeatures
                                        |-> CellSpace
                                        |-> CellSpaceBoundary
                |-> MultiLayeredGraph -> SpaceLayers -> SpaceLayer
                                                                   |-> Nodes -> State
                                                                   |-> Edges -> Transition
```

  - Step 3: Downloading IndoorGML document (GET request)

InFactory – Example: Creating a document

- Sample Javascript code

```
<!DOCTYPE html>
<html>
<body>

<h2>The XMLHttpRequest Object</h2>

<button type="button" onclick="loadDoc()">Request data</button>

<p id="demo"></p>

<script>
function loadDoc() {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 201) {
      document.getElementById("demo").innerHTML = this.responseText;
    }
  };
  var url = 'http://localhost:9797/documents/doc1';
  var contentType = 'application/json';
 var data = {
    "docId":"doc1"
}
  xhttp.open("POST", url, true);
  xhttp.setRequestHeader("Content-type", contentType);
  xhttp.send(JSON.stringify(data));
}
</script>

</body>
</html>
```

InFactory – Example: Creating a cell space

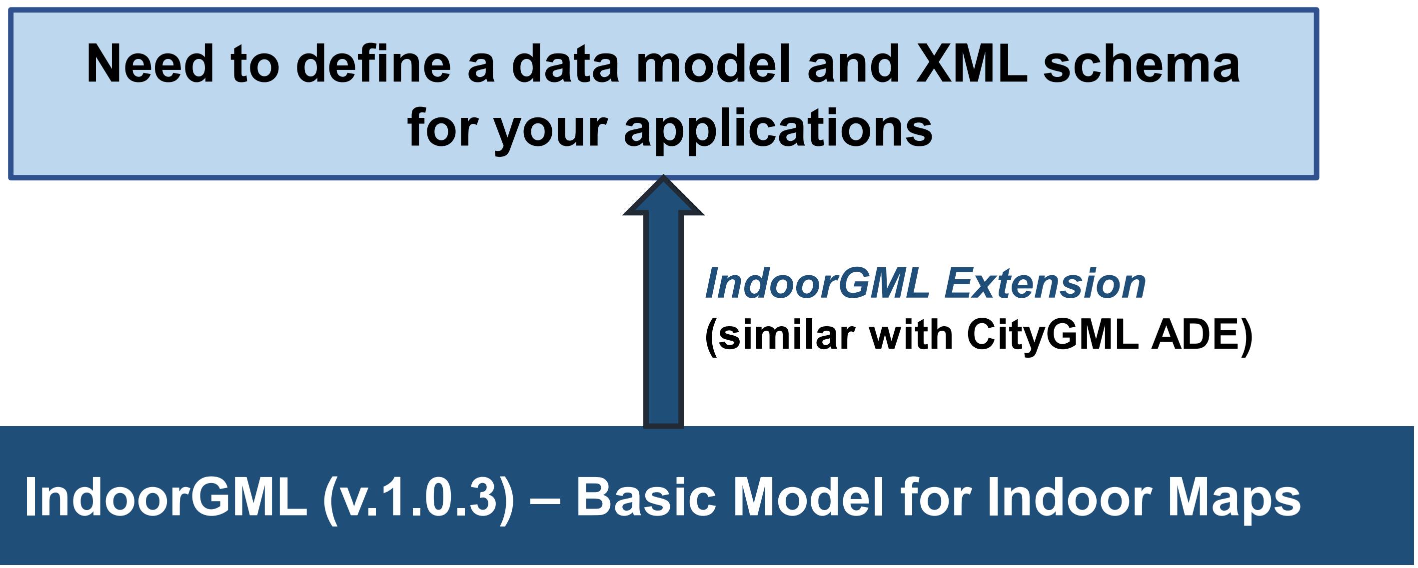
- More sample codes are found at https://github.com/STEMLab/InFactory/wiki/RESTful-API

Conversion from other data sets

- Several approaches are possible by converting other formats to IndoorGML
  - BIM, CityGML
  - Indoor OpenStreetMap, IMDF
  - CAD
  - Point cloud
  - Blueprint, Paper maps
- ETL tools support some of the conversions.
