



HAL
open science

Gestion et visualisation multiscalair du territoire au bâtiment : Application à la Gestion et Maintenance assistée par Ordinateur

Clément Colin

► **To cite this version:**

Clément Colin. Gestion et visualisation multiscalair du territoire au bâtiment : Application à la Gestion et Maintenance assistée par Ordinateur. Environnements Informatiques pour l'Apprentissage Humain. Université Lumière - Lyon II, 2024. Français. NNT : 2024LYO20010 . tel-04712358

HAL Id: tel-04712358

<https://theses.hal.science/tel-04712358v1>

Submitted on 27 Sep 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



N° d'ordre NNT : 2024LYO20010

THÈSE de DOCTORAT DE L'UNIVERSITÉ LUMIÈRE LYON 2

École Doctorale : ED 512
Informatique et Mathématiques

Discipline : Informatique

Soutenue publiquement le 18 juin 2024 par :

Clément COLIN

Gestion et visualisation multiscalaire du territoire au bâtiment.

Application à la Gestion et Maintenance assistée par Ordinateur.

Devant le jury composé de :

Marlène VILLANOVA-OLIVER, Professeure, Université Grenoble Alpes, Présidente

Géraldine MORIN, Professeure, Institut National Polytechnique de Toulouse, Rapporteuse

Alain BOUJU, Maître de conférences HDR, Université de la Rochelle, Rapporteur

Anne PUISSANT, Professeure, Université de Strasbourg, Examinatrice

Didier JOSSELIN, Directeur de recherche, CNRS, Examineur

Gilles GESQUIÈRE, Professeur, Université Lumière Lyon 2, Directeur de thèse

Sylvie SERVIGNE, Maîtresse de conférences, INSA Lyon, Co-Directrice de thèse

John SAMUEL, Enseignant chercheur, CPE Lyon, Co-Directeur de thèse

Contrat de diffusion

Ce document est diffusé sous le contrat *Creative Commons* « [Paternité – pas de modification](#) » : vous êtes libre de le reproduire, de le distribuer et de le communiquer au public à condition d'en mentionner le nom de l'auteur et de ne pas le modifier, le transformer ni l'adapter.

Gestion et visualisation multiscalair du territoire au bâtiment.

Application à la Gestion et Maintenance Assistées par Ordinateur

Clément Colin

Université de Lyon, Université Lumière Lyon 2, LIRIS UMR-CNRS 5205
2024

Une thèse présentée pour obtenir le grade de
Docteur en Informatique
École doctorale : ED 512 “Informatique et Mathématiques”



Composition du jury

Alain Bouju, Maître de conférences HDR, La rochelle Université, Rapporteur
Géraldine Morin, Professeur, Institut National Polytechnique de Toulouse, Rapportrice
Anne Puissant, Professeur, Unistra, Examinatrice
Didier Josselin, Directeur de recherche, CNRS, Examineur
Marlène Villanova-Oliver, Maîtresse de conférences HDR, Université Grenoble Alpes, Examinatrice
Gilles Gesquière, Professeur, Université Lumière Lyon 2, Directeur de thèse
Sylvie Servigne, Maîtresse de conférences, INSA Lyon, Co-encadrante de thèse
John Samuel, Enseignant-Chercheur, CPE Lyon, Co-encadrant de thèse
Christophe Bortolaso, Responsable recherche appliquée, Berger-Levrault, Invité

Remerciements

En premier lieu, je souhaite exprimer ma profonde gratitude envers Gilles Gesquière, Sylvie Servigne et John Samuel, qui m'ont accordé leur confiance dès le début et tout au long de cette aventure. Je les remercie pour leur soutien infaillible et leurs conseils avisés, ainsi que pour le partage de leur expérience. Je suis extrêmement reconnaissant envers Carl Software | Berger-Levrault pour le financement de ce projet de recherche. En particulier, je suis très reconnaissant envers Christophe Bortolaso, Laurent Truscello et Youssef Miloudi pour leur confiance constante et leur encouragement. Je souhaite à tous doctorants d'avoir la chance d'être encadrés par des personnes si bienveillantes, encourageantes et inspirantes au quotidien. Ils m'ont permis de grandir en tant que chercheur, mais aussi en tant que personne.

Je tiens également à remercier les rapporteurs et membres du jury, Alain Bouju, Géraldine Morin, Anne Puissant, Didier Josselin, Marlène Villanova-Oliver, d'avoir accepté d'évaluer ce travail de thèse.

De plus, j'aimerais remercier les membres du projet VCity au sein du Liris, Eric Boix, Diego Alvarez-Vinasco, Corentin Gautier, Valentin Machado, Mathieu Livebardon, Lorenzo Marnat, et Jey Puget Gil, avec qui j'ai pris grand plaisir à travailler et évoluer pendant toutes ces années. Nos nombreux échanges, parfois empreints de profonde réflexion, ont été des moments précieux pendant toutes ces années. J'aimerais également remercier mes collègues de chez Carl Software | Berger-Levrault : Mehdi Ayadi, Elodie Toufailli, Anais Vatinel, Benoit Verhaeghe, Boubou Thiam Niang, Elric Gumbau, Sebastien Dufour, Jean-Michel Joerger et Karim Boutamine, pour leur aide et leur encouragement permanents.

Il est bien connu que la réalisation d'une thèse est un parcours semé d'embûches, tant d'un point de vue scientifique, professionnel ou personnel. Une grande partie de ces travaux de thèse ont été effectués pendant la période particulière de la pandémie du COVID. Dans ce contexte, je tiens à remercier une nouvelle fois Eric Boix de m'avoir donné l'impulsion et l'accompagnement nécessaire afin d'entamer les démarches pour me sortir d'une situation particulièrement difficile mentalement.

À ma famille, je tiens à exprimer ma gratitude pour leur constante sollicitude et leur attachement sans faille.

Un immense merci envers ma seconde famille, mes amis précieux : Axel et Grâce, William et Léa, Michael et Valentine, Lucas, et Hugo. Leur compagnie et leur soutien inconditionnel dans les bons et mauvais moments m'ont été d'une aide inestimable. Leur présence et leur amitié ont été des sources de force et de joie précieuses.

Enfin, je ne remercierai jamais assez Chloé qui est un pilier essentiel dans ma vie : sa présence bienveillante, ses mots d'encouragement et sa foi inébranlable en moi ont formé un foyer de réconfort et de force pendant toutes ces années tumultueuses, qui ouvrent aujourd'hui la voie à un avenir lumineux.

Merci à tous de me donner l'envie et la motivation d'être la meilleure version de moi-même au quotidien.

Résumé

La ville et les objets qui la composent, comme les bâtiments, les réseaux d'eau, électriques ou routiers, agrémentent aujourd'hui de véritables jumeaux numériques toujours plus précis jouant un rôle important pour la compréhension des territoires. L'avènement de l'utilisation de Systèmes d'Informations Géographiques (SIG), de modélisation de bâtiments (BIM - Building Information Model) et de la ville (CIM - City Information Model) a entraîné la création d'un nombre important de représentations géospatiales de ces objets urbains, constituées de données géométriques et sémantiques, structurées par de nombreux standards. Ces représentations apportent diverses informations thématiques et spatiales pour décrire ce que sont ces objets physiquement, fonctionnellement, et opérationnellement. Une meilleure compréhension de ces objets urbains peut être apportée par des applications permettant aux utilisateurs d'accéder, de visualiser et d'analyser ces objets urbains en utilisant ces différentes représentations.

Dans cette thèse, nous nous intéressons à la navigation et visualisation interactive multiscalaire sur le web de multiples représentations d'un même objet. Nous prendrons en compte différents standards hétérogènes permettant la représentation de l'intérieur et de l'extérieur d'un bâtiment et d'une ville. Nos deux premières contributions permettent la création de vues navigables et contextuelles de ces représentations hétérogènes dans un même contexte web à l'aide d'approches reposant sur des méthodes d'intégration de données. Pour cela, nous proposons une méthodologie et un outil open-source, Py3DTilers, pour extraire, manipuler et visualiser la géométrie de données géospatiales, ainsi qu'une méthodologie d'intégration de données sémantiques basée modèle, afin de garantir que toutes les informations présentes dans ces données soient disponibles. Notre troisième contribution est la formalisation du concept de variant -instance ou ensemble d'instances représentant le même objet- et une proposition de son utilisation pour référencer et naviguer dans un ensemble de représentations d'un même objet. Enfin, notre dernière contribution s'intéresse au choix de la représentation géométrique à afficher d'un objet selon le contexte 3D de l'utilisateur. Nous proposons une étude des niveaux de détails décrits dans différents standards de données géospatiales ainsi qu'une métrique permettant de décrire la complexité d'une représentation géométrique pour permettre ce choix.

Enfin, cette thèse s'est déroulée en partenariat avec la société Carl Software - Berger-Levrault, éditeur de logiciels et de solutions de gestion et maintenance assistée par ordinateur. Un intérêt tout particulier a donc été porté à l'interopérabilité (utilisation

de standards), la réutilisabilité (création d'architecture logicielle partagée basée sur des outils open-source) et la reproductibilité des solutions proposées. L'objectif au travers de cette thèse est d'améliorer la compréhension des équipements pour en faciliter la maintenance et la gestion. Cela est permis grâce à la mise en place d'un lien naturel entre les représentations d'un équipement existantes dans un outils de gestion d'équipements et dans diverses sources de données géospatiales, la visualisation 3D de ces objets et l'exploitation des connaissances que l'on peut retrouver dans ces représentations.

Abstract

Cities and the objects that make them up, such as buildings, water, electricity and road networks, have increasingly precise digital twins that play an important role in understanding territories. The growing use of Geographic Information Systems (GIS), Building Information Model (BIM) and City Information Model (CIM) has led to the creation of a large number of geospatial representations of these urban objects, made up of geometric and semantic data, structured by numerous standards. These representations provide a variety of thematic and spatial information to describe what these objects are physically, functionally and operationally. A better understanding of these urban objects can be provided by applications enabling users to access, visualize and analyze these urban objects using these different representations.

In this thesis, we focus on multiscale interactive web navigation and visualization of multiple representations of the same object. We will consider various heterogeneous standards for representing the interior and exterior of a building and a city. Our first two contributions enable the creation of navigable and contextual views of these heterogeneous representations in a single web context, using approaches based on data integration methods. To this end, we propose a methodology and an open-source tool, Py3DTilers, for extracting, manipulating and visualizing the geometry of geospatial data, as well as a model-based semantic data integration methodology, to ensure that all the information present in these data can be brought and understood by the users. Our third contribution is the formalization of the concepts of Variant-instance or set of instances representing the same entity- and Variant Identifier to reference and navigate through a set of representations of the same object. Finally, our last contribution focuses on the choice of geometric representation of an object to be displayed, depending on the users' 3D context. We propose a study of the levels of detail described in different geospatial data standards, as well as a metric for describing the complexity of a geometric representation to enable this choice.

Finally, this thesis was carried out in partnership with Carl Software - Berger-Levrault, a publisher of computer-aided maintenance software and asset management solutions. Particular attention was paid to interoperability (use of standards), reusability (creation of shared software architecture based on open-source tools) and reproducibility of the proposed solutions. This thesis aims to improve the understanding of equipment to facilitate its maintenance and management, by allowing the 3D visualization of equipment and the exploitation of the knowledge that can be found in various representations. This is achieved by establishing a natural link between

equipment representations existing in this domain and various geospatial data sources.

