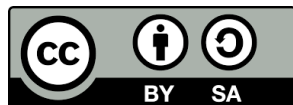




BIRGIT

# Estándares de datos 3D

Ariana.kubart@ocellus.se



## Resultados de aprendizaje

Al final de esta clase, se espera que el alumno sea capaz de:

- Nombrar varias formas en que se pueden almacenar los datos 3D, con codificaciones CityGML enfocadas
- Resumir las partes principales del modelo conceptual CityGML y cómo se pueden utilizar
- Comprender los aspectos de CityGML que son importantes para la conversión hacia y desde BIM

## Modelos semánticos de ciudades en 3D

- Proporcionar información geográfica, topográfica y semántica sobre los objetos
- Interacciones entre objetos
- Descomposición jerárquica en partes más pequeñas (por ejemplo, edificio, pared, ventana)
- Puede ser complejo y cubrir grandes áreas

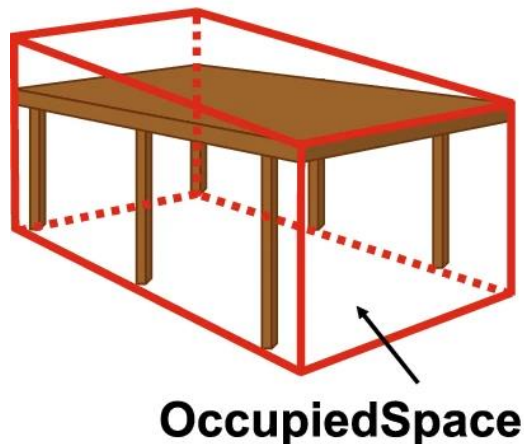


Se puede agregar mucha información semántica a un modelo 3D  
Modelo 3D de Nancy, Francia, Google Earth

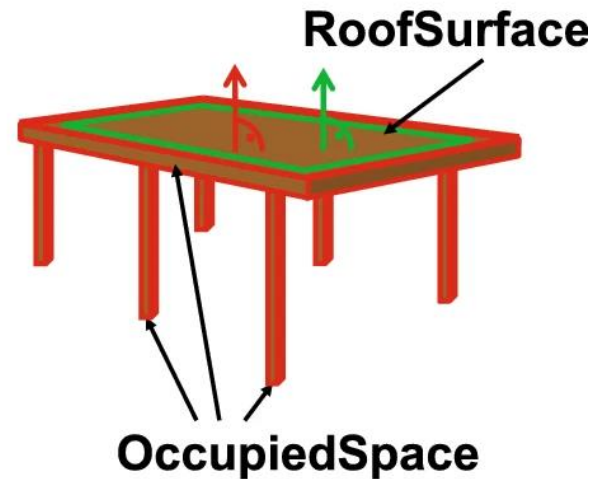
## Estándar CityGML

- CityGML es un modelo de datos abiertos del Open Geospatial Consortium (OGC)
- Destinado a representar modelos 3D semánticos
- La versión actual es la 3.0, aprobada en 2021

### Carport in LOD1



### Carport in LOD2/3

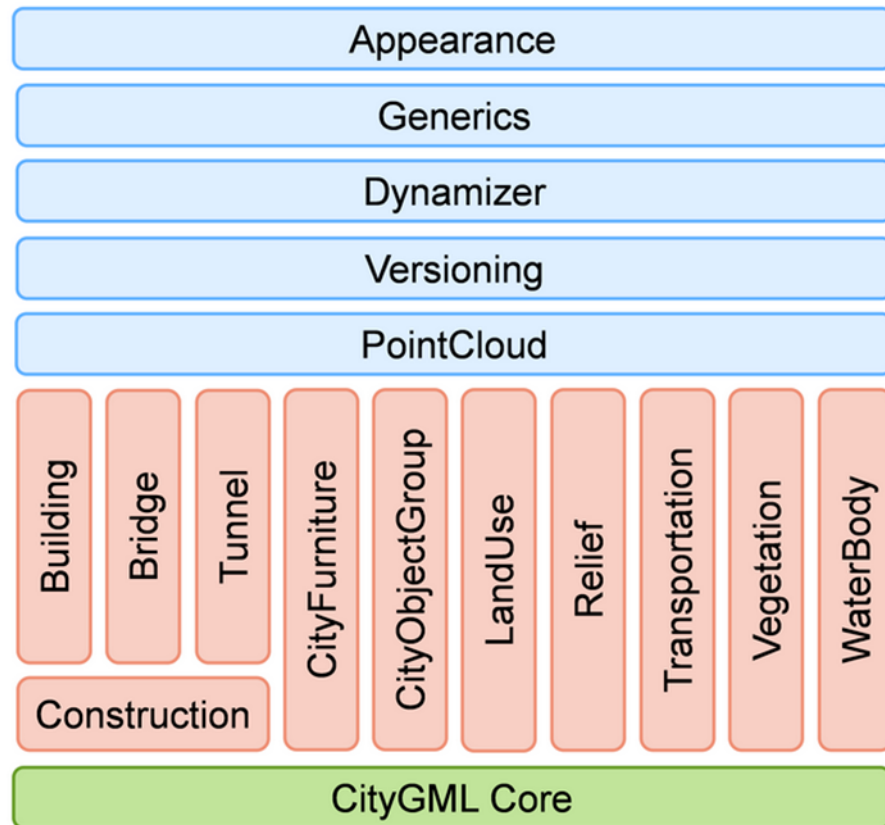


Representation of a carport as OccupiedSpace in different LODs. From: [CityGML 3.0: New Functions Open Up New Applications](#)



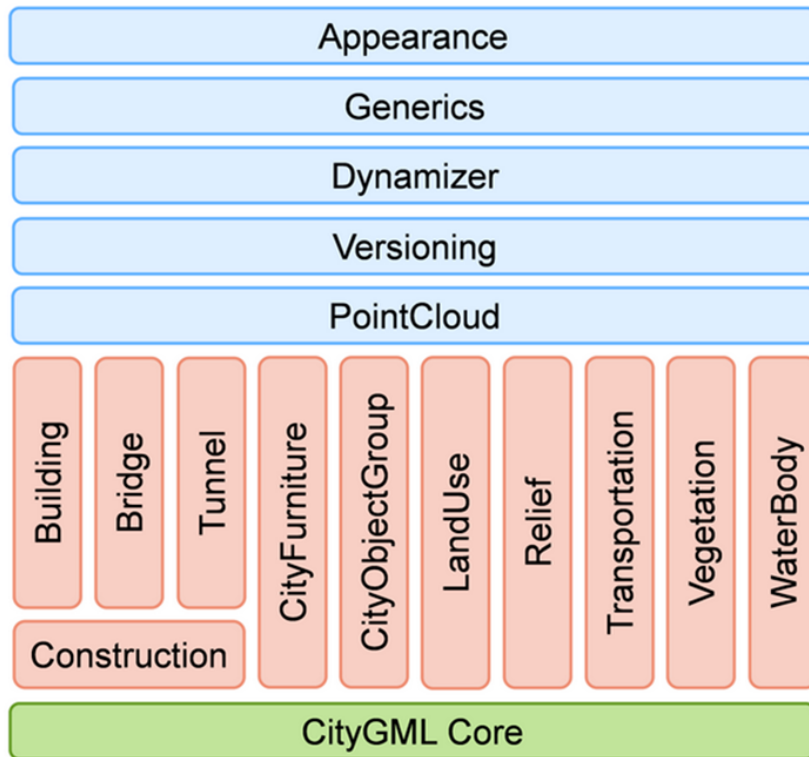
## Modelo Conceptual CityGML (CM)