  - FME
  - Hale
- Some open source tools are available at http://github.com/stemlab

3. Extending IndoorGML for your applications

Extending IndoorGML



Basic ways to extend IndoorGML

Define your own entity types by

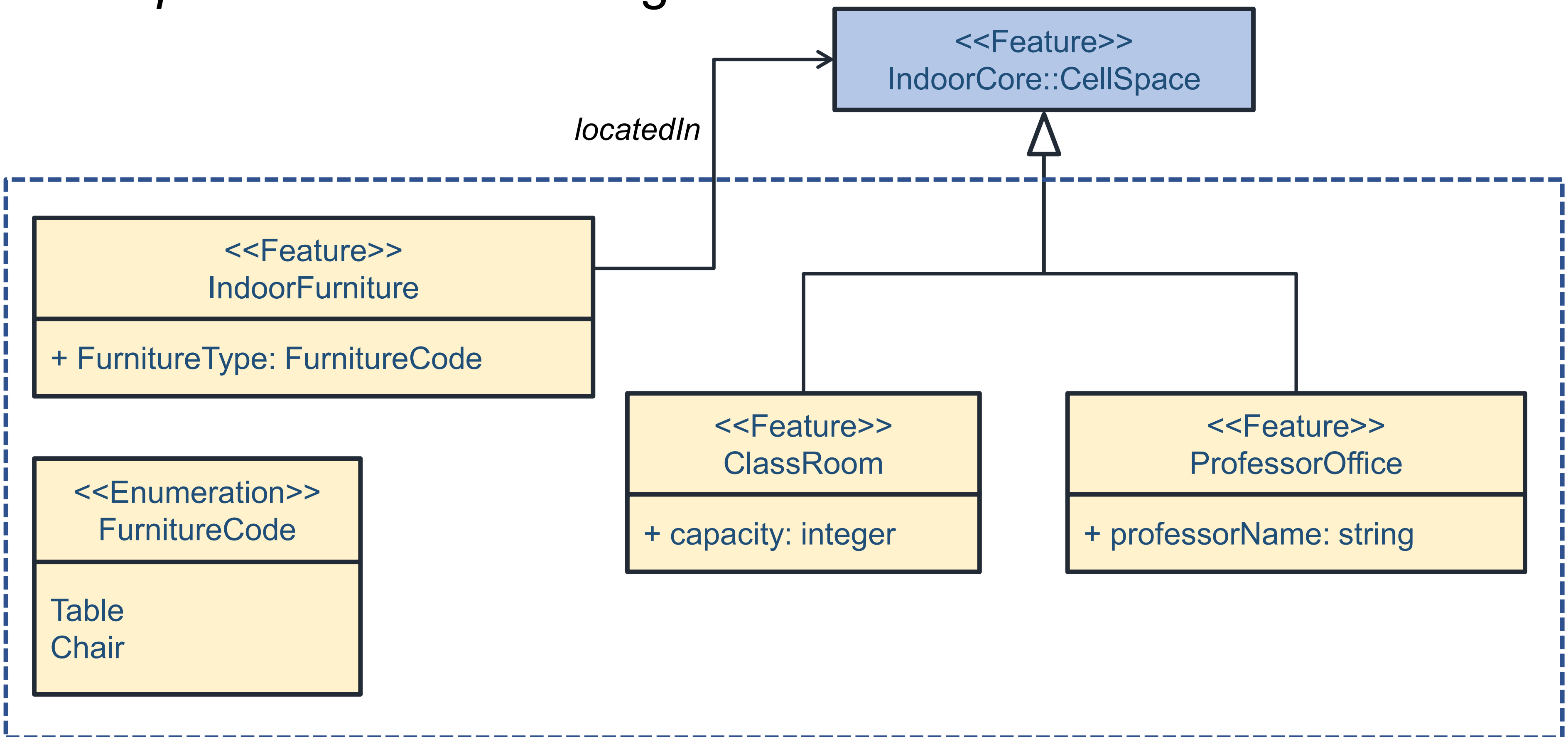
Inheritance from classes in IndoorGML, particularly from
- CellSpace,
- State,
- SpaceLayer

Aggregation of entity types defined in IndoorGML

Association with entity types in IndoorGML

Basic ways to extend IndoorGML