Table des matières

1	Introduction	1
1.1	Contexte industriel et scientifique	4
1.2	Problématiques	5
1.3	Contributions	9
1.4	Plan	11
2	Etat de l'art	13
2.1	Modélisation et intégration de données géospatiales	13
2.2	Navigation multiscalaire dans des représentations hétérogènes d'une même entité	25
3	Création de vues unifiées et navigables de données géospatiales	35
3.1	Vue contextuelle d'une entité géospatiale	35
3.2	Homogénéisation de données géométriques géospatiales à l'aide d'un format pivot : 3DTiles	39
3.3	Démonstration et résultats	43
3.4	Discussion	48
4	Intégration des données sémantiques et géométriques de données géospatiales	51
4.1	Méthodologie d'intégration basée modèles de données géospatiales	51
4.2	Démonstration et résultats	55
4.3	Discussion	61
5	Gestion des multiples représentations d'une même entité	63
5.1	Variant : Définition et formalisation des concepts	64
5.2	Navigation dans les variants d'une entité géospatiale	69
5.3	Démonstration et résultats	72
5.4	Discussion	79
6	Navigation multiscalaire	81
6.1	Conceptualisation d'une entité : étude de cas des modèles CityGML et IFC	82
6.2	Visualisation multiscalaire d'entités et de leurs représentations spatiales	94
6.3	Expérimentation, premiers résultats et conclusion	99

7 Contributions techniques : applications, outils open source et reproductibilité	107
7.1 Méthodologie	107
7.2 UD-SV : Plateforme de services et de visualisation de données urbaines pour le partage de la recherche multidisciplinaire	109
7.3 BL-Viz : Viewer 3D de jumeau numérique, du prototype à l’outil intégré aux solutions de GMAO	113
7.4 Discussion	118
8 Conclusion : Contributions et perspectives	121
8.1 Contributions	121
8.2 Perspectives	123
Bibliographie	127

Table des figures

1.1	BuildingSMART France-Mediaconstruct, “BIM Book : l’essentiel pour la transformation numérique de la Construction”, 2019	3
1.2	Un exemple de deux maquettes numériques d’une même entité, apportant différentes données géométriques et sémantiques (adapté de [Jai+21]). Ces modèles apportent des informations thématiques (fonction ? description ?), spatiales (où ?) et temporelles (quand ?)	4
1.3	Différentes représentations n-dimensions (2D, 3D, thématique) d’un même bâtiment aux échelles CIM et BIM	6
2.1	Les relations entre les données, les schémas(ou langages), les modèles de données abstraits et un univers du discours (adapted from [Kut16]).	14
2.2	Représentation des données sémantiques et géométriques d’une maquette numérique CityGML 3.0[OGC18a]	15
2.3	Les différentes modélisations possibles des données géométriques, adapté de [Don+16] avec l’ajout de "polygonal modeling"	16
2.4	Les différentes approches pour intégrer des données BIM-GIS, par [BBK20]	18
2.5	Vue d’ensemble d’un tileset 3DTiles [OGC19]	20
2.6	Illustration du fonctionnement des feature table et batch table des tuiles au format B3DM et I3DM[OGC19]	21
2.7	Comparaison des différentes méthodes d’intégrations de données BIM-GIS, selon [Liu+17]	23
2.8	Etude des différentes approches pour intégrer des données géométriques avec des technologies du web sémantique, selon [Wag+20]	24
2.9	Différence de modélisation des géométries et de conceptualisation des entités entre IFC et CityGML[LB11]	25
2.10	Attributs d’un 3D-Container selon le modèle logique 3D-Container et exemple d’une collection de 3D-Container[MTS20]	27
2.11	Niveaux de zoom SIG utilisés par OpenStreetMap et un exemple de tuiles au niveau d’échelle 2	28
2.12	Les différents niveaux de détails (de LoD0 à LoD4) décrits dans le modèle de données CityGML 3.0[OGC18a]	29

2.13	Trois exemples de maquettes CityGML d'un bâtiment avec différentes cohérences spatio-sémantiques [SK07] : une maquette sans données sémantique, une maquette avec des données géométriques structurées et hiérarchisées et une maquette avec des données sémantiques et géométriques structurées et hiérarchisées, dites cohérentes.	30
2.14	Exemple de description de niveau de développement (LOD) de certaines classes BIM [AB22], constitués de niveau de géométrie (LoG) et niveau d'information (LoI).	31
2.15	Création de niveaux de détails d'un quartier à partir de données existantes, adapté de [GPG18]	31
2.16	Exemple d'un système de niveaux de détails (LoD) [La +22] défini selon différents modèles de données géométriques.	32
2.17	Alignement des LoD CityGML et IFC sur le niveau d'échelle de OpenStreetMap	33
3.1	Méthodologie de création de vues contextuelles proposée, adaptée de [Col+22]	37
3.2	Diagramme d'activités du process de création de 3DTiles à partir de données géospatiales from Marnat et al.[Mar+22]	42
3.3	Architecture technique, adapté de [Col+22]	44
3.4	Visualisation de 3D Tiles avec différents viewers [Mar+22]	45
3.5	Comparaison de géométries CityGML et IFC transformées en 3D Tiles [Mar+22]	46
3.6	Vue contextuelle d'un toit constituée de représentations thématiques issues d'un fichier IFC (dans le cadre A), de la géométrie (dans le cadre B), d'un document (dans le cadre C), et d'une représentation spatiale issue de la géométrie d'un fichier IFC (dans le cadre D) [Col+22]	47
3.7	Création de vues contextuelles, depuis une vue de la ville d'un bâtiment à l'aide d'une représentation spatiale CityGML (en haut à gauche) jusqu'à une vue d'un objet dans ce bâtiment à l'aide d'une représentation géométrique IFC (en bas à droite), adapté de [Col+22]	47
3.8	Exemple de navigation possible entre une représentation thématique issue de la GMAO (à gauche) et une représentation spatiale issue d'un IFC (géométrie 3D en bleu à droite) [Col+22]	48
4.1	Méthodologie d'intégration de données géospatiales proposée. Les éléments en gris ont été ajoutés par rapport à la précédente contribution.	53
4.2	Architecture logique obtenue après application de la méthodologie sur les représentations identifiées par 'Bdg :1' et 'Building :548', qui représentent la même entité du monde réel.	55

4.3	Architecture technique. La base de connaissances permettant d'héberger des données sémantiques (modèles de données et données sémantiques) en graphes (en gris) a été ajoutée par rapport à la précédente contribution.	56
4.4	Visualisation de deux représentations spatiales d'une même entité (ici un bâtiment) issues de données IFC et CityGML, transformées en 3DTiles	59
4.5	Visualisation de deux représentations thématiques d'un même bâtiment issues de données IFC et CityGML, transformées en RDF	59
4.6	Visualisation d'un bâtiment à l'aide d'une représentation thématique, sous forme de graphe, et d'une représentation spatiale, sous forme de géométrie 3D, issues de données CityGML. Ces représentations sont liées, permettant d'interagir entre elles.	60
4.7	Visualisation d'un bâtiment à l'aide d'une représentation thématique, sous forme de graphe, et d'une représentation spatiale, sous forme de géométrie 3D, issues de données IFC. Le graphe représente les entités qui en composent une autre (ici, un bâtiment). La géométrie de l'étage correspondant au noeud survolé est coloré en bleu.	61
5.1	Exemple 1 : Multiples instances (Deployment Diagram) issues d'un même modèle (Class Diagram) et représentant une même entité géospatiale (Real World).	64
5.2	Modèle logique de variants selon [Bec+21]	65
5.3	Exemple 2 : Multiples instances (Deployment Diagram) issues de modèles hétérogènes (Class Diagram) provenant de diverses sources de données et représentant une même entité géospatiale (Real-World Diagram) Les instances de gauche sont des instances de bâtiment CityGML LOD1 et LOD2. Celle du centre est une image. Celle de droite est une instance de la classe IFCBuilding.	65
5.4	Exemple 3 : Ensemble d'instances représentant une entité géospatiale par composition. Les instances sont hébergées par un même serveur et sont décrites par le modèle CityGML. L'ensemble des instances représentant des bâtiments et un pont composent une représentation de la ville de Lyon.	66
5.5	Modèle logique de Variant proposé : Instance ou ensemble d'instances hébergée dans une source de données (base de donnée, fichier, service etc.) représentant la même entité géospatiale différemment.	67
5.6	Modèle logique de Variant Complexe proposé : Ensemble d'instances représentant des entités différentes qui composent une entité parente. Deux instances de deux classes du même modèle A qui représentent respectivement les entités B et C représentent l'entité A par composition.	68

5.7	Modèle logique de Variant Identifier : Un Variant Identifier permet de référencer et décrire un variant. Il se compose d'un <i>Variant Locator</i> qui décrit la source de données et d'un <i>Variant Accessor</i> qui décrit comment récupérer l'instance ou l'ensemble d'instances à partir de cette source. .	70
5.8	Modèle logique d'abstraction d'une entité géospatiale pour naviguer entre ses variants à l'aide de Variant Identifiers	72
5.9	Extension du modèle 3D-Container (en blanc) à l'aide du modèle de Variant Identifier (en gris) . L'attribut <i>href</i> correspond au <i>Variant locator</i> . La nouvelle classe <i>LinkToVariant</i> hérite de la classe <i>link</i> et correspond au <i>Variant Accessor</i> . La classe <i>3DContainer</i> correspond à une abstraction d'une entité géospatiale.	73
5.10	Architecture technique. Les données 3DContainer permettant de référencer les variants d'entités (en gris) ont été ajoutés par rapport à la précédente contribution.	74
5.11	Descriptions des entités géospatiales et leurs variants utilisés dans la démonstration.	75
5.12	Page d'accueil de l'API Geovolume . Les entités géospatiales et leurs variants sont listés grâce à leurs représentations sous forme de 3D-Container dans le serveur.	76
5.13	Vue des entités sous UD-Viz . 5 entités géospatiales sont visibles : la métropole de Lyon, deux villes : Villeurbanne et Limonest, et deux bâtiments à l'intérieur de ces villes. Elles sont affichées sous forme de simples cubes géolocalisés dans la scène 3D (à droite) et sous forme de liste avec les variants disponibles pour chacune d'entre elles (à gauche).	77
5.14	Vue composée de multiples variants d'une même entité . Deux variants sont affichés dans la même vue de la ville de Villeurbanne : une représentation géométrique (au centre) et un graphe (sur la droite). .	78
5.15	Visualisation de deux variants de deux entités géospatiales distinctes à partir de la même source de données . Le variant de la ville de Limonest est un fichier 3DTiles entier (a). Le variant du bâtiment Carl Software est un objet géométrique (b) dans ce fichier, visualisé en utilisant son identifiant.	78
5.16	Visualisation de deux variants de deux entités géospatiales distinctes à partir de la même source de données . Le variant de la ville de Villeurbanne est un fichier complet de nuage de points 3DTiles (a). Le variant d'un bâtiment à l'intérieur de la ville de Villeurbanne est l'ensemble de points contenu dans sa bounding box issu du même fichier 3DTiles (b).	79

6.1	Diagramme UML des concepts d'espaces dans CityGML 3.0 [OGC18a]	83
6.2	Exemple de deux espaces logiques CityGML 3.0, définis selon le type des entités : arbre ou bâtiment	84
6.3	Exemple de deux espaces physiques CityGML 3.0 : le premier est entièrement délimité par des objets physiques, le second est partiellement délimité par des objets physiques	85
6.4	Classification en espace occupé et non occupé d'entités urbaines selon CityGML 3.0 [OGC18a]	86
6.5	Illustration des concepts CityGML 3.0 d'espaces de circulations (TrafficSpace), composé d'un espace de déplacement (ClearanceSpace) et d'une aire de circulation (TrafficArea)	86
6.6	Héritages des types <i>IfcElement</i> et <i>IfcSpatialElement</i> selon le modèle IFC	88
6.7	Exemple d'une structure spatiale d'un bâtiment constitué de deux sections de deux et trois étages chacune, selon le modèle IFC	89
6.8	Exemple d'une structure spatiale d'un bâtiment constituée de deux sections, d'un escalier et de deux murs, selon le modèle IFC	90
6.9	Illustration des trois différents modes de représentations identifiés : une représentation spatiale directe, une représentation spatiale créée par composition et une représentation spatiale créée par agrégation.	93
6.10	Exemple de décomposition d'une entité physique en entités physiques et abstraites. Les entités enfants qui font parties des limites physiques d'une autre entité sont annotées comme <i>externe</i> .	95
6.11	Illustration des différentes représentations spatiales que peuvent apporter des variants issus de données géospaciales.	96
6.12	Association des LoD (Level Of Detail) CityGML et LOD (Level Of Development) IFC à des niveaux d'échelles. L'ensemble du tableau est proposé dans la figure 2.17	97
6.13	Illustration d'une complexité de variants basée sur le nombre de points (ici, des sommets) (C) de la représentation et le nombre d'entités distinctes (N)	98
6.14	Extension du modèle 3D-Container (en blanc) avec la classe Geometry Descriptor (en gris) qui permet d'associer des métadonnées aux variants correspondant à sa complexité.	99
6.15	Interface de l'expérimentation mise en place pour comparer différents LoC. Dans le cadre A, on retrouve le LoS de l'entité, le LoS minimum et maximum et le critère utilisé pour calculer le LoC. La bounding box B représente l'espace qu'occupe l'entité. Le cadre C représente les différents niveaux d'échelles et le niveau d'échelle courant.	101

6.16	Enchaînement de deux représentations de Villeurbanne en fonction d'un LoC défini par le nombre de vertex dans la représentation	102
6.17	Enchaînement de trois représentations du bâtiment de CarlSoftware en fonction d'un LoC défini par le nombre de vertex dans la représentation	103
7.1	Méthodologies appliquées dans la plateforme UD-SV (à gauche) et Berger-Levrault (à droite) pour construire une solution scientifique robuste, documentée, réutilisable et reproductible	108
7.2	Architecture technique à 4 niveaux de la plateforme UD-SV, adapté de [Sam+23]. Les composants développés ou étendus au cours durant les travaux de cette thèse sont surlignés en bleu. Les autres composants ou logiciels utilisés sont entourés en bleu.	110
7.3	Page d'accueil de CarlSource, avec le prototype intégré	114
7.4	Vue 3D d'équipements dans le prototype. Un équipement est sélectionné et affiché en bleu, ses informations sont affichés dans le cadre en bas à gauche, dans lequel il est possible d'effectuer des opérations de maintenance supplémentaires.	115
7.5	Vue composée d'une vue 3D des équipements dans le prototype et d'une vue de l'arborescence des équipements. Un équipement est sélectionné et affiché en bleu, ses informations sont affichés dans le cadre en bas à gauche du prototype, sa représentation dans l'arborescence est colorée en bleu.	116
7.6	Architecture technique de BL-Viz, outil intégré dans CarlSource	117
7.7	Visualisation d'un bâtiment IFC dans le viewer BL-Viz. Un objet (en bleu) est sélectionné et l'interface affichée permet d'effectuer des opérations de maintenance classique.	118