- Los objetos comunes de la ciudad deben describirse de la misma manera en todos los modelos
- CityGML define las clases de los objetos más importantes, cómo descomponerlos y los atributos útiles
- Estas clases se definen en el modelo conceptual CityGML



## Módulos de clase en CM

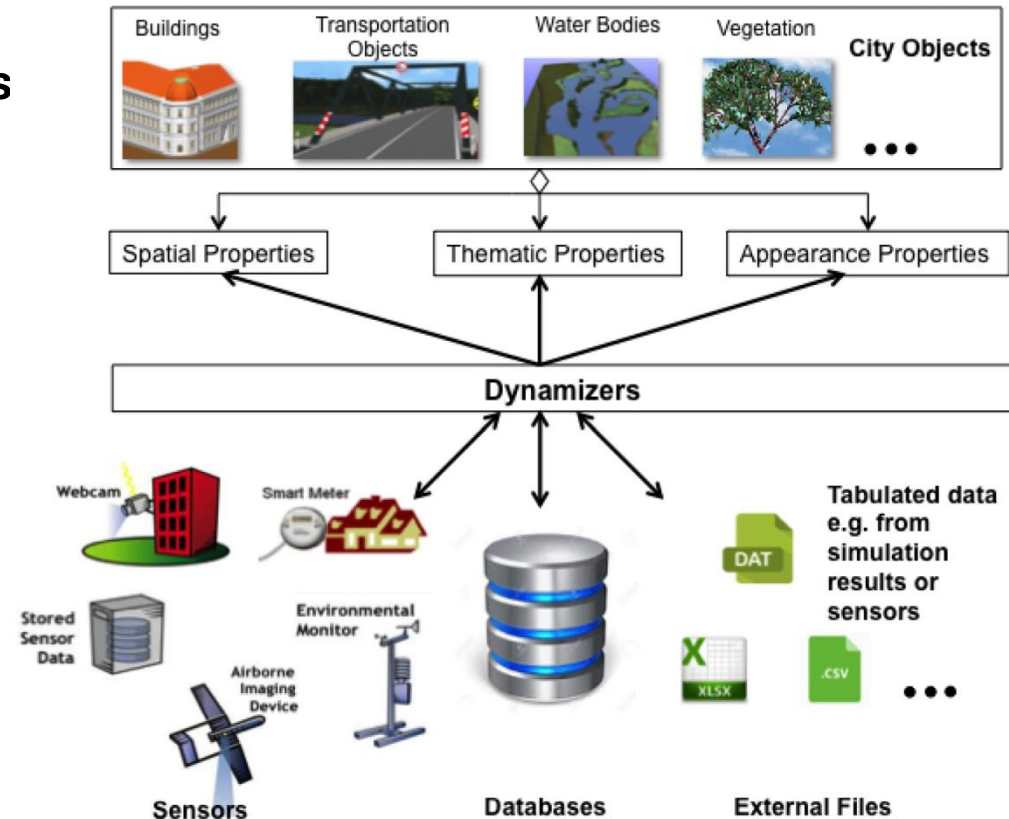
- Módulos de clase de extensión, mostrados verticalmente en rosa
- Los módulos son clases temáticas específicas de los objetos más importantes
- No es necesario incluir todos los módulos de extensión en un modelo 3D



CityGML Conceptual Model, [CityGML - Open Geospatial Consortium \(ogc.org\)](http://www.opengeospatial.org/standards/citygml)

## Módulos de aspectos específicos

- Los cinco módulos verticales de color azul en el CM
- Añade aspectos específicos de modelado, por ejemplo, texturas, colores, tiempo, versiones del modelo...
- Se puede utilizar junto con todos los módulos de extensión





## Información semántica en los modelos 3D

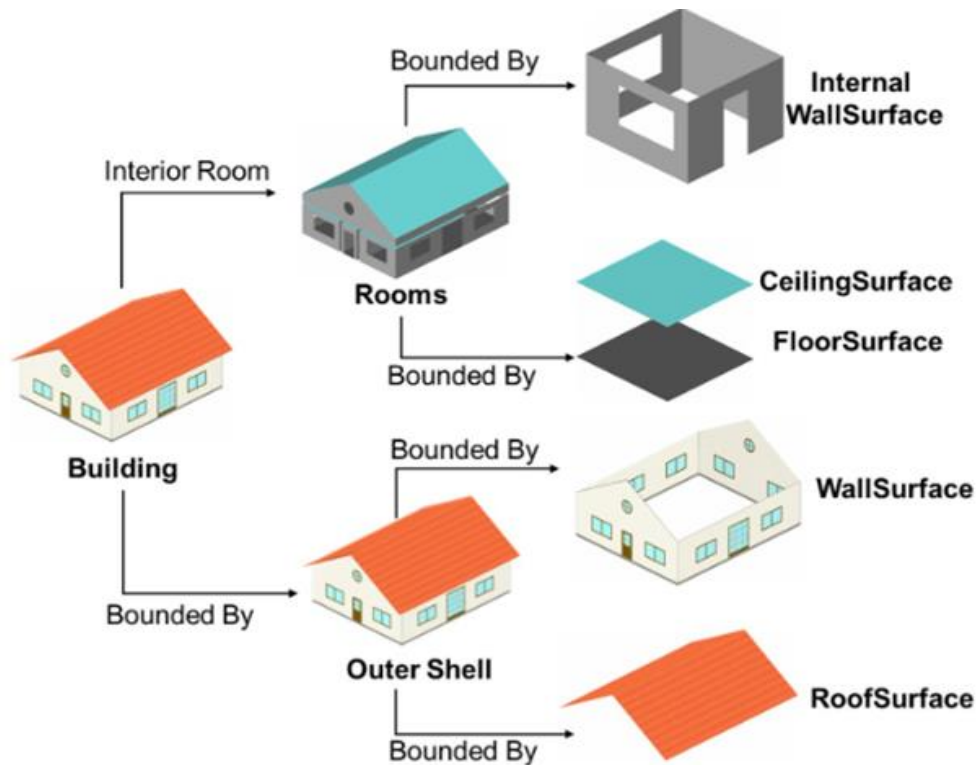
- Todos los objetos pertenecen a una clase o pueden definirse como objetos "genéricos"
- El objeto puede ser representado por semántica (= propiedades no espaciales), geometría 3D, topología 3D, apariencias y cambios a lo largo del tiempo
- FeatureID único y obligatorio para cada objeto