Example – UML Class Diagram



Basic ways to extend IndoorGML

Example – XML Schema

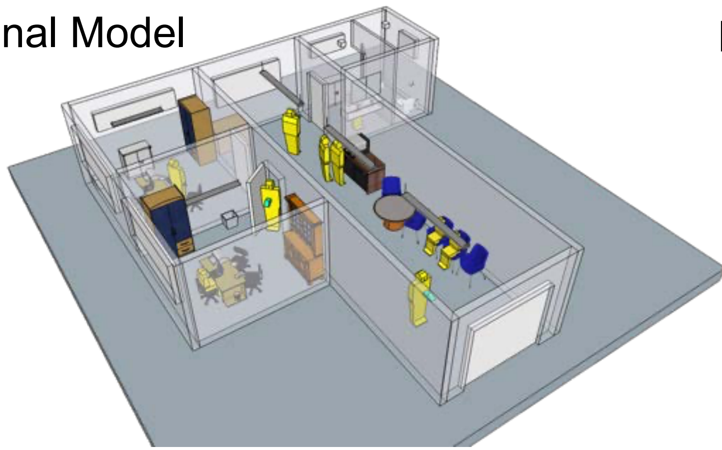
```
<xs:element name=“ClassRoom" type=“ClassRoomType" substitutionGroup=“core:CellSpace">
  <xs:annotation>
    <xs:documentation>Class Room as a subclass of CellSpace in IndoorGML
core</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:complexType name=“ClassRoomType”>
  <xs:complexContent>
    <xs:extension base=“core:CellSpaceType">
      <xs:sequence>