Liste des tableaux

4.1	Taille des fichiers avant et après transformation des données sémantiques en graphes (RDF) et des données géométriques en un format homogène 3D (3DTiles).	58
6.1	Proposition d'alignement des concepts d'espaces et d'entités par CityGML et IFC	91

7.1	Identifiants permanents SoftWare Heritage (SWHIDs) des différents composants UD-SV et d'applications créées avec ces composants pendant les travaux de cette thèse (11 et 12).	112
7.2	Étapes des contributions scientifiques et technologiques proposées vers leurs industrialisations. ✓ : réalisé, / : en cours, × en attente	119

Les publications dans des revues et annales de conférences suivantes ont été produites au cours de cette thèse :

[Col+21] Clément Colin, John Samuel, Sylvie Servigne, Youssef Miloudi, Gilles Gesquière. “Visualisation d’équipements de la maintenance à l’aide de données géospatiales”. Les journées Françaises de l’Informatique Graphique (JFIG) 2021, Nov 2021, Sophia-antipolis, France. ⟨hal-03464886⟩

[Col+22] Clément Colin, John Samuel, Sylvie Servigne, Christophe Bortolaso, Gilles Gesquière. “Creating contextual view of CMMS assets using geospatial 2D/3D data”. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2022, X-4/W2-2022, pp.37-44. ⟨10.5194/isprs-annals-X-4-W2-2022-37-2022⟩. ⟨hal-03818710⟩

[Mar+22] Lorenzo Marnat, Corentin Gautier, Clément Colin, Gilles Gesquière. “PY3DTILERS : An open source toolkit for creating and managing 2D/3D geospatial data”. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2022, X-4/W3-2022, pp.165-172. ⟨10.5194/isprs-annals-X-4-W3-2022-165-2022⟩. ⟨hal-03852828⟩

[Sam+23] John Samuel, Vincent Jaillot, Clément Colin, et al. “UD-SV : Urban data services and visualization framework for sharing multidisciplinary research”. en. In : Transactions in GIS 27.3 (Apr. 2023), pp. 841-858

[Col+23] Clément Colin, Corentin Gautier, Diego Vinasco-Alvarez, et al. “UD-SV : Plateforme d’exploration de données urbaines à n-dimensions — Espace, Temps, Thématiques”. fr. In : Mappemonde. Revue trimestrielle sur l’image géographique et les formes du territoire 135 (Mar. 2023). Number : 135 Publisher : UMR Espace

[Vin+24] Clément Colin, Diego Vinasco-Alvarez, et al. “A model-driven methodology for integrating heterogeneous geospatial semantic and geometric data“ 27th AGILE Conference on Geographic Information Science

Les travaux réalisés pendant cette thèse ont notamment pu être présentés lors de différents évènements : dans le salon BIM World ¹ de 2022, 2023 et 2024, dans le webinaire "Autour de la 3D", coorganisé par les GDR CNRS MAGIS et IG-RV ², et devant le groupe City Science du MIT Media Lab ³.

1. <https://bim-w.com/>

2. <https://github.com/VCityTeam/MAGIS-AP3D>

3. <https://www.media.mit.edu/groups/city-science/overview/>

Introduction

Le terme "jumeau numérique" provient de la gestion du cycle de vie d'équipements (PLM - Product Life Cycle Management) [Gri15], dans lequel un modèle numérique (le jumeau) d'une **entité** (*quelque chose qui a une existence séparée et distincte et une réalité objective ou conceptuelle* [ISO16]) est créé pour pouvoir surveiller, tester et simuler différents phénomènes au cours de son cycle de vie [Jon+20]. Un jumeau numérique urbain (Urban Digital Twin) de la ville et de ses objets fédère des données géospatiales, sans les dégrader, dans une visualisation de la ville qui permet de transformer ces données en information. On retrouve d'ores et déjà des projets de jumeaux numériques urbains en France, à Rennes¹, à Angers² ou encore très récemment le projet "La Vallée" sur la commune de Châtenay-Malabry [DMJ23]. Au-delà d'usages natifs tels que la visualisation 3D, la gestion de risques et de phénomènes physiques, les applications de jumeaux numériques ont pour but d'agir comme plateforme de collaboration, participation et médiation entre différents acteurs d'une ville : les citoyens, les municipalités, les entreprises et commerçants. Il s'agit d'un pont entre différents domaines permettant d'en améliorer l'interopérabilité - la capacité de différents systèmes à opérer ensemble - :

- Les **SIG** (Systèmes d'Information Géographique) pour la gestion et l'aide à la décision et la communication de données géographiques ;
- Le **BIM** (Building Information Modeling) pour la modélisation et la gestion de données de bâtiments ;
- Le **CIM** (City Information Modeling) pour la modélisation et la gestion de données des villes ;
- **IoT** (l'internet des objets) pour la remontée d'informations mesurables en temps réel ;
- La **GMAO** (Gestion et Maintenance Assistée par Ordinateur) pour la gestion et la maintenance des données d'infrastructures et équipements techniques.

Depuis les années 1990, les villes et territoires se sont équipés de SIG pour gérer, analyser, visualiser et géoréférencer leur patrimoine à l'aide de données géospatiales

1. www.ecocites.logement.gouv.fr/IMG/pdf/ecocite_fiches_top_10_2019_rennes_metropole_pia_bat_en.pdf

2. www.engie-solutions.com/sites/default/files/assets/2020-10/CPConstruktionduJumeauNumerique3DALMSiradeLVDEF%20EN.pdf

sur des cartes numériques 2D. Ils sont notamment utilisés pour travailler avec des plans de rues, des données de recensement, ou encore des informations diverses sur les bâtiments. Ils permettent une analyse et un traçage de réseaux, et servent également à créer des itinéraires et à la planification.

Grâce à une amélioration et une facilitation de l'acquisition de données 3D depuis ces dix dernières années, l'informatique géospatiale dispose aujourd'hui d'une troisième dimension offrant de nouveaux cas d'utilisation [Bil+15] comme l'analyse de visibilité ou l'estimation d'ombres et du potentiel solaire. Un exemple de cette popularisation est, depuis octobre 2023, la mise à disposition par Google de cartes 3D de nombreuses zones peuplées du monde, permettant de mieux comprendre une zone et son évolution³.

À l'échelle de la ville, des maquettes numériques tridimensionnelles (3D) sémantiquement enrichie, issue de processus orientés CIM, sont dorénavant disponibles. De nombreuses villes⁴ proposent d'ailleurs un accès libre à ces maquettes, comme Montpellier⁵, Lyon⁶, Berlin⁷ ou encore New York⁸.

À l'échelle du bâtiment, ce sont les processus centrés BIM qui permettent la création de maquette 3D sémantisée. Le BIM est normalisé et défini [ISO18a] comme une représentation numérique des caractéristiques physiques et fonctionnelles d'une infrastructure, qui peut être utilisée pour la prise de décision tout au long de son cycle de vie. La norme souligne l'importance d'une approche collaborative des différentes parties prenantes à la conception et réalisation d'une infrastructure. La création de maquette numérique 3D BIM se démocratise. En France, on retrouve notamment des programmes comme le Plan Transition Numérique du Bâtiment (2014-2018)⁹, le Plan BIM 2022¹⁰, poursuivi par le Plan BIM ayant pour missions de généraliser l'utilisation du BIM dans l'ensemble du milieu du bâtiment et de déployer le BIM dans les territoires grâce aux outils adaptés. Dans le monde, comme illustré dans la figure 1.1, un grand nombre de pays a d'ailleurs rendu obligatoire l'utilisation du BIM lors de la construction de nouveaux bâtiments.

3. www.developers.google.com/maps/documentation/tile/3d-tiles?hl=fr

4. www.github.com/OloOcki/awesome-citygml

5. www.data.montpellier3m.fr/dataset/modele-3d-de-la-ville-de-montpellier

6. www.data.grandlyon.com/portail/fr/jeux-de-donnees/maquettes-3d-texturees-2018-communes-metropole-lyon/info

7. www.daten.berlin.de/datensaetze/lod2-geb%C3%A4udedaten-berlin

8. www.data.cityofnewyork.us/City-Government/3-D-Building-Model/tnru-abg2

9. www.ecologie.gouv.fr/sites/default/files/Bilan%202018%20PTNB.pdf

10. www.ecologie.gouv.fr/batiment-et-numerique

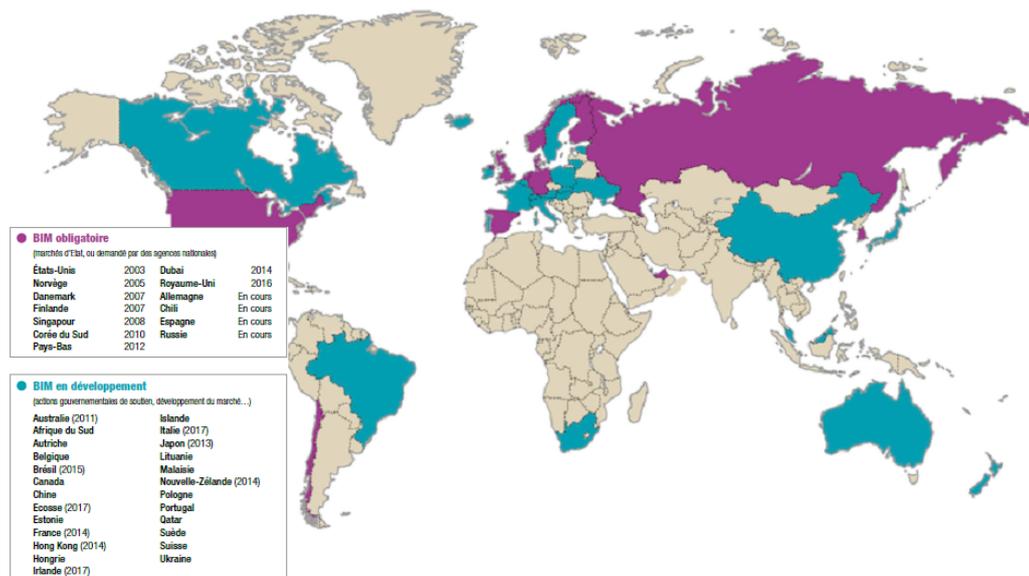


Figure 1.1 : BuildingSMART France-Mediaconstruct, “BIM Book : l’essentiel pour la transformation numérique de la Construction”, 2019

Pour faciliter l’interopérabilité, différents organismes (ISO ¹¹, OGC ¹², Building-Smart ¹³, W3C ¹⁴ ...) et chercheurs proposent des standards hétérogènes pour structurer et diffuser de manière uniforme des données 3D géospatiales urbaines. Ces standards peuvent être une simple documentation informative, ou bien une définition stricte et formelle des concepts et vocabulaires utilisés. Ils peuvent également fournir des modèles et schémas de données pour définir la structure abstraite et formelle (lisible par une machine) des données.

Des données géospatiales décrivant des entités urbaines - bâtiments, végétation, réseaux d’eau, d’électricité et de transport - sont donc de plus en plus disponibles [LSB23 ; Bat18a ; Bat21] pour répondre aux besoins de différentes parties prenantes, telles que des urbanistes, des citoyens, des historiens, des entreprises, ... Ces données fournissent diverses informations pour décrire ces entités physiquement, fonctionnellement et opérationnellement. Ces données sont structurées par un ensemble de modèles de données hétérogènes qui apportent différentes représentations thématiques, spatiales et temporelles à propos d’une même entité, comme illustré dans la figure 1.2.

11. www.iso.org
 12. www.ogc.org
 13. www.buildingsmart.org/standards
 14. www.w3.org

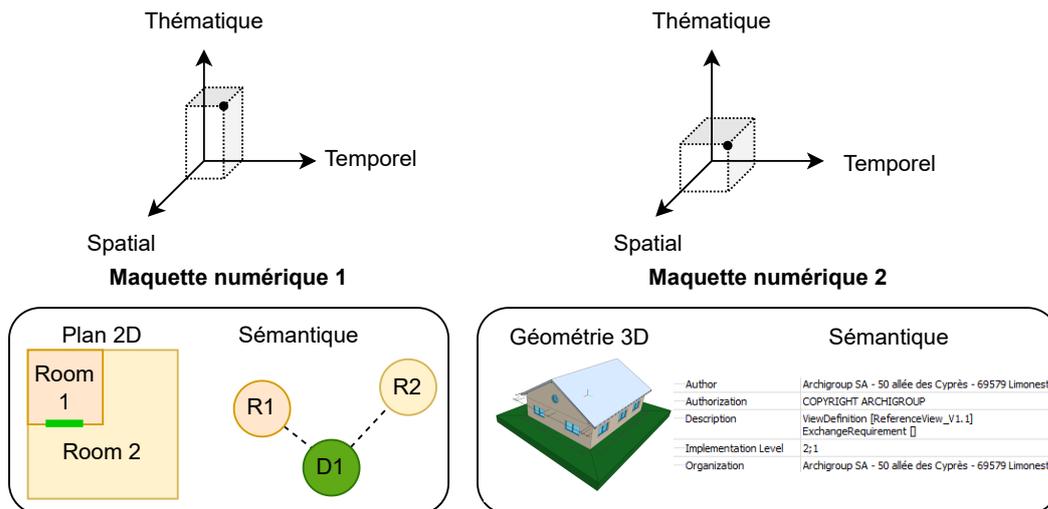


Figure 1.2 : Un exemple de deux maquettes numériques d'une même entité, apportant différentes données géométriques et sémantiques (adapté de [Jai+21]). Ces modèles apportent des informations thématiques (fonction? description?), spatiales (où?) et temporelles (quand?)