*Ejemplo de la Universidad Técnica de Delft de cinco tipos de techos: planos, a dos aguas, a cuatro aguas, piramidales y cobertizos.  
GitHub - tudelft3d/3dbook: Libro para el curso GEO1004: Modelado 3D del entorno construido*

## Esquema de agregación

- Puede haber interrelaciones jerárquicas entre los objetos de la ciudad
- Jerarquía de agregación: todo el edificio y su descomposición en partes (ver figura)
- Útil para consultas, simulaciones y análisis

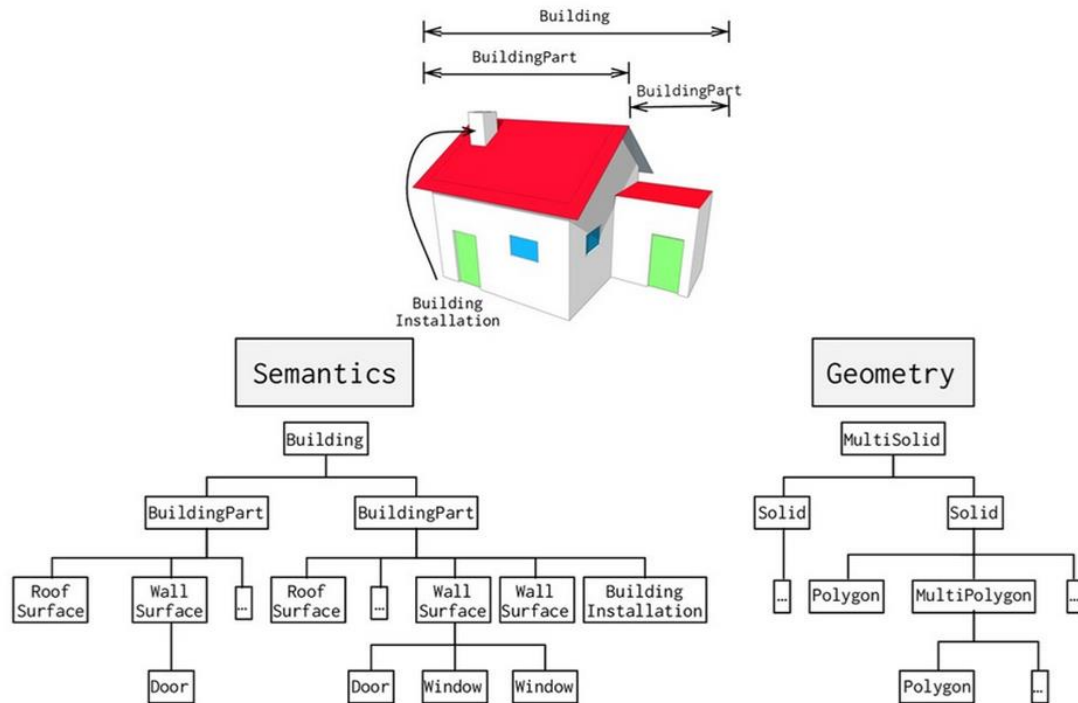


Vista general de un edificio representado en formato CityGLM. De Malhotra et al (2021): Herramienta de código abierto para transformar los niveles de detalle de CityGML

## Semántica coherente -Modelado geométrico

- El edificio y sus partes = jerarquía de agregación semántica
- También existe una jerarquía geométrica: ubicación, forma, extensión...
- Es crucial que la semántica y la geometría de los objetos correspondientes estén vinculadas entre sí

## Spatio-semantic coherence



## Nivel de detalle, LoD I

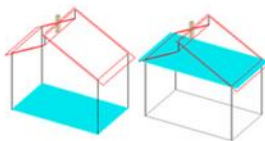
LOD0 – Modelo altamente generalizado

LOD1 – Modelo de bloques / objetos de extrusión

LOD2 – Modelo realista, pero aún generalizado

LOD3 – Modelo muy detallado

LOD0



LOD1



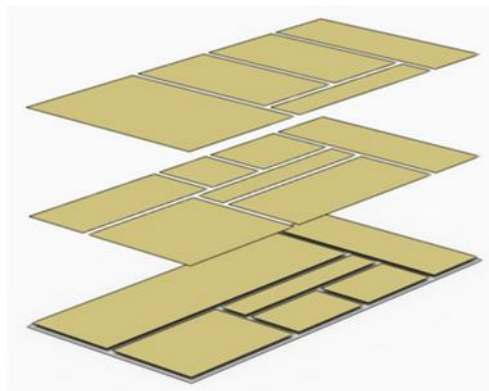
LOD2



LOD3



Representation of the same real-world building in the Levels of Detail 0-3



Floor-plan representation in LoD0 (left) and in LoD2 (right). From: <https://www.gim-international.com/content/news/citygml-3-0-conceptual-model-approved-as-official-ogc-standard>

## Nivel de detalle, LoD II

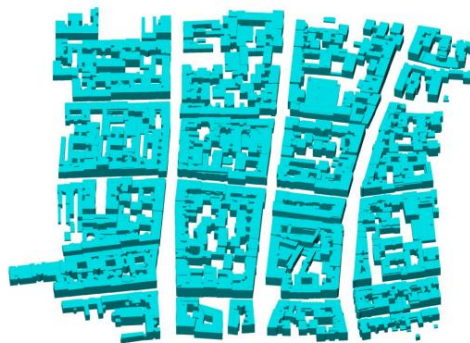
- Los LoD son aplicables tanto en interiores como en exteriores
- Edificio individual o barrios enteros
- posibilidad de combinar diferentes niveles de detalle en el mismo modelo



(a)



(b)



(c)