        <xs:element name=“capacity” type=“integer” minOccurs=“1” maxOccurs=”1” />
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>

<xs:element name=“IndoorFurniture” type=“IndoorFurnitureType”>
 …
```

No overlapping with other cells

Within one single space layer, NO overlapping is allowed

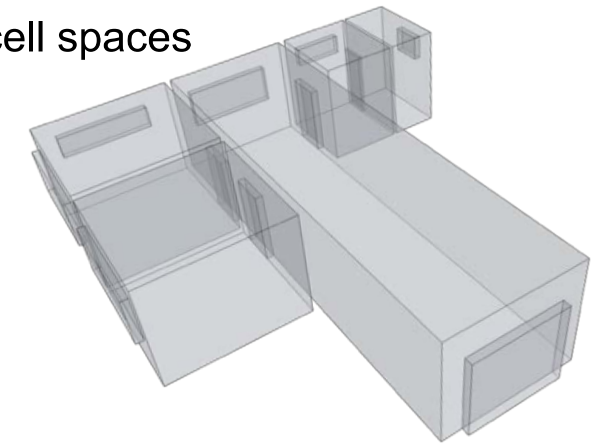
Example – Basic cell spaces and feature space

Original Model

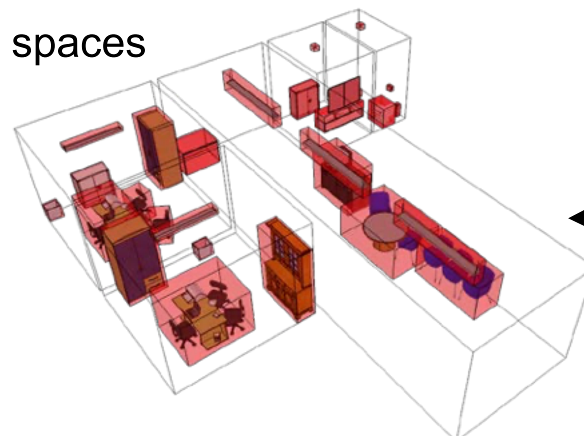


Source: Abdoulaye A. Diakite, 2019, OGC Toulouse