L'utilisation de ces données issues de domaines hétérogènes dans un même contexte reste néanmoins encore un problème à résoudre pour permettre une meilleure compréhension de la ville et des objets urbains. Cela requiert la capacité d'identifier et de référencer les représentations d'une même entité existantes dans de multiples sources de données, de travailler avec des modèles de données hétérogènes et de gérer les problématiques distinctes sous-jacentes à l'utilisation et la visualisation de données géométriques et sémantiques dans un même contexte. La création de vues navigables de représentations hétérogènes d'une même entité peut notamment être réalisable à l'aide d'approches reposant sur des méthodes d'intégration de données urbaines [Tra+20].

1.1 Contexte industriel et scientifique

Cette thèse est le fruit d'un travail de recherche effectué en collaboration entre le Laboratoire d'Informatique en Image et Systèmes d'Information, l'Université Lyon 2 et la société Carl Software | Berger-Levrault. Carl Software | Berger-Levrault propose, depuis plus de 30 ans, des solutions de gestion des équipements (Enterprise Asset Management - EAM) et de Gestion et Maintenance Assistée par Ordinateur (GMAO). Cette société déploie depuis plusieurs années une dynamique de développement

de son activité de recherche pour fournir à ses clients des produits innovants, mais également pour construire des programmes à vocation de valorisations concrètes.

Dans ce contexte de recherche, Carl Software | Berger-Levrault souhaite développer une approche scientifique et outillée permettant de penser la gestion des parcs matériels -ensemble des équipements, machines, outils et matériels physiques à maintenir- de leur client, tels que les bâtiments, les installations, les infrastructures et les autres équipements, sur l'ensemble de leur cycle de vie, en s'intéressant à leur conception, leur construction, leur maintenance et leur destruction ou démolition. Plus précisément, cette approche permet d'aborder différents aspects :

- la représentation spatio-temporelle des équipements.
- la fusion de données géospatiales hétérogènes : maquette 3D CIM et BIM, données SIG, données de capteurs.
- la contextualisation de l'information et des fonctionnalités métiers en fonction de l'utilisateur.

Entre autres, cette approche vise l'apport d'autres représentations spatiales et sémantiques d'équipements depuis des données géospatiales pour améliorer la compréhension des équipements à maintenir pour en améliorer la gestion.

Au sein du Liris, cette thèse prends place au sein du projet de recherche VCity¹⁵, qui se concentre sur l'étude de la dynamique des villes et des différents modes de représentations pour permettre une meilleure compréhension de celles-ci. Ces travaux, ancrés entre l'informatique graphique, la géomatique et la science des données, propose notamment de nouveaux algorithmes de traitement et visualisation de données 2D, 3D, sémantiques et temporelles. Dans le cadre de ce projet, des outils open-source sont développés au sein de la plateforme Urban Data Services and Visualization (UD-SV [Sam+23 ; Col+23]) dans l'objectif d'être proposés par et pour les communautés scientifiques et industrielles.

1.2 Problématiques

L'objectif de ces travaux de thèse est de permettre de visualiser et naviguer dans différentes représentations issues de sources de données géospatiales hétérogènes de différentes entités dans un même contexte web. Ces représentations peuvent être formées par des données 2D ou 3D, des données sémantiques, ou encore des données de multimédias, comme illustrées dans la figure 1.3. La visualisation et la navigation

15. www.projet.liris.cnrs.fr/vcity

sous-tendues que nous souhaitons pouvoir laisser la liberté à l'utilisateur de requêter, accéder et visualiser un ensemble de représentations géométriques et sémantiques d'une entité. L'utilisateur doit également pouvoir interagir avec ces données, en effectuant des opérations de maintenance classiques, en naviguant dans l'espace, ou en sélectionnant les entités à visualiser. Afin de s'intégrer aux solutions existantes de l'entreprise Carl-Software, la navigation et l'interaction avec ces représentations doit être réalisable dans un même contexte web. Enfin, les représentations d'une même entité peuvent la décrire à différentes échelles. Il s'agira donc de proposer des solutions afin de pouvoir sélectionner automatiquement les représentations pertinentes d'une entité en fonction de contexte de l'utilisateur : son placement dans l'espace, son métier, son besoin ... Atteindre cet objectif soulève différentes problématiques et questions de recherche.

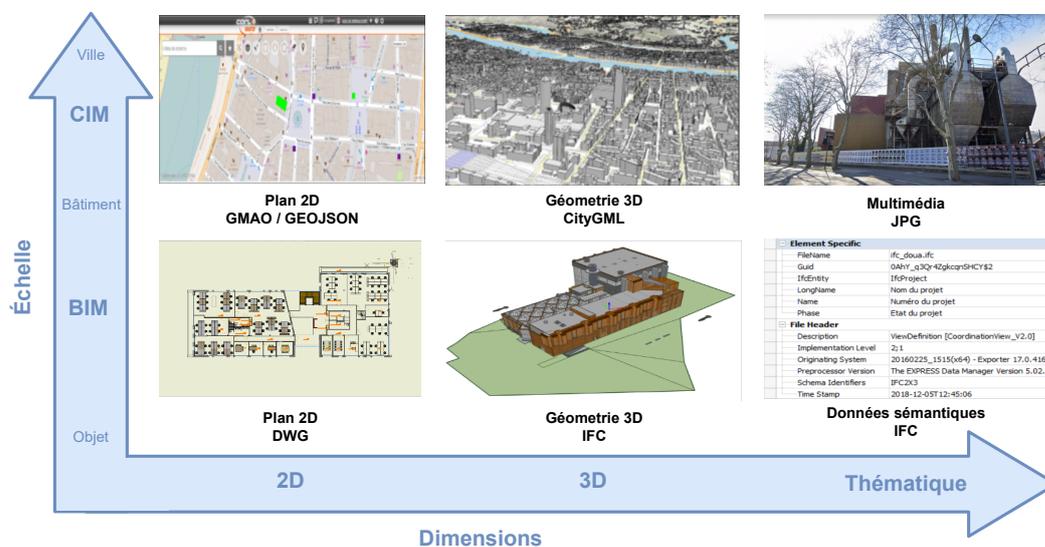


Figure 1.3 : Différentes représentations n-dimensions (2D, 3D, thématique) d'un même bâtiment aux échelles CIM et BIM

Les données géospatiales décrivent des entités urbaines à l'aide de multiples dimensions.

On retrouve dans les données géospatiales des représentations n-dimensions (géométrique 2D, géométrique 3D, sémantique, temporelle, multimédia) d'entités. Chaque dimension dispose de ses propres spécificités et problématiques. Par exemple, les données géométriques doivent être interprétées afin de leur donner du sens et les visualiser dans un contexte 3D dans un viewer web. Les données sémantiques peuvent nécessiter des traitements afin de pouvoir répondre à des questions utilisateurs. De plus, un modèle numérique d'une entité peut la représenter à l'aide d'une seule dimension, par exemple un simple plan 2D, ou de plusieurs dimensions, par

exemple une géométrie 3D enrichie sémantiquement. Ainsi, visualiser l'ensemble des représentations d'un modèle requiert des solutions spécifiques à chaque dimension afin de pouvoir rendre ces informations compréhensibles pour l'utilisateur.

Les données géospatiales sont structurées selon des modèles de données hétérogènes.

L'hétérogénéité des modèles de données urbains existe à différents niveaux : sémantique, conceptuel et technique. L'hétérogénéité sémantique [Kut16] se réfère à la signification ou à l'interprétation de l'information. Un même concept peut être décrit en utilisant un homonyme, un synonyme ou une traduction, comme rue / route / street. L'hétérogénéité conceptuelle [Kut16] se réfère à la différence de définition d'un même concept. Cela apparaît lorsque les modèles de données sont créés selon des points de vue et paradigmes différents. Par exemple, d'un point de vue de la ville, on peut décrire un bâtiment comme un simple ensemble de murs et un toit. D'un point de vue du bâtiment, celui-ci sera constitué d'étages, de pièces et d'objets. Ainsi, différentes informations peuvent être apportées en utilisant différents modèles de données. Enfin, l'hétérogénéité technique [Kut16] concerne les diverses solutions de stockage, langages de requêtage et de description utilisés avec ces données. On peut retrouver par exemple des données sous forme de fichiers ou bases de données. Concernant les fichiers, différents langages (XML, STEP, JSON) peuvent être utilisés pour stocker et transmettre les données. Dans le cas de bases de données, différents langages existent pour requêter et récupérer des données, selon la structure de la base.

De nombreux travaux de recherche existent pour visualiser des données géospatiales hétérogènes. Néanmoins, ceux-ci se concentrent souvent sur une dimension en particulier, soit géométrique [Gai+15 ; Che+18 ; ZCM19 ; La +22], soit sémantique [Kyz+18 ; Alv+21 ; HJS16]. Il existe quelques travaux permettant de gérer de multiples dimensions mais sont spécifiques à un ou deux modèles de données précis [HS21 ; Hij+20 ; Jai20a]. En prenant en compte ces deux problématiques, nous nous posons la question suivante : **P1 : Comment créer une vue unifiée de données sémantiques et géométriques géospatiales hétérogènes dans un même contexte web ?**

Il existe des représentations d'une même entité dans de multiples sources de données géospatiales

Une même entité dispose aujourd'hui d'un ensemble de représentations mises à disposition depuis plusieurs sources de données hétérogènes. Si cette observation existe depuis longtemps dans le monde du Semantic Web, il s'agit d'une problématique

récente dans le monde des données géospatiales. Quelques travaux mettent en avant cette observation pour la formaliser afin de pouvoir référencer différentes représentations d'une même entité [Bec+21 ; MMG20 ; Kne+22 ; OGC16b]. Néanmoins, ce besoin de formalisation et d'identification des différents types de représentations que l'on peut retrouver dans les données géospatiales reste présent. Il est également nécessaire d'identifier ces sources, de décrire les données qu'elles possèdent et les entités qu'elles représentent. Sachant qu'une même source peut contenir plusieurs représentations de plusieurs entités, il est également nécessaire de trouver des solutions pour lier les représentations aux entités qu'elles représentent afin de pouvoir les récupérer depuis leurs sources. Ces représentations nécessitent donc des descriptions pour les référencer, pour interagir avec elles et naviguer dans cet ensemble de représentations d'une même entité. Cette problématique entraîne donc la question suivante : **P2 : Comment peut-on référencer et naviguer dans un ensemble de représentations existantes d'une même entité issues de sources hétérogènes ?**

Gestion du contexte des entités et de l'utilisateur

Le contexte peut être défini comme toute information qui peut être utilisée pour caractériser la situation d'une entité [Abo+99]. L'exploration de l'environnement spatial et/ou fonctionnel d'une entité peut contribuer à sa compréhension. L'exploration de ses relations avec d'autres entités peut contribuer à enrichir son contexte. Cela peut aider à comprendre le système dont elle fait partie et son rôle dans ce système. Les données géospatiales apportent des informations sur les relations entre les entités, qu'il est important de pouvoir remonter à l'utilisateur. D'autre part, le contexte de l'utilisateur est lui aussi important. Il conditionne le choix des entités à représenter et les représentations à utiliser. En effet, un plombier ne s'intéresse qu'aux réseaux d'eaux, les données relatives à la végétation ne l'intéresseront donc que très peu. Également, selon les besoins de l'utilisateur, seules certaines représentations seront pertinentes. Pour localiser une entité, un simple point sur la carte, ou une forme 3D simple peuvent suffire. En revanche, pour comprendre les réseaux d'un bâtiment, l'apport de la 3D est indéniable. Il s'agira donc de trouver des solutions afin de répondre à la question suivante : **P3 : Comment sélectionner puis représenter les entités à visualiser dans un environnement 3D en tenant compte du contexte de l'utilisateur ?**

1.3 Contributions

Dans cette thèse, les contributions scientifiques et techniques suivantes ont été réalisées :

C1 : Méthodologie d'intégration de données géospatiales hétérogènes pour la création de vues contextuelles

Nous proposons une méthodologie permettant de créer des vues contextuelles d'une entité, que nous définissons comme une vue unifiée de représentations d'une entité et son contexte pour aider à sa compréhension. Cette méthodologie nous permet de répondre en partie aux problématiques P1 et P3. Nous proposons de transformer les données géométriques vers un format pivot, dédié aux données 3D, pour permettre de les visualiser dans un même contexte, indépendamment de leur modèle d'origine. Nous proposons également de conserver un lien vers les données sources afin de pouvoir récupérer les données sémantiques associées.

C2 : Méthodologie générique d'intégration des données géométriques et sémantiques de modèles de données géospatiales

Pour proposer une vue complète des données géospatiales, nous proposons une méthodologie générique d'intégration de données géospatiales, en homogénéisant d'une part les données géométriques à l'aide d'un format adapté et d'autre part, en homogénéisant les données sémantiques sous forme de graphe à l'aide d'une méthodologie d'intégration basée modèle pour garantir la conservation de l'ensemble des données sémantiques. Cette méthodologie permet de répondre entièrement à la problématique P1 et pose les bases nécessaires afin de répondre à la problématique P3.