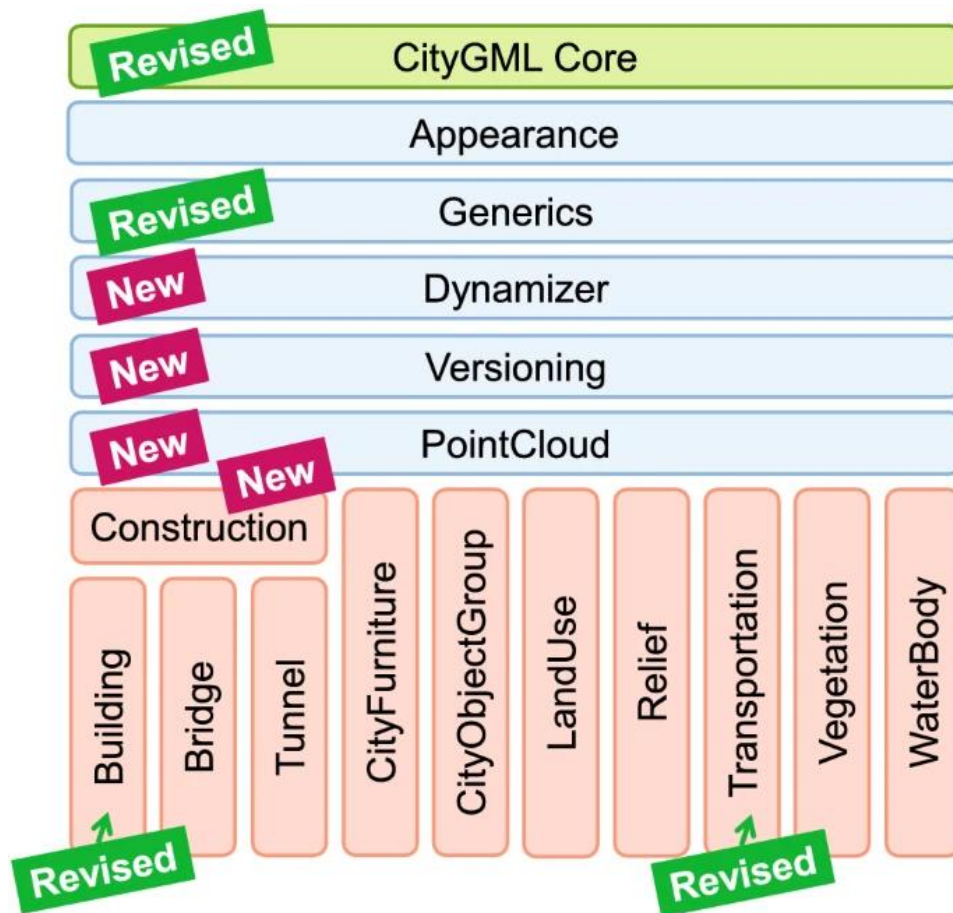
(d)

Visual representation of several housing-blocks in Vienna (a) underlying image from Google Maps, (b) representation of open LoD2 CityGML model, (c) LoD1 model transformed from LoD2, (d) LoD0 CityGML models. From Malhotra et al (2021)

## Versiones anteriores de CityGLM, 1.0 y 2.0

- CityGML 3.0 es una evolución de las versiones 1.0 y 2.0.
- 1.0 y 2.0 todavía se usan en muchos modelos, pero se pueden actualizar a 3.0
- La versión 3.0 tiene muchas capacidades nuevas

Partes nuevas y revisadas de la versión 3.0, en comparación con la 2.0. De: Kutzner et al (2020) [CityGML 3.0: New Functions Open Up New Applications](#) | PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science ([springer.com](#))





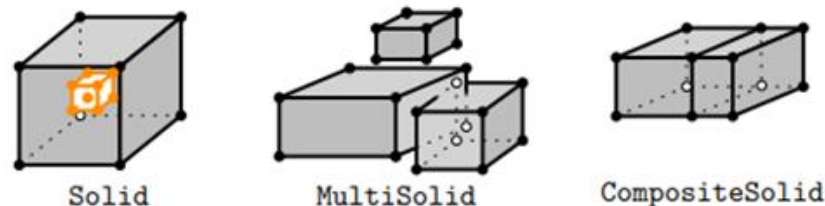
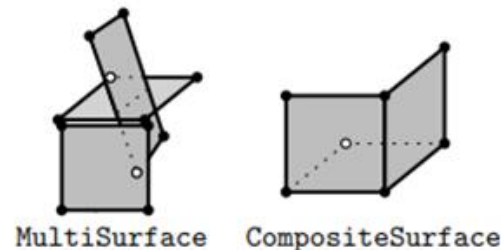
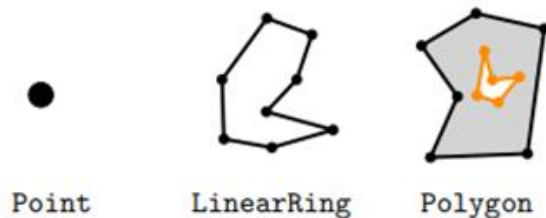
## Representación de geometría

Las propiedades espaciales de todos los objetos CityGML están representadas por clases de geometría definidas en la norma ISO 19107

Entre ellas se encuentran:

Geometrías primitivas: puntos, curvas, superficies y sólidos  
Diferentes tipos de geometrías agregadas

Todas las geometrías guardadas en el módulo Core



Algunas de las primitivas de CityGML, incluidos agregados y compuestos.  
De Ohori et al (2020-2022) [Releases · tudelft3d/3dbook \(github.com\)](https://github.com/tudelft3d/3dbook)

## Representación de topología

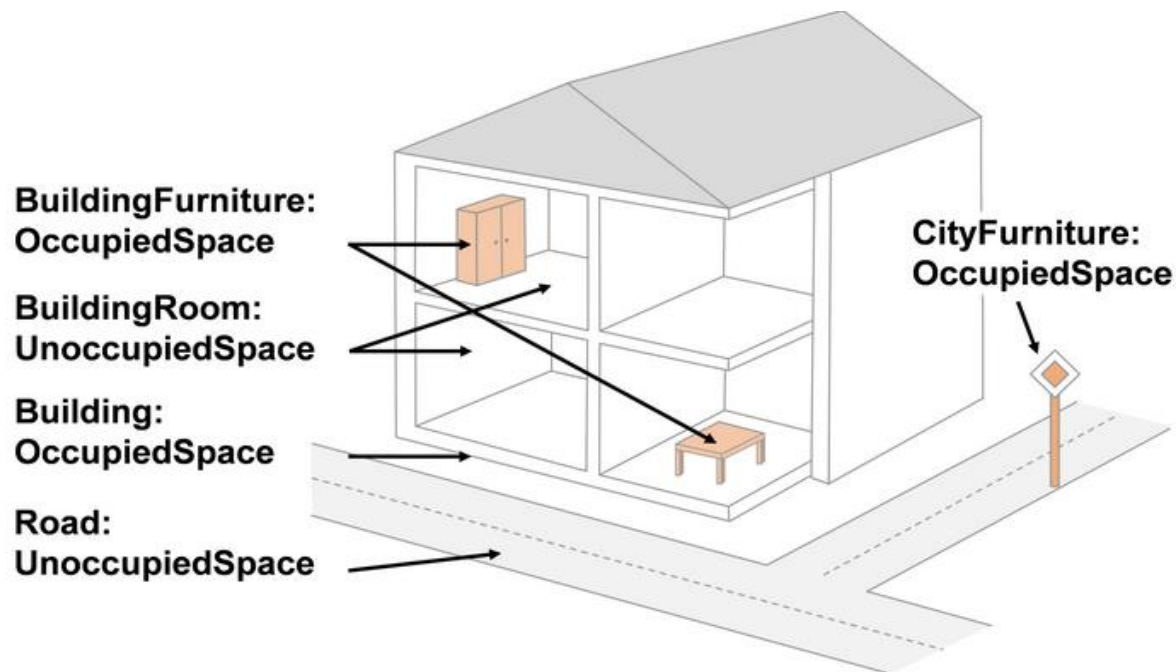


Esquema CityGML que muestra la relación entre la ventana y la habitación y el edificio. De:  
Salheb (2019) Conversión automática de CityGML a IFC

- La topología sigue la descomposición completa, similar a la geometría
- Relaciones entre elementos bien definidos en CityGLM 3.0
- Espacios: objetos del mundo real
- Límites espaciales: delimite y conecte los espacios (por ejemplo, superficie de la pared, superficie de la carretera...)