https://portal.ogc.org/files/?artifact_id=91370

Basic cell spaces

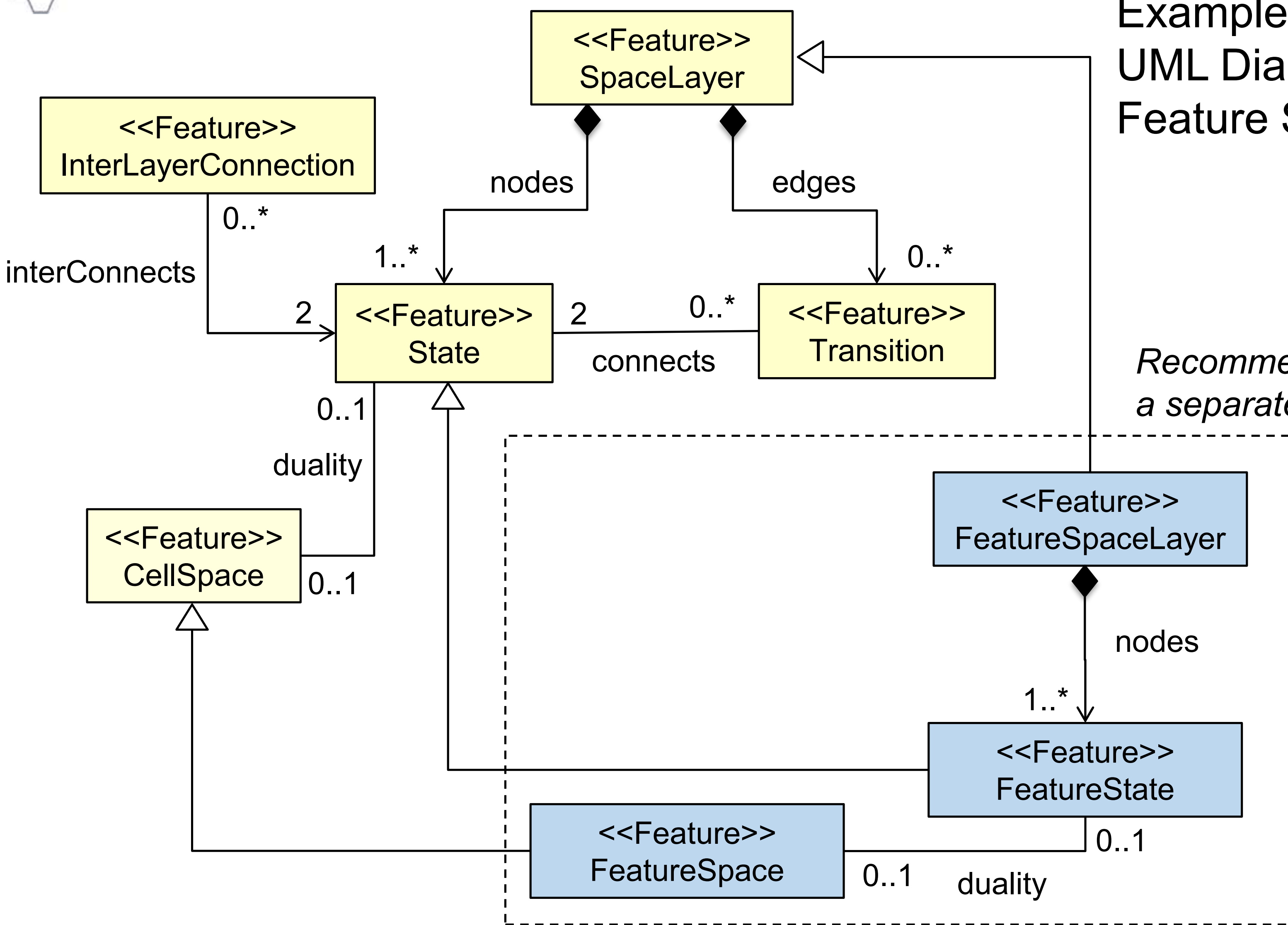


Feature spaces



Should be separate space layers

Separate space layer if necessary



Example –
UML Diagram for
Feature Space Layer

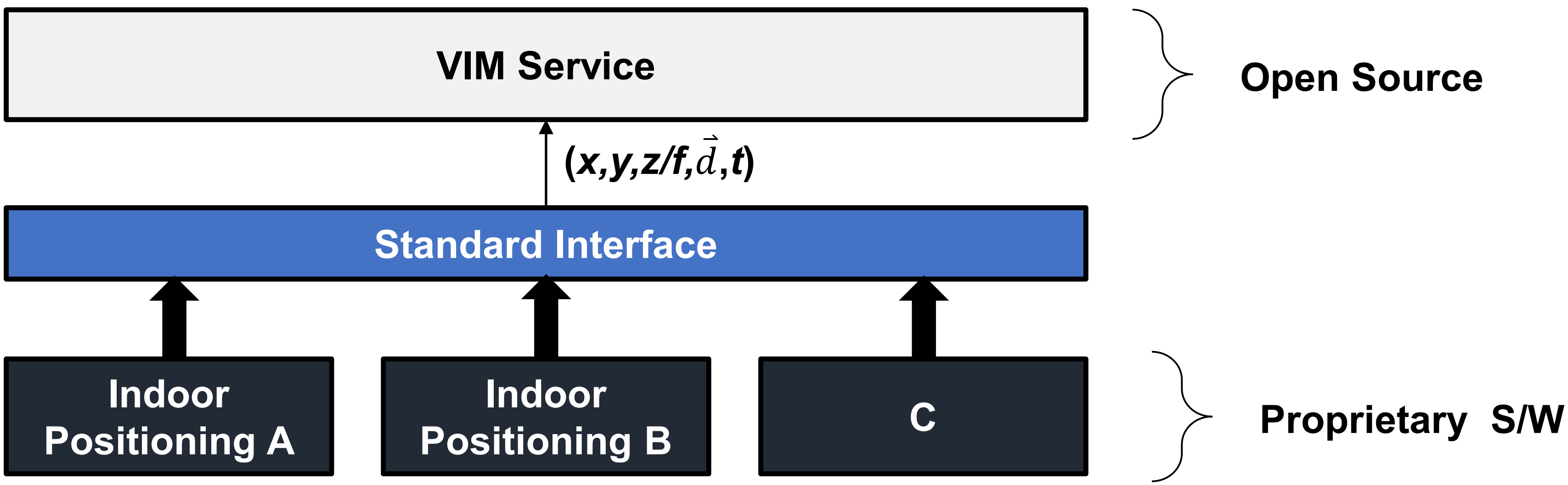
Recommend to define a separate space layer

OGC IndoorGML

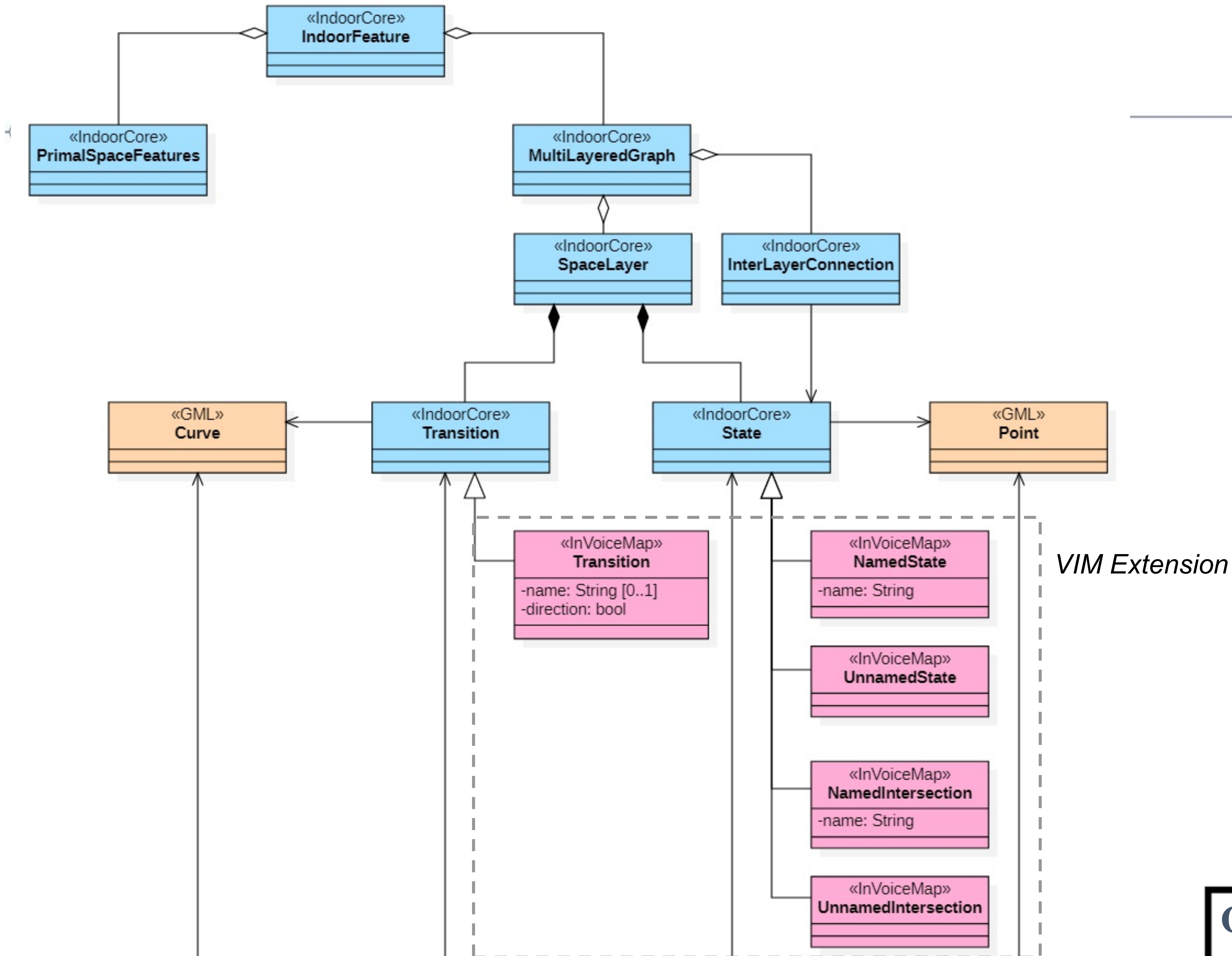
4. Use-Cases

VIM – Voice Indoor Maps for Visually Impaired

- Smartphone app.