C3 : Formalisation des concepts de *Variant* -instance ou ensemble d'instances représentant la même entité- et de *Variant Identifier* -solution pour décrire la localisation et l'accès à un variant-.

Il existe un réel besoin de formaliser le phénomène de représentations multiples d'une entité dans des sources hétérogènes afin de permettre de naviguer entre elles et répondre à la problématique P2. Nous proposons ainsi le concept de *Variant* qui nous permet de décrire les données qui constituent une représentation d'une entité géospatiale. Pour localiser, accéder et naviguer dans des variants, nous proposons également le concept de *Variant Identifier*.

C4 : Etude des formalisations d'une entité selon les modèles IFC et CityGML et proposition d'une méthode permettant de sélectionner les entités et leurs représentations à visualiser selon le contexte utilisateur.

Une entité peut avoir différentes représentations géométriques, certaines 2D et d'autres 3D. Elles peuvent être constituées de plusieurs entités filles, qui ensemble, constituent l'entité parente : quatre murs et un toit forment un bâtiment. En fonction du contexte utilisateur dans l'espace 3D, ce qu'il regarde, sa position, son intérêt, il est judicieux de lui proposer une représentation pertinente. Afin de répondre à la dernière problématique, P3, nous proposons une étude des formalisations et descriptions existantes d'une entité par les modèles de données IFC et CityGML, et un alignement de ces conceptualisations. Suite à cette étude, nous proposons les concepts de Level of Scale (niveau d'échelle) et Level of Complexity (niveau de complexité) afin d'aider à la navigation multiscalaire, semblable au concept de Niveau de Détail (LoD) utilisé couramment en informatique graphique et adapté pour les données géospatiales.

C5 : Développement d'un outil open-source pour intégrer des données géométriques issues de modèles de données géospatiales hétérogènes.

Pour répondre à la problématique P1, nous proposons un outil open-source pour extraire et transformer les données géométriques depuis leurs sources vers un modèle de donnée homogène, 3DTiles, adapté pour la visualisation, le partage et l'interaction avec des données géométriques. En effet, les modèles de données géospatiales ont été originellement pensés pour répondre à des besoins d'échange et de partage d'informations. La visualisation de l'information géométrique de ces données en temps réel sur le web est donc complexe, car elles nécessitent des solutions, souvent propriétaires, adaptées à leur modèle de données. L'architecture de cet outil a été pensée pour créer des 3DTiles depuis plusieurs modèles de données mais aussi pour être capable de créer, de modifier et de tester de nouvelles méthodes génériques d'organisation des entités dans l'espace et de création de niveaux de détails.

C6 : Applications de gestion et de navigation multiscalaire d'entités et de leurs représentations

Afin de démontrer l'ensemble des contributions scientifiques, des applications web basées sur la librairie UD-Viz permettant de visualiser et naviguer dans l'ensemble des représentations de différentes entités urbaines à différentes échelles sur le web ont été développées. Les architectures techniques de ces applications sont décrites publiquement. Elles mettent en oeuvre différents outils qui ont été développés dans le cadre de la plateforme UD-SV, comme UD-Graph pour transformer des

données sémantiques en graphe, ou bien Py3DTilers pour homogénéiser les données géométriques.

C7 : Reproductibilité et réutilisabilité

Une attention particulière a été portée tout au long de ce travail de recherche afin de garantir la pérennité des savoirs acquis, la reproductibilité et la réutilisabilité des résultats, outils et démonstrations proposés au sein de la plateforme UD-SV [Sam+23; Col+23], à l'aide de documentations et de technologies de conteneurisation, comme Docker.

C8 : Création d'un outil dans la suite logicielle de Berger-Levrault (BL-Viz) basé sur les contributions C1 et C5

Enfin, les contributions techniques et scientifiques sont en cours d'appropriation et de développement au sein de Carl|Berger-Levrault, au sein d'un produit nommé BL-Viz4D. Celui-ci permet de naviguer depuis un environnement géospatiale 3D entre des maquettes CIM et BIM, des données de la GMAO classique et des données SIG. Grâce à un lien entre celles-ci, il est possible d'effectuer des opérations de maintenance (déclarer un désordre, déclarer une intervention ...) sur une entité à l'aide de différentes représentations : celle sémantique dans le logiciel de GMAO, celle 2D issue de SIG et celles 3D issue du BIM et du CIM.

1.4 Plan

Suite à cette introduction, ce travail de recherche est présenté de la façon suivante :

Le chapitre 2 présente les standards et méthodes existants pour modéliser et visualiser des données géospatiales. Puis, différents travaux sur l'intégration de données géospatiales hétérogènes dans un même contexte seront exposés.

Le chapitre 3 présente l'approche C1 étudiée pour intégrer des données géométriques, et son utilisation dans la méthodologie C2 de création de vues contextuelles de données.

Le chapitre 4 présente notre troisième contribution C3, à savoir une méthodologie générique d'intégration des données géométriques et sémantiques de données géospatiales.

Le chapitre 5 présente la contribution C4, une formalisation du concept de *Variant-instance* ou ensemble d'instances représentant la même entité-, l'identification de

quelques types de variants souvent rencontrés, et du concept de *Variant Identifier*-solution pour décrire la localisation et l'accès à un variant-.

Le chapitre 6 présente une étude des formalisations d'une entité par différents modèles de données géospatiales ainsi qu'une métrique permettant de décrire la complexité d'une représentation géométrique et du niveau d'échelle d'une entité pour permettre de sélectionner automatiquement les entités à visualiser selon le contexte utilisateur. Il s'agit de la contribution C5.

Le chapitre 7 présente la méthodologie de construction d'une solution scientifique robuste, documentée, réutilisable et reproductible, appliquée au sein de la plateforme Urban Data Services and Visualization (UD-SV) du Liris, qui nous a amené aux contributions techniques C6 et C7 réalisées durant ces travaux de thèse. Nous présentons également le produit en cours de développement au sein de Carl|Berger-Levrault basé sur les résultats de cette thèse.

Le chapitre 8 conclut ce manuscrit avec une synthèse des contributions et les travaux de recherche futurs associés à ce projet.

Etat de l'art

Nous présentons dans la section 2.1 les concepts liés à la modélisation, utilisés dans les travaux de recherches et de standardisation des données géospatiales. Puis nous présentons une vue d'ensemble des standards et méthodes pour modéliser et visualiser des données géospatiales 3D hétérogènes dans un même contexte web. Cette étude a été réalisée afin de répondre à notre problématique P1 et a pour objectif d'aider à comprendre comment sont définies et structurées les données sémantiques et géométriques qui constituent des données géospatiales issues de sources hétérogènes. La section 2.1.1 présente différents concepts utilisés pour modéliser des données géospatiales 3D et les standards existants ; la section 2.1.2 présente des solutions pour visualiser les données issues de ces standards sur le web ; et la section 2.1.3 résume ces études.

Dans la section 2.2.1, nous présentons les quelques solutions existantes permettant d'identifier et de naviguer dans des représentations hétérogènes d'une même entité issues de sources différentes ; la section 2.2.2 détaille et compare les notions de niveaux de détails que l'on peut retrouver dans différents modèles de données pour représenter des entités à différentes échelles ; la section 2.2.3 résume ces propositions, met en avant leurs limitations et donc, les besoins sous-tendus afin de pouvoir proposer à l'utilisateur une navigation multiscalaire entre les représentations hétérogènes d'une même entité.

2.1 Modélisation et intégration de données géospatiales

2.1.1 Modélisation de données géospatiales

Les jumeaux numériques urbains et les applications de smart city [Bat18b ; Jul+18 ; SH20] peuvent aider à visualiser et simuler des événements et des évolutions dans le temps, réels ou imaginaires, du paysage urbain [Bil+15 ; CK19 ; JSG20 ; SSG20]. Plusieurs standards et outils ont été définis pour construire ces applications,

principalement par deux organisations : l'International Organisation for Standardization¹ (notamment l'ISO/TC 211, qui est spécialisé dans les données géospatiales²) et l'Open Geospatial Consortium³ (OGC). Ils définissent les données géospatiales en tant que "propriétés qui peuvent inclure toute information spatiale -regroupant localisation géographique et topologie- de *features*, ainsi que les relations entre elles, et des informations descriptives"[OGC24]. Une *feature* étant définie comme une "abstraction d'un phénomène du monde réel" [ISO14a]. Les attributs principaux associés à une feature peuvent être thématiques, temporels [ISO19b] ou spatiaux [ISO19a] (en rapport à la localisation, la topologie ou l'aspect physique de l'entité)[ISO15b]. Lorsque ces données sont interprétées et mises en contexte par les humains, on parle alors d'*informations géospatiales*[OGC24].

Originellement utilisés pour abstraire et décomposer des problèmes et systèmes du monde réel, des modèles de données ont été créés afin de structurer et donner du sens aux données géospatiales représentant des phénomènes du monde réel. Le standard ISO 19101 de modélisation d'informations géographiques définit le terme **Universe of Discourse** (fr : Univers du Discours) -ou "domain of discourse" ou "domain of interpretation"[GOS09]- comme "vision du monde réel ou hypothétique qui inclut tout ce qui est intéressant" [ISO14b] pour référer à ce qui peut être modélisé. Les modèles de données définissent la structure et la sémantique des instances de données qui sont utilisées pour représenter des phénomènes du monde réel 2.1.

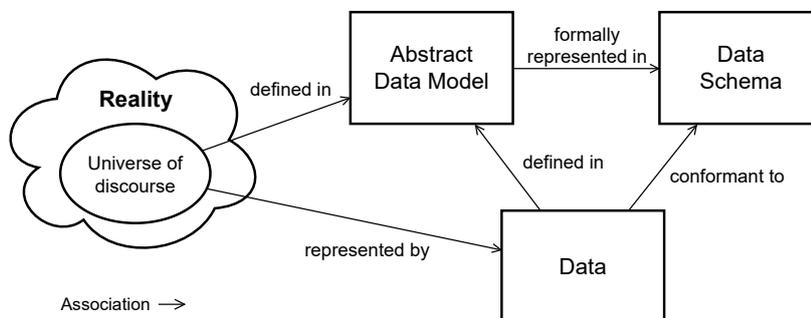


Figure 2.1 : Les relations entre les données, les schémas(ou langages), les modèles de données abstraits et un univers du discours (adapted from [Kut16]).

Les modèles de données peuvent décrire ou définir les données à différents niveaux d'abstraction : conceptuel, logique et physique [Avi91]. Au niveau conceptuel, les données sont définies de manière abstraite sans tenir compte du matériel physique,

1. <https://www.iso.org/home.html>
 2. <https://www.iso.org/committee/54904.html>
 3. <https://www.ogc.org/>

des formats de données ou des architectures de bases de données. Le niveau logique utilise des modèles de données *logiques* moins abstraits qui tiennent compte des systèmes d'information dans lesquels les données sont stockées (par exemple, une base de données relationnelles) [Avi91]. Au niveau physique, les schémas, ou langages de description, peuvent être utilisés pour définir des modèles de données *physiques* qui tiennent compte de formats de données ou d'architectures de bases de données spécifiques (par exemple, l'organisation des tables). Un même modèle de données logique peut donc être exprimé sous différents modèles de données physiques grâce à différents langages ou formats, qui suivent différents paradigmes. Les langages UML⁴ (Unified Modeling Language) et EXPRESS [ISO04] sont "orienté objet"; XML (eXtensible Markup Language) est un langage de description de données web, basé sur le concept d'*élément* hiérarchique; RDF (Resource Description Framework) est basé sur le formalisme de graphe orienté, constitué de triplets "sujet", "prédicat", "objet".

Dans le domaine du City Information Modelling (CIM), l'OGC a proposé le modèle logique CityGML 3.0⁵ [OGC18a; KGP05; KCK20] décrit à l'aide du langage UML. Il est composé d'un modèle géométrique et d'un modèle sémantique, et est utilisé pour décrire l'apparence, la sémantique et la topologie d'objets urbains tels que des bâtiments, des ponts, des routes ... [GP12], comme illustrés dans la figure 2.2. En utilisant ce modèle logique, l'OGC propose aussi les modèles physiques de CityGML formalisés avec GML[KCK20], une syntaxe XML, avec JSON[OGC18b], et différents travaux proposent des traductions de ces modèles physiques vers le langage RDF/OWL[UJS21; Vin23]. CityGML a été mondialement accepté, et plus d'une vingtaine de pays mettent à disposition des modèles numériques 3D de leur ville [WSW22; LSB23].