## Espacios Físicos y Lógicos

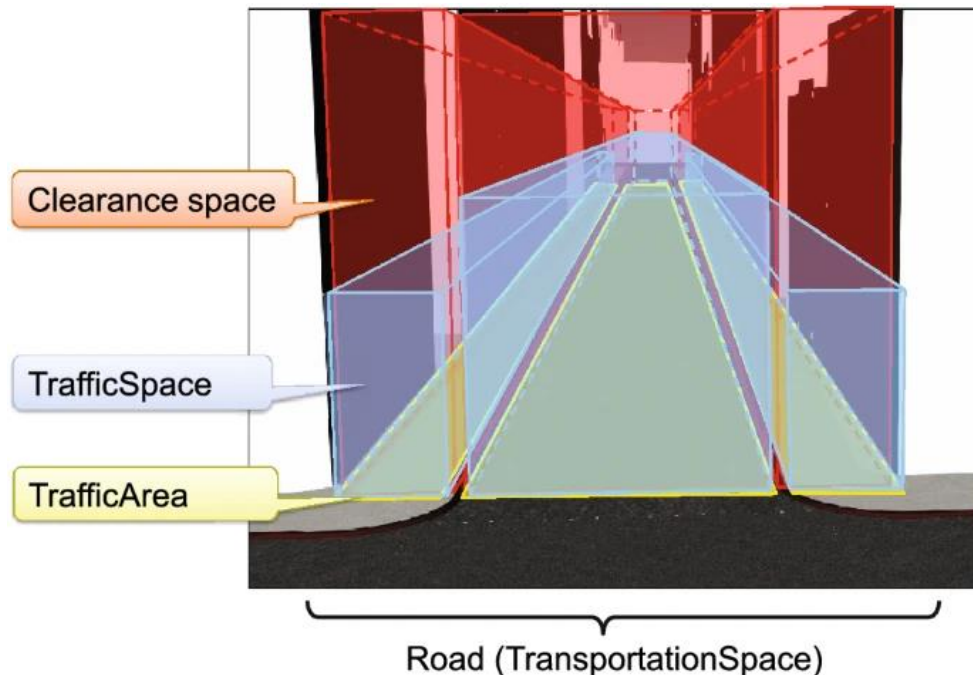
- Los espacios físicos están delimitados por objetos físicos, por ejemplo, edificios, árboles...
- Ocupado o desocupado (ver figura)
- Espacios lógicos: temáticos, p. ej., distrito de la ciudad, unidad de construcción



Espacios ocupados y desocupados. De: Kutzner et al (2020) CityGML 3.0: Las nuevas funciones abren nuevas aplicaciones

## Exteriores, Interiores y Jerarquías

- Cada edificio puede tener exterior, interior, subterráneo
- La clasificación y descomposición automáticas solo son posibles para las partes externas (visibles), no para las internas
- Las solicitudes requieren información más detallada
- Posibilidad de clasificación semiautomática para unidades más pequeñas



Descomposición del espacio de transporte; Es difícil conseguirlo de forma totalmente automatizada. De: Kutzner et al (2020) CityGML 3.0: Las nuevas funciones abren nuevas aplicaciones

## Coordenadas y elevación

Todas las geometrías de CityGML deben:

utilizar valores de coordenadas 3D

- estar absolutamente georreferenciado
- Pueden incluir modelos de terreno

2.5D: solo una coordenada Z para todos los edificios

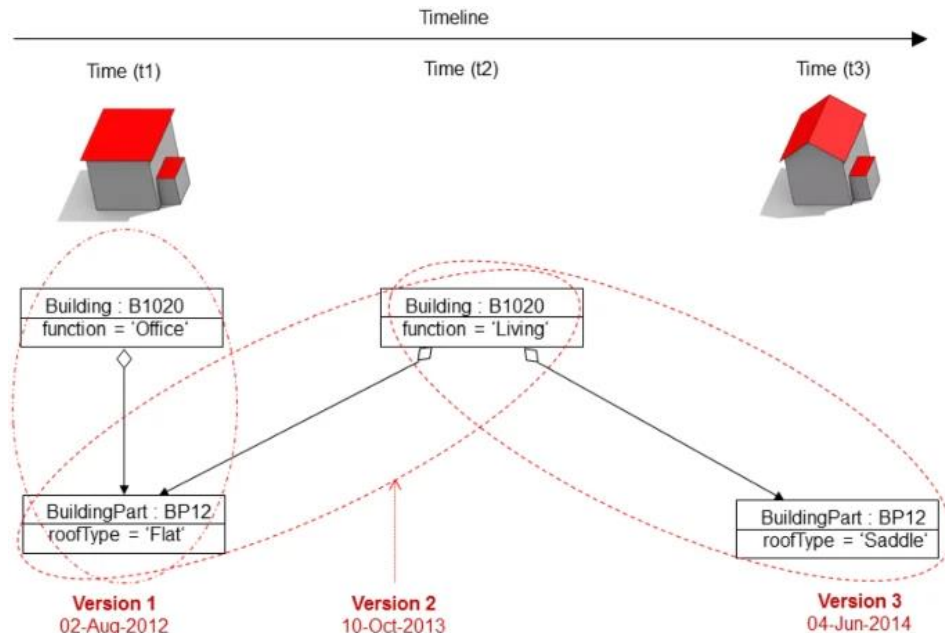


Captura de pantalla de ArcGIS Pro con los datos de entrenamiento de Esri

## 4D – Dimensión Temporal

- Importante en las ciudades inteligentes, los gemelos digitales
- Módulo de control de versiones: cambios lentos
- Módulo dinamizador: cambios rápidos, p. ej., datos de sensores

```
<cityObjectMember>
  <Building gml:id="B1020_t1">
    <identifier>B1020</identifier>
    <consistsOfBuildingPart>
      <BuildingPart xlink:href="//identifier[text()='BP12']"/>
    </consistsOfBuildingPart>
    <creationDate>2012-08-02</creationDate>
    <terminationDate>2013-10-10</terminationDate>
    <function>Office</function>
  </Building>
</cityObjectMember>
<cityObjectMember>
  <Building gml:id="B1020_t2">
    <identifier>B1020</identifier>
    <consistsOfBuildingPart>
      <BuildingPart xlink:href="//identifier[text()='BP12']"/>
    </consistsOfBuildingPart>
    <creationDate>2013-10-10</creationDate>
    <function>Living</function>
  </Building>
</cityObjectMember>
```