OGC IndoorGML

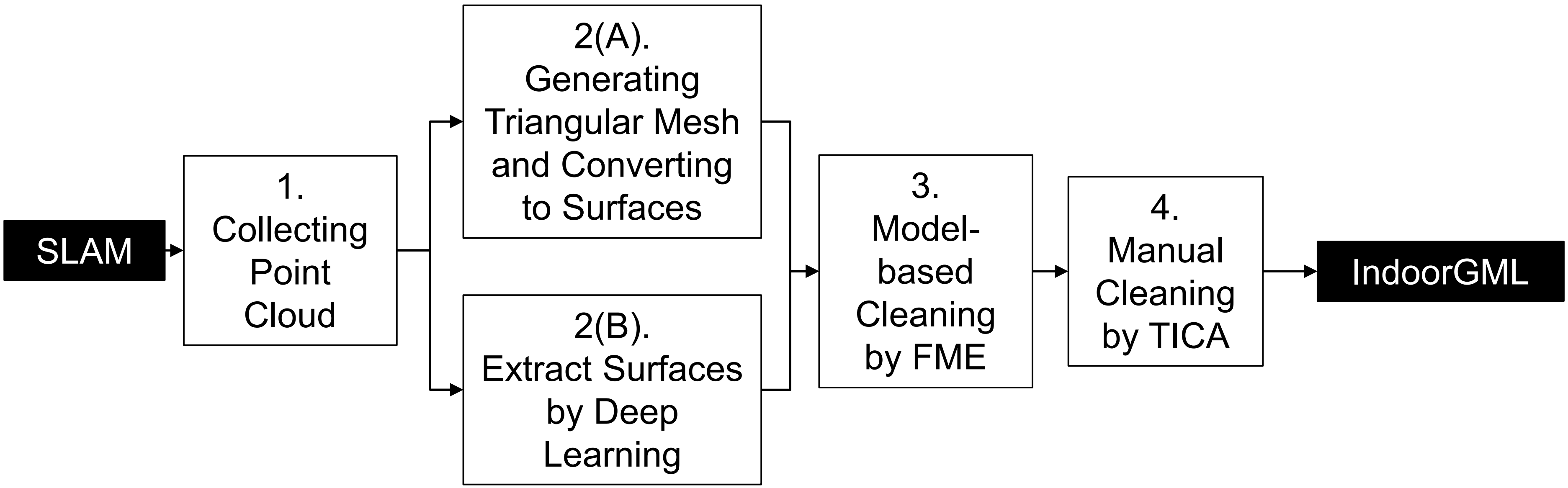

«IndoorCore»
IndoorFeature
«IndoorCore»
PrimalSpaceFeatures
«IndoorCore»
MultiLayeredGraph
«IndoorCore»
SpaceLayer
«IndoorCore»
InterLayerConnection
«GML»
Curve
«IndoorCore»
Transition
«IndoorCore»
State
«GML»
Point
«InVoiceMap»
Transition
-name: String [0..1]
-direction: bool
«InVoiceMap»
NamedState
-name: String
«InVoiceMap»
UnnamedState
«InVoiceMap»
NamedIntersection
-name: String
«InVoiceMap»
UnnamedIntersection
VIM Extension

A Haptic User-Interface of Indoor Maps for VI