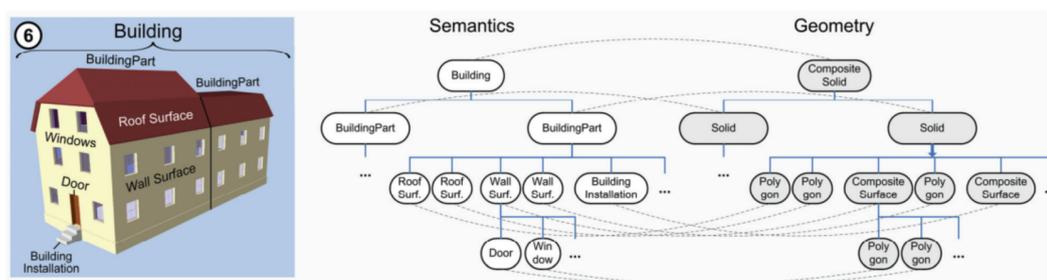


Figure 2.2 : Représentation des données sémantiques et géométriques d'une maquette numérique CityGML 3.0[OGC18a]

4. <https://www.omg.org/spec/UML/2.5.1>

5. <https://docs.ogc.org/is/20-010/20-010.html>

Dans le domaine du Building Information Modelling (BIM), on retrouve le standard Industry Foundation Classes (IFC) [ISO18b ; Mil12] proposé par le consortium BuildingSmart⁶ pour modéliser des infrastructures, comme des bâtiments ou des ponts, et l'ensemble des objets qui composent leur intérieur : murs, réseaux, fenêtres, ... Le modèle logique IFC est décrit à l'aide du langage EXPRESS, et formalisé en modèle physique avec le format STEP (Standard for the Exchange of Product model data) [HRB20], mais aussi XML, RDF⁷. A cette échelle, on retrouve également le standard IndoorGML [OGC18c], également formalisé à l'aide de GML, qui se concentre sur la modélisation des espaces pour la navigation.

Les données géospatiales sont composées de données sémantiques, mais aussi de données géométriques 3D, qui nécessitent leur propre modélisation pour être utilisées, celles-ci sont représentées dans la figure 2.3.

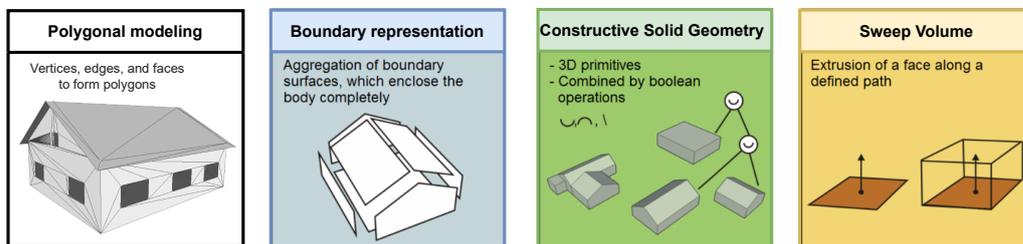


Figure 2.3 : Les différentes modélisations possibles des données géométriques, adapté de [Don+16] avec l'ajout de "polygonal modeling"

Modélisation polygonale : La modélisation polygonale est une technique fondamentale utilisée dans la modélisation 3D. Elle permet de former des maillages à partir d'ensemble de points, arêtes et faces (les polygones) représentant la surface ou le volume d'un objet 3D et sa topologie. Les faces sont le plus souvent modélisées sous forme de triangles.

Modélisation par courbe et surface paramétrique : Les techniques de modélisation par surface, comme les surfaces de Béziérs ou les Non-Uniform Rational Basis Splines (NURBS), permettent de définir des courbes et surfaces à l'aide de point de contrôle.

Boundary Representation (B-Rep) : Cette méthode consiste à créer la forme de l'objet à partir d'un ensemble de surfaces paramétriques qui peut être qualifié de

6. <https://www.buildingsmart.org/standards/bsi-standards/industry-foundation-classes/>

7. <https://technical.buildingsmart.org/standards/ifc/ifc-formats>

modélisation surfacique. Elle est principalement utilisée en architecture ou en CAO (Computer-Aided Design).

Constructive Solid Geometry (CSG) : La modélisation volumique CSG consiste à combiner des formes simples (cube, sphère, pyramide, ...) à l'aide d'opérations géométriques booléennes (union, intersection, soustraction) pour représenter des objets solides. Cette technique a l'avantage de pouvoir simplement créer des représentations d'objets et est couramment utilisée en CAO.

Volume extrudé (Sweep Volume) : Cette technique permet de représenter un volume à partir d'une surface extrudée le long d'une droite. Elle est notamment utilisée par les SIG pour représenter les bâtiments en 2.5D, en extrudant les surfaces des cadastres horizontalement à l'aide d'informations concernant la hauteur des bâtiments.

Nuage de points (Point cloud) : Un nuage de points est un ensemble de points 3D auxquels sont associés diverses informations, comme la couleur. Il existe diverses méthodes d'acquisitions de ces nuages de points : Airborne Laser Scanning (ALS), Terrestrial Laser Scanning (TLS), Mobile LiDAR Systems (MLS), Global Navigation Satellite Systems (GNSS), et par photogrammétrie 3D aérienne et terrestre. De nombreux travaux s'intéressent à reconnaître des objets, comme des bâtiments ou de la végétation, dans ces nuages pour classifier les points qui les constituent [Zeg21]. A l'aide de ces classifications, il existe des travaux permettant de créer des données BIM [Lam+22 ; Soi+20] et CIM [PID12 ; JR18 ; Bei+21] à partir de nuage de points.

2.1.2 Intégration de données géospatiales

L'intégration de données peut être définie comme "le processus qui consiste à combiner des données provenant de différentes sources afin de fournir aux utilisateurs une vue unifiée des données." [Tra+16]. L'intégration de données BIM-GIS peut être appliquée pour différents cas d'usage [HRB20][Liu+17][Fos+15b], comme par exemple l'analyse énergétique, l'analyse d'ensoleillement, la planification ou encore l'estimation des dommages après désastre naturel. Selon KOLBE et al. [Kol+20], comme les applications géospatiales requièrent des données produites par diverses parties prenantes [NK21] à propos du même objet, la capacité d'intégrer diverses sources de données pourrait aider à trouver les informations manquantes.

Les approches d'intégration de données BIM-GIS peuvent être classifiées dans différentes catégories [BBK20] (figure 2.4) : la **conversion** d'un modèle de données vers un autre ; l'**extension** d'un modèle de données pour structurer de nouvelles données provenant d'autres modèles ; l'utilisation de méthodes et technologies issues

du domaine du **web sémantique et du linked data** pour établir des liens entre différents modèles de données et instances de données ; et la **fusion** de deux modèles de données pour en créer un nouveau, contenant l'ensemble des informations nécessaires. [Fos+15a] et [Liu+17] ajoutent une catégorie pour rassembler les méthodes basées sur des **standards web**.

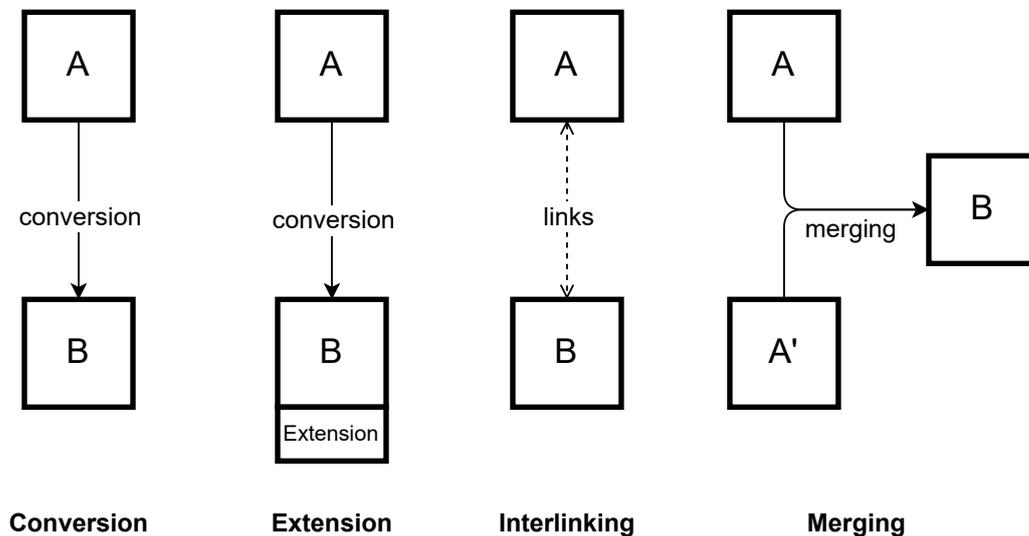


Figure 2.4 : Les différentes approches pour intégrer des données BIM-GIS, par [BBK20]

Conversion :

Les travaux de conversion de modèles de données géospatiales s'intéressent principalement à la conversion de données IFC vers CityGML [Hij+09 ; Don13]. Ils reposent sur des propositions d'alignement de ces modèles [Col+20], qui peuvent être définis à l'aide de grammaire sous forme de graphes [STB18]. [DCA16] propose une ontologie qui met en correspondance différents concepts d'IFC et de CityGML pour permettre une conversion bidirectionnelle. En utilisant une première transformation d'IFC vers CityGML, puis de CityGML vers IndoorGML, [KDK15] permet la génération de graphe de navigation en intérieur selon différents moyens de locomotions depuis des données BIM.

Extension :

Convertir des données IFC en CityGML ou inversement implique de la perte de données, car les modèles de données décrivent les objets avec différentes précisions. Pour pallier ce problème, [LB11] propose une extension au modèle CityGML, nommé GeoBIM, pour disposer de sémantique BIM dans un contexte CIM. Les données IFC sont importées dans une base open-source nommée BIMServer [BBL10], et peuvent

être alors exportées en format CityGML. [MN14] propose d'étendre le BIM à l'aide d'une extension nommée Urban Information Modelling, apportant des concepts spatiaux et temporels.

Fusion :

D'autres travaux proposent de créer de nouveaux modèles de données pour représenter l'ensemble des objets de différents modèles. [EÖH12; YZD21] ont notamment proposé le Unified Building Model (UBM) pour combiner l'ensemble des possibilités apporter par IFC et CityGML dans un seul modèle de données. Plus récemment, le modèle de données GeoBIM [Mor+21] a également été proposé et utilisé pour construire un jumeau numérique et fusionner des données issues du BIM et du CIM.

Standards web :

Les modèles de données géospatiales ont été créés pour faciliter le stockage et l'échange d'information entre différentes applications. Cependant, ils n'ont pas été pensés pour la visualisation des données géométriques, qui font partie intégrante de ces modèles [Kol09]. L'une des solutions les plus populaires pour intégrer les données géométriques de ces modèles est de les convertir vers des formats spécifiques aux données géométriques 3D afin de les visualiser sur le web.

Le format 3DTiles est un standard communautaire, proposé par Cesium⁸ et par l'OGC⁹ [OGC19]. Il a été spécifiquement conçu pour prendre en charge la visualisation massive de contenu géospatial en 3D, tout en tenant compte des aspects liés au streaming et au rendu. Très semblable au format I3S [OGC17], proposé par ESRI et l'OGC en tant "Community Standard", 3DTiles rencontre un franc succès et devient populaire, comme le démontre le nouveau service de streaming de 3DTiles proposé par Google¹⁰. [JSG20] a d'ailleurs proposé un modèle conceptuel, Gen3DCity permettant de généraliser ces deux modèles. Ce standard décrit comment créer des arbres de tuiles (*tiles*) 3D, appelés *tileset*, illustré dans la figure 2.5, qui donne une vue d'ensemble d'un *tileset*. Cet arbre permet une organisation spatiale des tuiles optimisée pour le rendu des objets spatiaux, notamment en supportant différentes méthodes de tuilage (arbre K-d, octree...) mais aussi le concept de niveau de détail hiérarchique.

8. <https://github.com/CesiumGS/3d-tiles>

9. <https://www.ogc.org/standard/3dtiles>

10. www.developers.google.com/maps/documentation/tile/3d-tiles?hl=fr

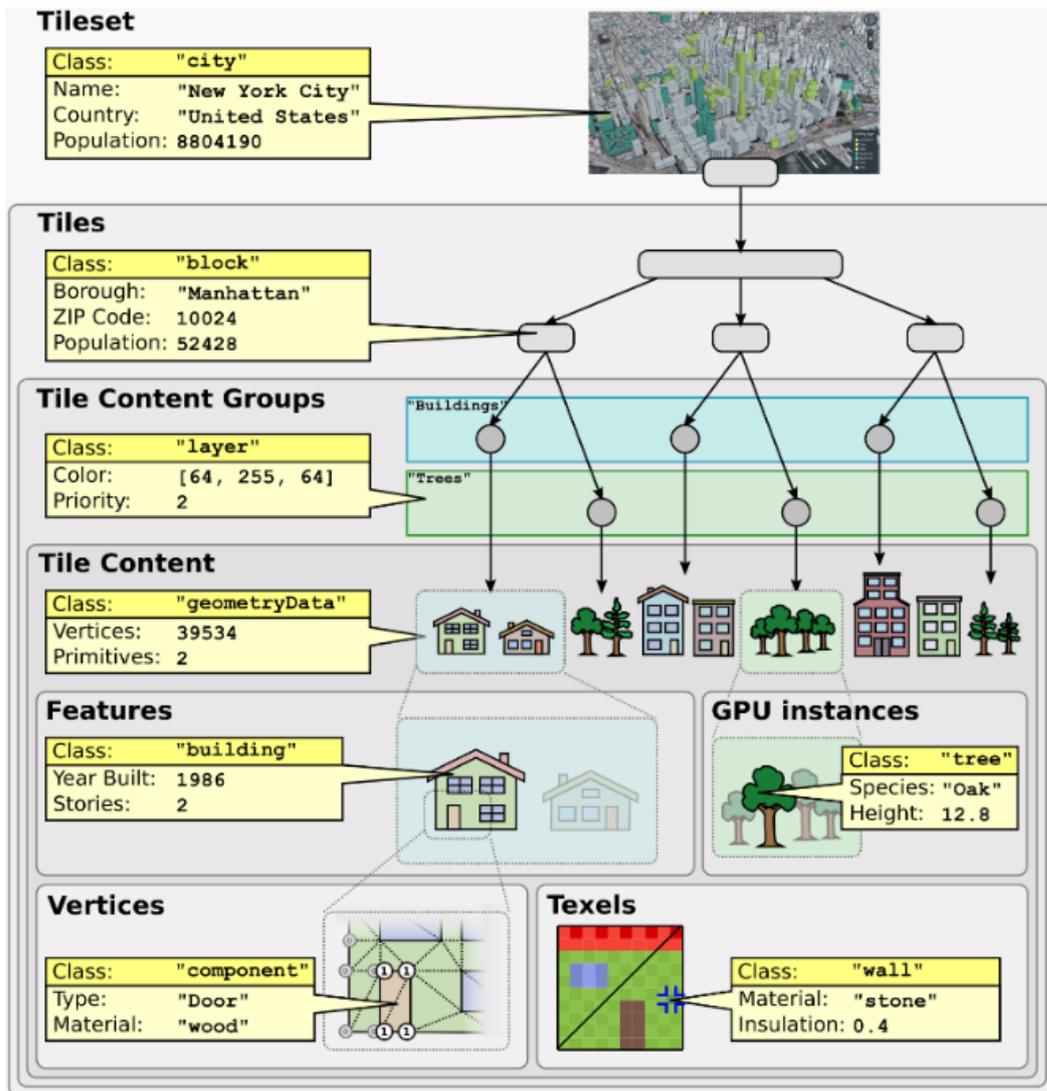


Figure 2.5 : Vue d'ensemble d'un tileset 3DTiles [OGC19]