Ejemplo de versiones que representan modificaciones de un edificio (up)  
 Representación de diferentes versiones de objetos de ciudad dentro de un conjunto de datos CityGML codificado en GML (izquierda)  
 De: Kutzner et al (2020) CityGML 3.0: Las nuevas funciones abren nuevas aplicaciones

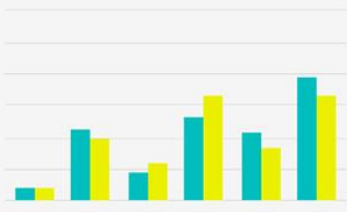


## Extensiones de dominio de aplicación, ADE

Data added using the CityGML ADE feature

LOD (-2)

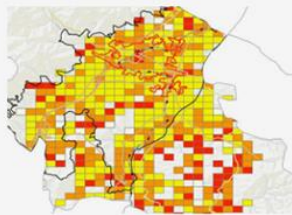
Graphs and charts



- National level, city level
- Indexing and projecting
- National government income and expenditures/economic activity

LOD (-1)

Heat maps



- Subregions/grids/administrative divisions
- Aggregate values for indices/areas
- Population/aging rates/road-to-area ratios, etc.
- Comparisons between cities/city analysis
- GIS/i-Urban Revitalization/e-Stat

LOD 0

Basic city planning maps



- Buildings/land (flat)
- Topographic models (height data)
- Building use/structure/area
- City plan studies/analysis
- GIS data

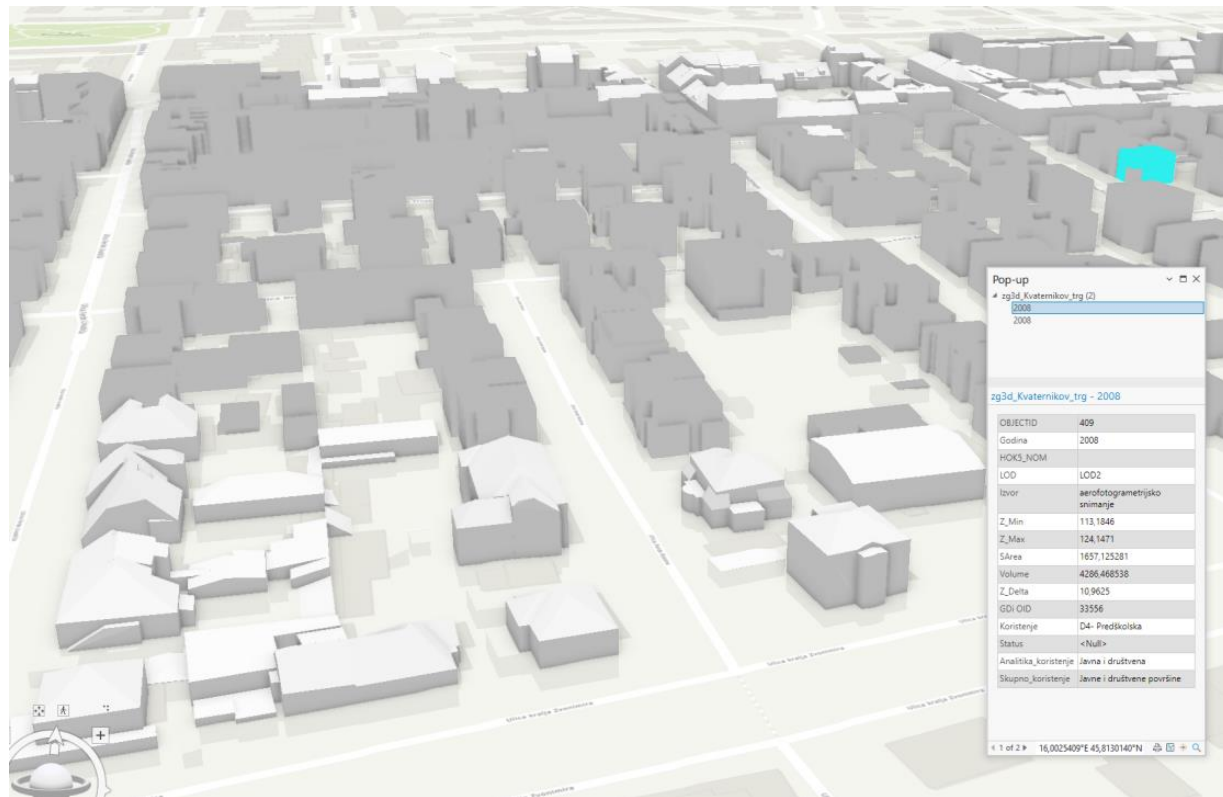
- Los ADE facilitan la adición de nuevas clases, atributos o relaciones
- por ejemplo, ADE de energía
- ADE de red de servicios

## Calidad de los datos

Los modelos de ciudad son grandes conjuntos de datos complejos

La calidad de los datos es un tema crucial

- Exactitud
- Integridad
- Usabilidad
- Consistencia
- Unicidad



Modelo semántico 3D de Zagreb, Croacia. Captura de pantalla de ArcGIS Pro.

## Codificaciones de CityGML

GML – Lenguaje de Marcado Geográfico

CityGML es el nombre de:

- Codificación GML basada en XML modelo de datos conceptual
- Emitido por Open Geospatial Consortium (OGC)

CityGML 3 permite codificar los datos en XML, en JSON o en esquemas de base de datos

Three encodings:

- XML-based →



- JSON-based →



- SQL-based →



## Codificación XML

### Codificación original de CityGLM

- verboso
- jerárquico
- complejo
- No adaptado para la web

Ya no se usa mucho

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <CityModel xmlns:xlink="http://www.w3.org/1999/xlink"
3   xmlns:gml="http://www.opengis.net/gml"
4   xmlns="http://www.opengis.net/citygml/2.0"
5   xmlns:bldg="http://www.opengis.net/citygml/building/2.0"
6   xsi:schemaLocation="http://www.opengis.net/citygml/2.0">
7   <cityObjectMember>
8     <bldg:Building gml:id="9a06451677c7">
9       <bldg:function>1070</bldg:function>
10      <bldg:lod1Solid>
11        <gml:Solid>
12          <gml:exterior>
13            <gml:CompositeSurface>
14              <gml:surfaceMember>
15                <gml:Polygon>
16                  <gml:exterior>
17                    <gml:LinearRing>
18                      <gml:pos>0.0 0.0 0.0</gml:pos>
19                      <gml:pos>0.0 1.0 0.0</gml:pos>
20                      <gml:pos>1.0 1.0 0.0</gml:pos>
21                      <gml:pos>1.0 0.0 0.0</gml:pos>
22                      <gml:pos>0.0 0.0 0.0</gml:pos>
23                    </gml:LinearRing>
24                  </gml:exterior>
25                </gml:Polygon>
26              </gml:surfaceMember>
27            ...
28          </bldg:Building>
29          <bldg:Building gml:id="jdhd76sa">
30            ...
31          </bldg:Building>
32        </cityObjectMember>
33      </CityModel>
```

## Codificación CityJSON II

- la alternativa más utilizada a la codificación XML JSON - JavaScript Object Notation
- incluso CityJSON es un estándar OGC
- Las coordenadas se almacenan en un solo lugar, en una matriz separada, es decir, los "vértices"

CityJSON permite la compresión completa y simplifica la estructura de archivos, en comparación con la codificación XML (figura derecha)  
Ejemplo de coordenadas en CityJSON (figura izquierda)