Youtube: https://youtu.be/XI6POKEFXCo

From Point Cloud to IndoorGML

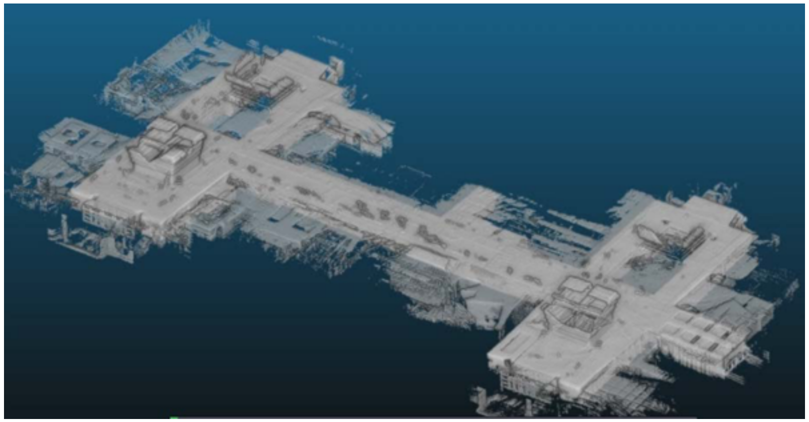
NIST-OGC Indoor Mapping and Navigation Pilot Project
Goal: Construct IndoorGML data from Point Cloud

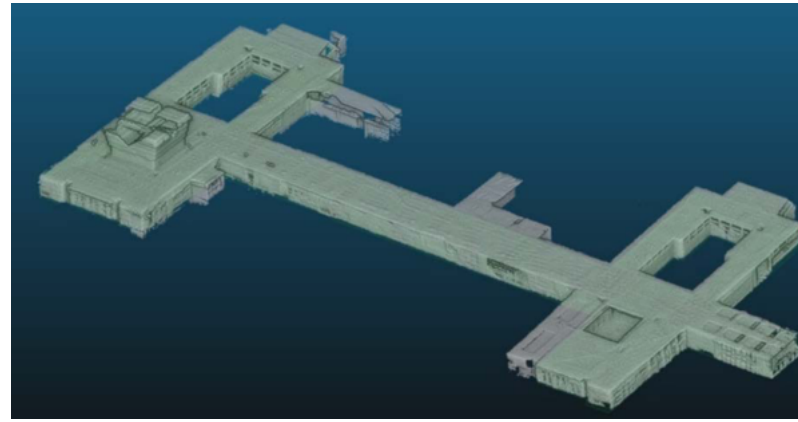


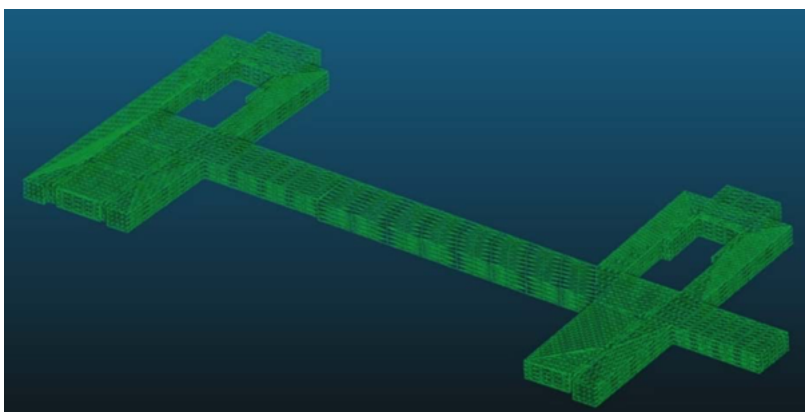
Youtube https://www.youtube.com/watch?v=U4e5UJVwRBQ&t=3s

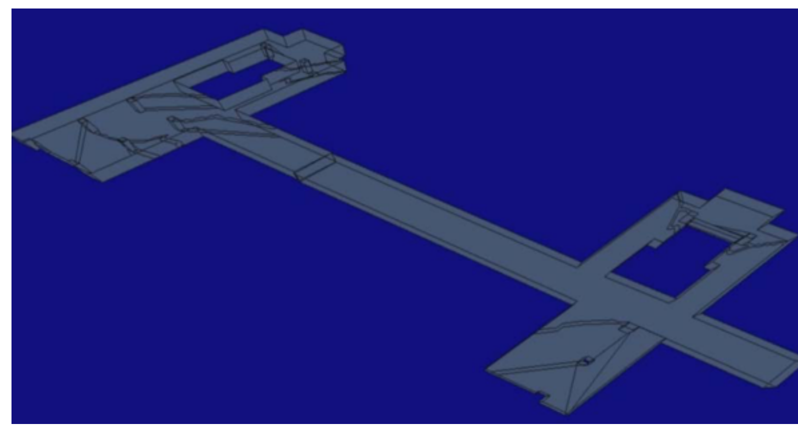
OGC IndoorGML

From Point Cloud to IndoorGML

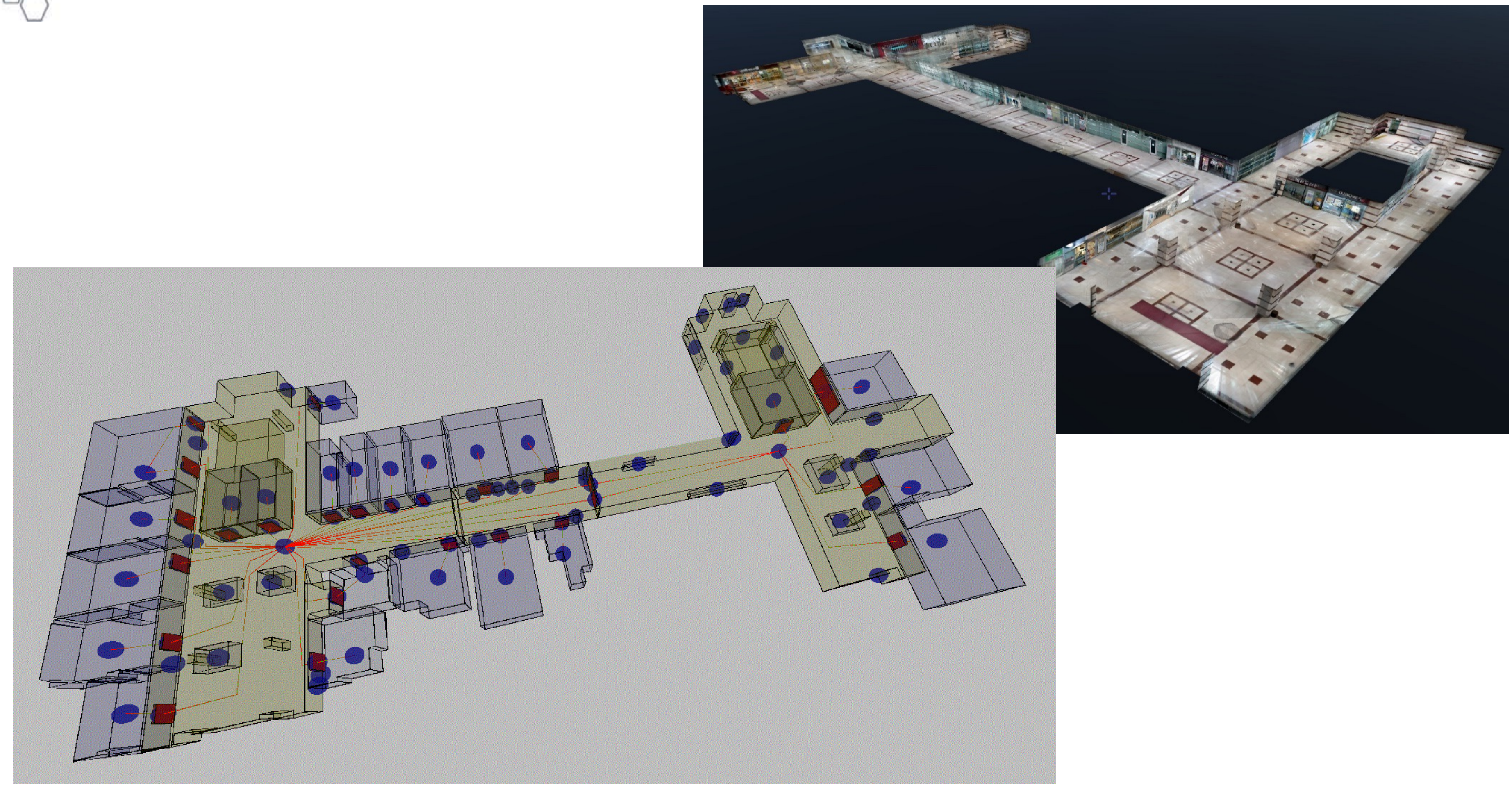








From Point Cloud to IndoorGML – Results



More information on NIST-OGC Indoor Mapping and Navigation Pilot Project
http://docs.opengeospatial.org/per/18-089.html

OGC IndoorGML

5. Summary

Summary

- IndoorGML is
  - Standard data model and format for indoor maps
  - Base standard model
  - Space model
  - Semantic model
  - To support indoor map applications (navigation, analysis,etc.)

- IndoorGML is NOT
  - Only for visualization
  - Only for navigation (although the initial goal was the navigation)
  - Feature model