Dans le modèle 3DTiles, les tuiles peuvent avoir différents formats :

- Batched 3D Model (B3DM) : Modèles 3D hétérogènes. Par exemple, des terrains et surfaces texturés, des extérieurs ou intérieurs de bâtiments en 3D.
- Instanced 3D Model (I3DM) : Instances de modèles 3D. Par exemple, des arbres, des moulins à vent, des boulons.
- Nuage de points.

Les formats B3DM et I3DM reposent sur le format glTF¹¹, qui est un format géométrique décrit par Khronos et qui vise à faciliter la diffusion et le rendu de modèles 3D

11. <https://github.com/KhronosGroup/glTF>

sur le web. Chaque tuile contient un ensemble de features sous forme de géométries 3D. Ces formats décrivent des mécanismes, nommés batch table et feature table, permettant d'associer de la donnée sémantique à chaque feature (figure 2.6).

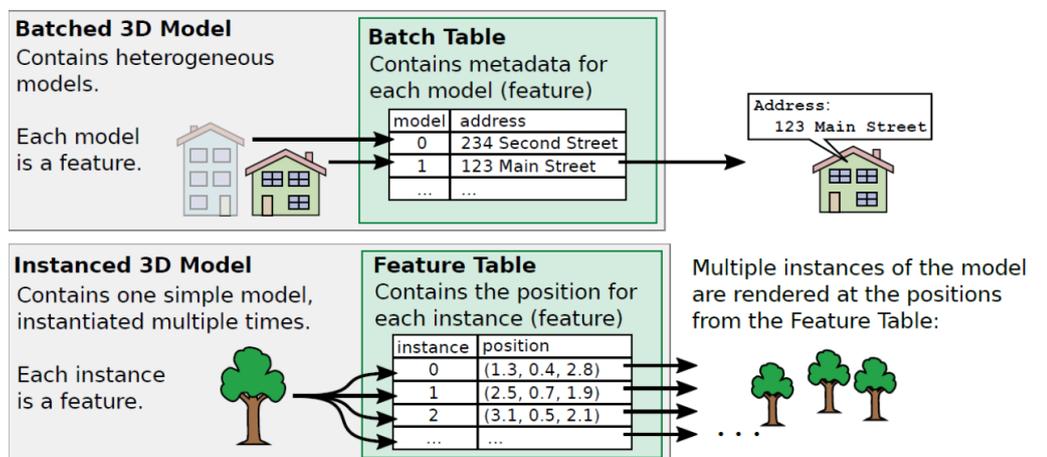


Figure 2.6 : Illustration du fonctionnement des feature table et batch table des tuiles au format B3DM et I3DM[OGC19]

Différentes extensions sont disponibles¹² afin d'étendre la spécification 3DTiles. On peut par exemple retrouver une extension permettant de décrire la hiérarchie entre les features¹³, ou de pouvoir compresser les données 3D grâce à la méthode Draco¹⁴. [JSG20] a notamment proposé une extension permettant de modéliser les évolutions des features dans le temps. Dans sa version 1.1, publié en 2022, les formats B3DM et I3DM sont abandonnés au profit de l'utilisation du format glTF et d'extensions de ce format, notamment pour conserver de la donnée sémantique, associée à la géométrie, comme le permettent les batch tables.

Conversion vers des standards web :

[XZX16; Xu+20] propose de **transformer des données IFC en .Obj** (format populaire pour modéliser des objets 3D) afin de les visualiser sur le web en utilisant la bibliothèque graphique WebGL. Les données sémantiques additionnelles sont conservées dans des fichiers JSON. [Che+18] propose une solution constituée de composants open-source permettant de transformer un fichier **IFC en 3DTiles** en passant par des fichiers .Obj. [ZJ21] propose une extension au logiciel Autodesk de Revit afin d'exporter un **IFC en 3DTiles**. Des études montrent qu'il est aussi possible de visualiser

12. <https://github.com/CesiumGS/3d-tiles/tree/main/extensions>
 13. https://github.com/CesiumGS/3d-tiles/tree/main/extensions/3DTILES_batch_table_hierarchy
 14. https://github.com/CesiumGS/3d-tiles/tree/main/extensions/3DTILES_draco_point_compression

des données **CityGML sur le web à l'aide de glTF** [SBN16] et de **3DTiles** [MBL20 ; GPG18]. D'autres études ont montré qu'il était possible d'utiliser des formats SIG 2D, comme **shapefile**, pour représenter les géométries 3D d'objets **CityGML** [Zhu+19a ; Zhu+19b] et **IFC** [ZW21].

La transformation des données géométriques 3D vers un format dédié permet d'appliquer des solutions existantes ou innovantes pour faciliter et améliorer la visualisation de celles-ci. [Lu+19] et [ZCC21] montre que le partitionnement spatial des objets IFC joue un rôle important pour l'optimisation du chargement des données dans la vue. [GPG18] propose différentes règles de chargement des objets 3D afin de prendre en compte des caractéristiques géométriques ou sémantiques sous-jacentes au feature. Par exemple, les bâtiments ayant les plus gros volumes ou présentant le plus d'intérêt pour l'utilisateur peuvent être chargés en premier.

Web sémantique et linked data :

La transformation des données géométriques géospatiales vers des modèles de données géométriques 3D semble particulièrement efficace pour les visualiser. Certains travaux proposent alors de conserver les données sémantiques dans des bases de données et de créer un lien entre les géométries transformées et les données sémantiques. [Yao+18] mets à disposition de la communauté une base de données implémentant le modèle CityGML, et permet des exports glTF des géométries. [Li+16] propose une base de données basée sur Oracle et [BBL10] offre une base de données open-source, BIMServer, pour héberger des données IFC. Ce dernier propose également des exports glTF. Enfin, [Hij+20] utilise 3DCityDB et BIMServer pour créer un lien entre des données BIM et CIM. Pour accéder à des données multimédias depuis des données CityGML, une base nommée MultimediaDB permet de géolocaliser des documents et de les associer à des instances CityGML [Jai+21].

Les solutions reposantes sur des approches issues du web sémantique et du linked data semblent les plus efficaces et les plus flexibles (figure 2.7)[Liu+17].

Les étapes clés d'une intégration à l'aide de technologies du web sémantique sont l'analyse des données, le développement d'une ou plusieurs ontologies pour structurer les données, transformer les données, puis de lier les données des différentes sources entre elles [Rad+15]. [HJS16] propose une plateforme implémentant l'ensemble de ces étapes, ainsi qu'une ontologie unique pour représenter les données BIM et GIS. Dans le contexte de l'utilisation de données BIM avec des données géospatiales (GeoBIM), [Hua+20] démontre la possibilité d'intégrer des données IFC et CityGML dans des graphes de connaissances en utilisant l'ontologie BOT pour simplifier les représentations et requêtes de données BIM, et GeoSPARQL pour les

données CityGML, puis en créant des liens entre les différentes données. Afin de permettre et faciliter l'utilisation de données géospatiales transformées en graphe, des extensions au langage SparQL ont été proposées pour pouvoir exploiter des données SIG, comme GeoSparql [CH22], et des données BIM, comme BimSPARQL [ZBV18].

Integration Methods	Effectiveness	Extensibility	Effort	Flexibility
New standards and models	case by case	case by case	case by case	case by case
Conversion, translation and extension of existing standards (manual)	medium	high	high	medium
Conversion, translation and extension of existing standards (semi-automatic)	medium	medium	medium	medium
Semantic web technologies	high	high	high	medium
Services-based methods	high	low	high	low
Application focused methods	case by case	low	low	low

Figure 2.7 : Comparaison des différentes méthodes d'intégrations de données BIM-GIS, selon [Liu+17]

L'utilisation de telles approches nécessite des ontologies globalement acceptées pour décrire les modèles conceptuels de manière uniforme. On retrouve des propositions d'ontologies de données BIM[KIH16], bien que le standard IFC dispose maintenant d'ontologies officielles nommées ifcOWL¹⁵. [Vin23] a étudié et proposé deux approches pour créer une ontologie CityGML 2.0 ou 3.0, une depuis le schéma XML de CityGML, la seconde, plus contrainte, est basée sur un standard de transformation d'UML vers OWL[ISO15a].

Le web sémantique a une capacité naturelle à intégrer des données sémantiques provenant de différentes sources. [Wag+20] étudie et compare différentes approches (figure 2.8) pour intégrer les données géométriques à l'aide de technologies du web sémantique ou non pour les lier et les héberger. L'utilisation de technologies dédiées aux données géométriques 3D pour représenter des géométries 3D à l'aide de graphes ou non implique une perte d'expressivité sémantique puisque l'on perd alors logiquement la capacité d'interroger ces données géométriques à l'aide de technologies du Web Sémantique. On peut remarquer un gain de flexibilité (la capacité à choisir des modèles et méthodes de création de lien), de simplicité (problèmes techniques qui doivent être résolus dans le cadre de cette approche), de support (le degré de soutien de la communauté dont bénéficie déjà une approche), tout en gardant une forte portabilité (la capacité à partager et lier la géométrie avec d'autres données RDF).

15. <https://technical.buildingsmart.org/standards/ifc/ifc-formats/ifcowl>

	linking	storage	content
Approach 1: RDF-based Geometry Descriptions	Semantic Web technologies		
Approach 2: JSON-LD for Web Geometry	Semantic Web technologies	Semantic Web & JSON technologies	
Approach 3: Non-RDF Geometry as RDF Literals	Semantic Web technologies		other technologies
Approach 4: Linking to Non-RDF Geometry Files	Semantic Web technologies	other technologies	

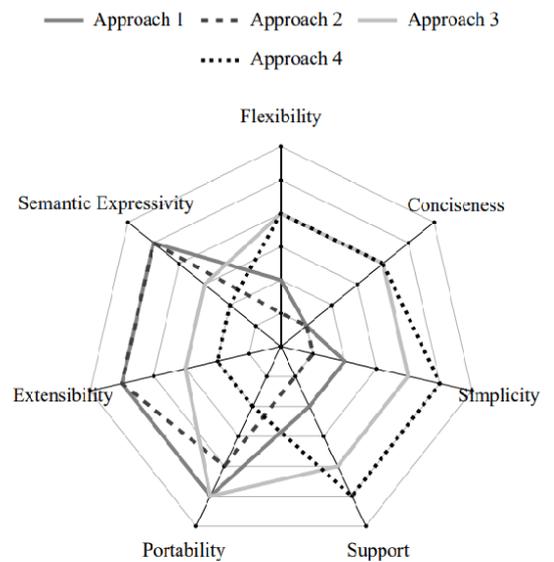


Figure 2.8 : Etude des différentes approches pour intégrer des données géométriques avec des technologies du web sémantique, selon [Wag+20]

2.1.3 Discussion

Cette première partie illustre les efforts récents pour modéliser des données géospatiales à l'aide de standards et les visualiser dans un même contexte. CityGML pour les données CIM, ainsi que qu'IFC pour les données BIM, sont les standards les plus communs pour structurer et échanger des données, mais n'ont pas été pensés pour visualiser les données géométriques. IFC et CityGML utilisent des modèles différents pour structurer les données sémantiques, mais aussi pour structurer les données géométriques [LB11]. Comme illustré dans la figure 2.9, IFC représente les géométries comme des volumes et CityGML comme des plans, ce qui pose des problèmes pour intégrer efficacement ce type de donnée.