```
1  "vertices": [  
2    [23234, 111009, 1392],  
3    [29456, 115134, 1007],  
4    [54508, 229995, 1961],  
5    ...  
6    [23134, 625134, 203]  
7  ]
```

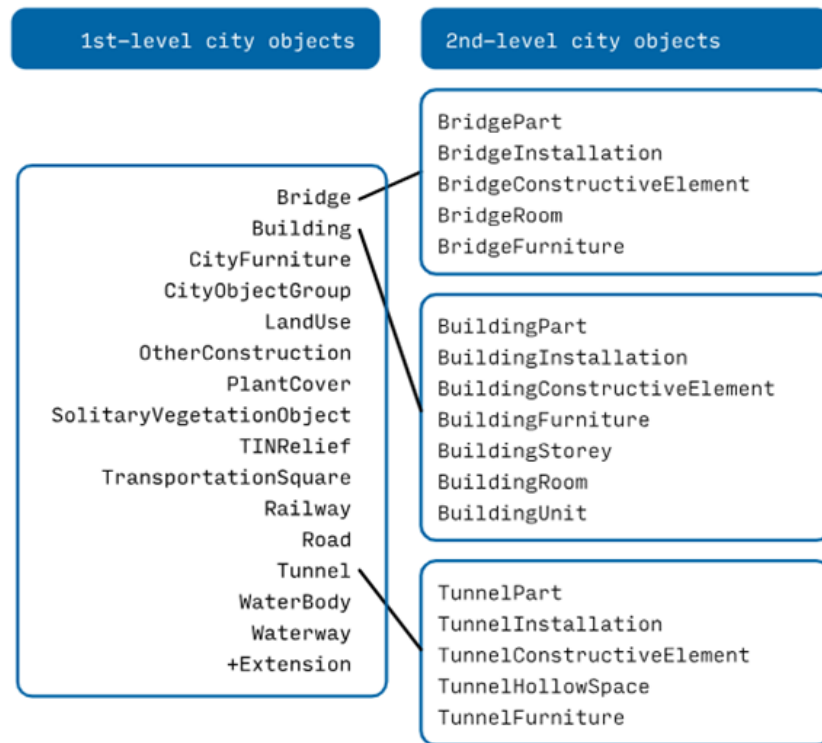
```
1  "CityObjects": {  
2    "id-1": {  
3      "type": "Building",  
4      "attributes": {...},  
5      "children": ["id-2", "id-3"],  
6      "geometry": [{...}]  
7    },  
8    "id-2": {  
9      "type": "BuildingPart",  
10     "parents": ["id-1"],  
11     "geometry": [{...}]  
12     ...  
13   },  
14   "id-3": {  
15     "type": "BuildingPart",  
16     "parents": ["id-1"],  
17     "geometry": [{...}]  
18     ...  
19   }  
20 }
```

## Codificación CityJSON II

### CityJSON:

- Es muy adecuado para aplicaciones web
- Reduce el tamaño de los datos; El archivo JSON ocupa aproximadamente 6 veces menos espacio que XML
- Se puede almacenar tanto en bases de datos relacionales como en bases de datos no-SQL

Dos tipos de objetos de la ciudad: 1º y 2º nivel (padres e hijos)

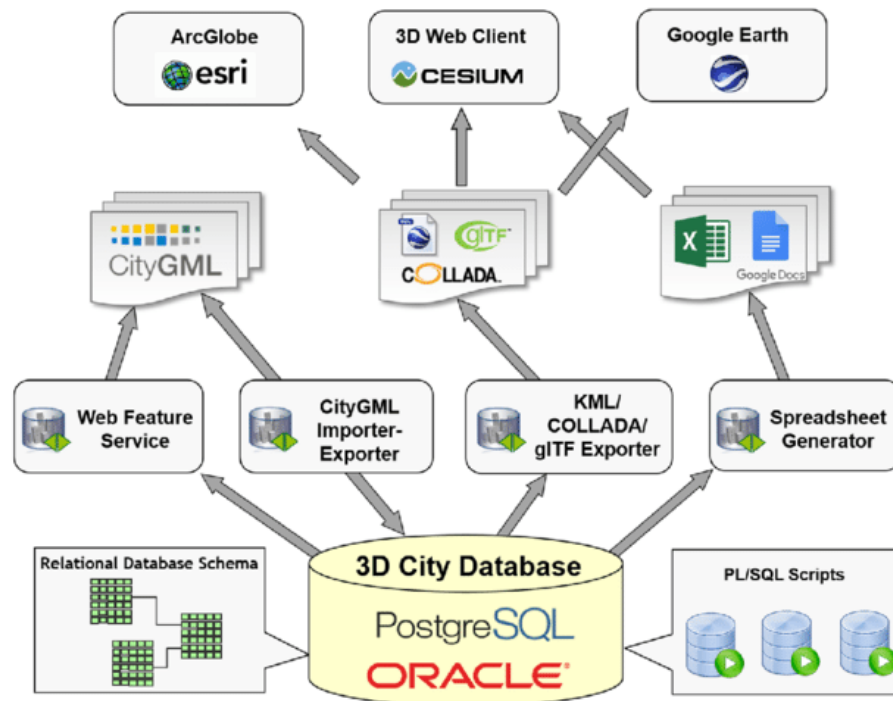




## Codificación de bases de datos de ciu

- Esquema de base de datos llamado 3DCityDB
- No es estándar oficial
- Incluso software de código abierto

[3DCityDB Database – Homepage](#)

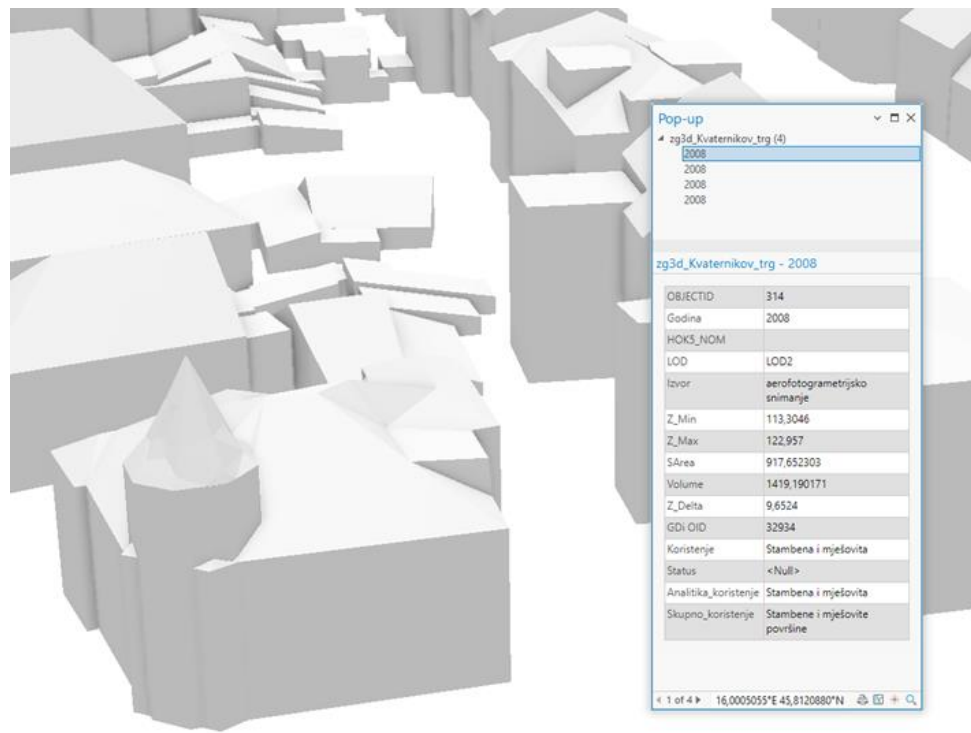


3DCityDB: una solución de geodatabase 3D para la gestión, el análisis y la visualización de modelos semánticos de ciudades en 3D basados en CityGML, Yao et al. (2018)

## Otros formatos 3D

GML se puede combinar con muchos otros formatos

- Web Feature Service (WFS)
- Web Processing Service (WPS)
- KML/COLLADA or X3D files
- Web 3D Service (W3DS)
- Web Terrain Service (WTS)
- Indoor GLM



Ejemplo de un modelo 3D

## LandInfra

- Otro estándar de datos 3D
- Ingeniería terrestre y civil
- Algunos se solapan con CityGML
- Incluye funciones no disponibles en CityGML



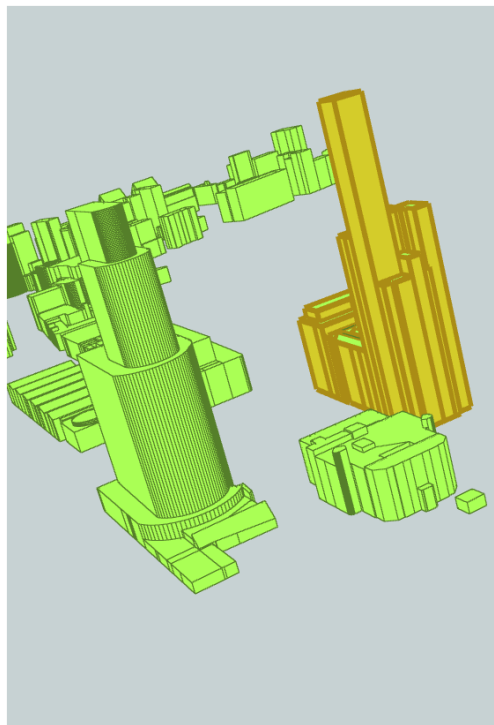
[Comparison between IFC, CityGML and LandInfra. Source: LandInfra BIM GIS.pdf](#)

## Inspirar edificios 3D

Influenciado por CityGLM, pero simplificado

Objetivo de garantizar la interoperabilidad de los datos y servicios espaciales de los diferentes países de la UE

Requisitos de las directivas de la UE (por ejemplo, ruido, rendimiento energético)



Property	Value
Feature Type	BuildingPart
Coordinate System	EPSG:28992
Dimension	3D
Number of Vertices	606
Min Extents	92930.945999999996, 435425.1972, 0.0
Max Extents	93029.243000000002, 435500.672000000002, 135.50008840000001
Attributes (12)	
beginLifespanVersion (encoded: utf-16)	2013-01-15T00:00:00
conditionOfConstruction.owns (encoded: utf-16)	false
conditionOfConstruction.xsi_nil (encoded: utf-16)	true
fme_geometry (string)	fme_aggregate
fme_type (string)	fme_area
geometry3DLoD2.BuildingGeometry3DLoD2.horizontalGeometry...	1.0
geometry3DLoD2.BuildingGeometry3DLoD2.horizontalGeometry...	m
gml_id (encoded: utf-16)	fme-gen-778f112c-1978-4-ca9-a52d-f1c525f42f52
gml_original_coordinate_system (encoded: utf-16)	EPSG:28992
inspireId.Identifier.localId (encoded: utf-16)	fme-gen-778f112c-1978-4-ca9-a52d-f1c525f42f52
inspireId.Identifier.namespace (encoded: utf-16)	EUJRC.BU
xml_type (string)	xml_area
IFMEMultiArea (123 Parts)	
Name (encoded: utf-16)	geometry3DLoD2.BuildingGeometry3DLoD2.geometryMultiSurface
Geometry Traits (1)	
Part 0: IFMEPolygon	
Name (encoded: utf-16)	surfaceMember
Geometry Traits (1)	
gml_id (encoded: utf-16)	fme-gen-778f112c-1978-4-ca9-a52d-f1c525f42f52-1
Linear Boundary	True
Convex	True
Orientation	Right Hand Rule
Boundary: IFMELINE (4 Coordinates)	(92971.863129999998, 435461.867299999998, 32.190855839999998), ...
Closed	Yes
Coordinates (4)	
0	Coordinate Dimension: 3
1	92971.863129999998, 435461.867299999998, 32.190855839999998
2	92983.949670000002, 435471.024700000001, 32.190962329999998
3	92983.949670000002, 435471.024700000001, 32.19067957
4	92971.863129999998, 435461.867299999998, 32.190855839999998
Part 1: IFMEPolygon	
Name (encoded: utf-16)	surfaceMember
Geometry Traits (1)	
gml_id (encoded: utf-16)	fme-gen-778f112c-1978-4-ca9-a52d-f1c525f42f52-2
Linear Boundary	True
Convex	True
Orientation	Right Hand Rule
Boundary: IFMELINE (5 Coordinates)	(92983.949670000002, 435471.024700000001, 32.190962329999998), ...
Closed	Yes

INSPIRE Buildings GML visto con el Inspector de datos. De: [Converting CityGLM to INSPIRE 3D Buildings \(Annex III\) \(safe.com\)](#)

**Gracias por su atención**



<https://birgitproject.eu/>

*Financiado por la Unión Europea. Las opiniones y puntos de vista expresados solo comprometen a su(s) autor(es) y no reflejan necesariamente los de la Unión Europea o los de la Agencia Ejecutiva Europea de Educación y Cultura (EACEA). Ni la Unión Europea ni la EACEA pueden ser considerados responsables de ellos.*