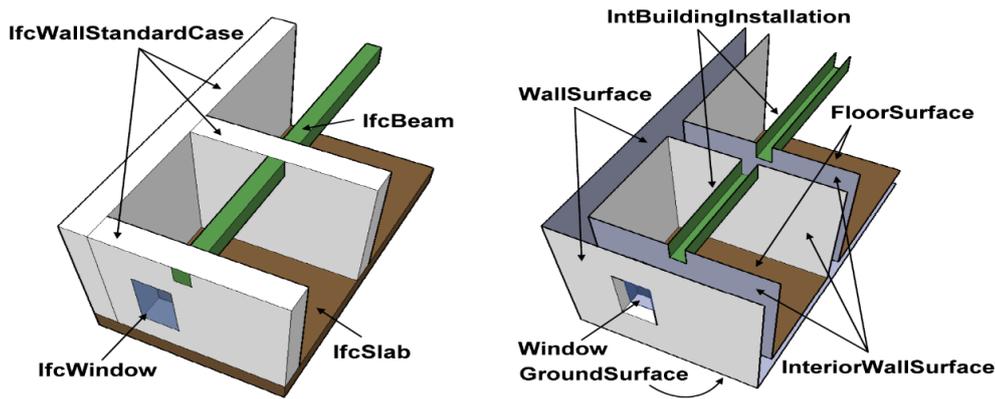


Figure 2.9 : Différence de modélisation des géométries et de conceptualisation des entités entre IFC et CityGML[LB11]

Nous proposons dans les chapitres 3 et 4 des solutions pour intégrer des données géométriques hétérogènes sur le web, en utilisant une conversion de ces modèles vers un format pivot, 3DTiles. Les solutions de conversion vers des standards web sont limités aux données géométriques et ne permettent pas de conserver l'ensemble des données sémantiques associées. Lors de la conversion des données géométriques, nous proposons donc des solutions afin de conserver un lien entre elles et les données sémantiques originellement présentes, afin de pouvoir accéder à l'ensemble des informations présentes dans les sources. Dans un premier temps, ces données sémantiques sont conservées dans des bases de données adaptées à leurs modèles. Puis, dans un second temps, nous proposons une solution pour homogénéiser ces données sémantiques en utilisant des technologies du web sémantique. Ces travaux permettent de visualiser dans un même contexte web l'ensemble des données géométriques et sémantiques issues de données structurées par des modèles hétérogènes.

2.2 Navigation multiscalaire dans des représentations hétérogènes d'une même entité

2.2.1 Navigation et multiple représentations

Face à la croissance de la génération de données géospatiales, on retrouve de multiples représentations d'une entité géospatiale dans diverses sources de données, produites par différents acteurs. Un défi majeur consiste à développer des solutions

pour identifier les entités décrites et les relier à leur représentation. Pour cela, il faut définir comment les liens entre des représentations hétérogènes d'une entité et cette entité peuvent être exprimés, créés et gérés.

En proposant l'extension dynamizer à CityGML, désormais adoptée dans la version 3.0, [CK16] permet de créer des liens entre des données CIM et des données issues de capteurs, existant dans des bases de données externes aux données CIM. [Hij+20] ont étendu la base de données CityGML 3DCityDB pour pouvoir créer des liens vers des instances de données BIM vivant dans une base BIMServer. Ces solutions permettent de naviguer entre différentes représentations d'une même entité, mais sont néanmoins limitées aux modèles de données qu'elles utilisent.

Pour répondre à la nécessité de gérer l'hétérogénéité des données et d'accéder aux données à partir d'une source unique, un modèle de métadonnées a été proposé [MMG20 ; Kne+22] pour créer un catalogue des ressources de données disponibles, telles que les datasets, les services en ligne, les applications en ligne, etc. En utilisant trois types de relations sémantiques prédéfinies : *links_to*, *depends_on* et *child_of*, il est possible de créer des relations entre deux ressources. De même, l'OGC [OGC16b] a proposé un standard pour décrire un catalogue qui vise à soutenir la capacité à rechercher des collections d'informations descriptives (métadonnées) pour les données et les services. Des solutions web propriétaires, telles que Autodesk BIM360¹⁶ et ArcGIS GeoBIM¹⁷, permettent de naviguer entre les représentations BIM, SIG et documentaires.

Le standard GeoPackage¹⁸ proposé par l'OGC permet d'héberger différentes représentations 2D sur un même territoire, mais ne permet pas d'identifier les entités représentées à travers ces représentations. Suite à l'élaboration du standard 3DPortal [OGC16a ; KCG18] pour décrire et délivrer du contenu géospatial 3D, l'OGC est actuellement en train de travailler à l'élaboration d'une API qui permet de naviguer dans différentes représentations géospatiales d'une même entité. [MTS20 ; MTL20]. Ces travaux décrivent le modèle logique 3D-Container pour représenter une entité et pouvoir relier ses représentations. Un 3D-Container, illustré dans la figure 2.10, représente une entité avec une position et un volume délimité distinct. Il peut contenir des jeux de données de maquette 3D qui sont pertinents pour cette entité. Il peut inclure ou référencer d'autres 3D-Container dont les volumes sont entièrement contenus par le volume du 3D-Container parent. Dans cet exemple, l'entité *Amérique du Nord* est représentée par un 3D-Container. Il contient deux autres entités, *Montreal* et *New York City*. *New York City* est relié à ses différentes

16. <https://www.autodesk.com/bim-360/>

17. <https://www.esri.com/en-us/arcgis/products/arcgis-geobim/overview>

18. <https://www.ogc.org/standard/geopackage/>

représentations CityGML, 3D Tiles et I3S. [San+22] utilise cette API pour créer des applications de jumeaux numériques à l'aide de clients web ou de moteurs de jeu en utilisant des sources de données hétérogènes. Dans ces travaux, on peut alors naviguer entre les données CIM d'une ville et les données IoT associées à chaque bâtiment.

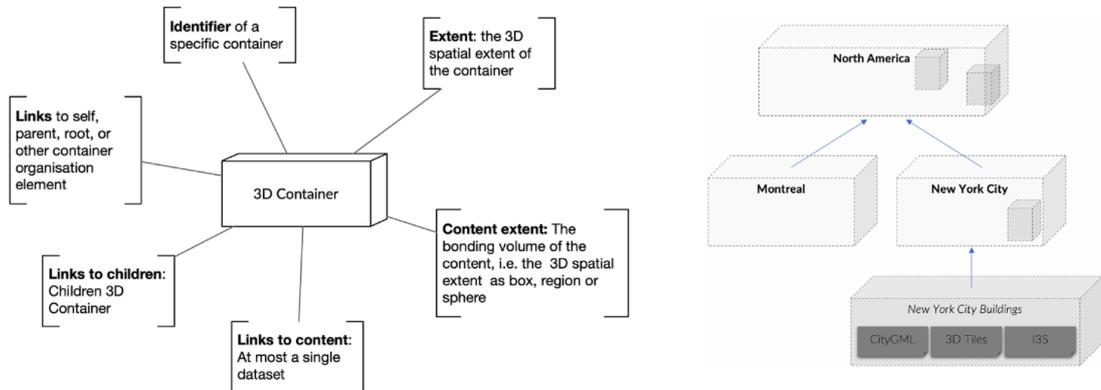


Figure 2.10 : Attributs d'un 3D-Container selon le modèle logique 3D-Container et exemple d'une collection de 3D-Container[MTS20]

2.2.2 Navigation multiscalaire

En infographie, le niveau de détail (LOD) fait référence à la complexité d'un modèle 3D. Le LOD peut être réduit lorsque le modèle 3D s'éloigne de l'observateur ou en fonction d'autres paramètres tels que l'importance de l'objet ou de sa position. La réduction de la qualité visuelle du modèle 3D passe souvent inaperçue en raison du faible effet sur l'apparence de l'objet lorsqu'il est éloigné. La création de niveaux de détail d'un même objet a pour objectif de fluidifier et d'améliorer le rendu de la scène 3D. [CLL16] montre qu'il est possible de transférer les données géométriques de modèles 3D urbains sous forme de maillage progressif, qui se raffine au fur et à mesure que les données se téléchargent sur le client web.

Dans les SIG, les données 2D sont visualisées à l'aide de tuiles représentant des zones géographiques. La taille de ces tuiles sont définies selon différents niveaux de zooms. À faible niveau de zoom, un petit ensemble de tuiles couvre une grande zone géographique. À des niveaux de zoom plus élevés, un plus grand nombre de tuiles couvre une zone géographique plus petite. De nombreux SIG adoptent les niveaux de zooms définis par Mapbox¹⁹ ou OpenStreetMap²⁰, illustrée dans la

19. <https://docs.mapbox.com/help/glossary/zoom-level>

20. https://wiki.openstreetmap.org/wiki/Zoom_levels

figure 2.11. La colonne "Tile width" donne la largeur de la carte en degrés, pour une carte large de 256 pixels à ce niveau de zoom. Les valeurs pour "m / pixel" sont calculées pour un rayon de la Terre de 6372,7982 km. La colonne "Scale" n'est qu'une approximation et se base sur les distances à l'Équateur. Ces tuiles peuvent être servies sous forme d'images (JPEG, PNG, ...), au format raster (matrice de pixel organisée en grille), à l'aide des standards de l'OGC WMS²¹ et WMTS²². Ces tuiles peuvent être également servies sous forme vectorielle (points, lignes et polygones) à l'aide de données JSON ou GeoJSON²³ grâce au standard WFS²⁴. Dans ce format, il est possible de représenter des attributs géométriques et sémantiques de features.

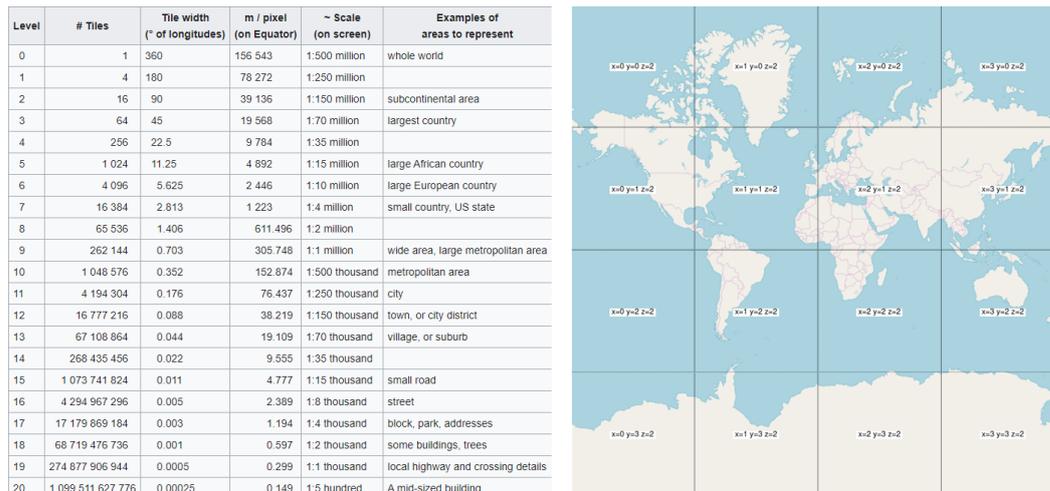


Figure 2.11 : Niveaux de zoom SIG utilisés par OpenStreetMap et un exemple de tuiles au niveau d'échelle 2

Le niveau de zoom ne peut être simplement utilisé pour caractériser le niveau de détail d'une tuile, car les données qu'elles contiennent ne sont pas simplement destinées à la visualisation cartographique. Le niveau de détail d'une base de données géospatiales est définie selon [Rua02] par différents éléments : le modèle conceptuel sous-jacent, la résolution sémantique (la quantité de données attributaires associées à une feature), la résolution, précision et granularité géométrique. Ainsi, pour chaque niveau d'échelle, il est nécessaire de définir ce qu'il est nécessaire de visualiser (quelles sont les éléments les plus importants ou les plus utiles à la compréhension de l'espace), et comment le visualiser, par exemple, avec quelle forme, quelle couleur, quel symbole ... Dans les bases de données géospatiales communautaires, les données SIG sont créées par différents acteurs à l'aide de différents outils, par image

21. <https://www.ogc.org/standard/wms>
22. <https://www.ogc.org/standard/wmts>
23. <https://geojson.org/>
24. <https://www.ogc.org/standard/wfs>

satellite ou photogrammétrie. Cela a induit la nécessité d'identifier les niveaux de détails associés à chaque feature automatiquement [TB13] afin de pouvoir les corriger et harmoniser [TB14].

Dans CityGML, tous les objets de la ville peuvent être représentés à cinq niveaux de détail différents et bien définis (LoD0 à LoD4)²⁵, avec une précision et une complexité structurelle croissantes [Kol09 ; Bil+14], le dernier niveau permettant de représenter l'intérieur des bâtiments (figure 2.12).

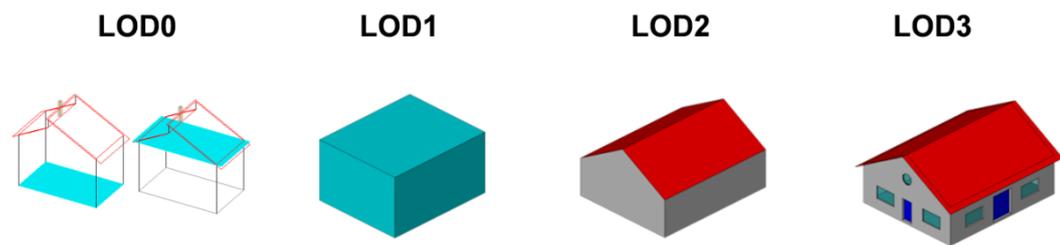


Figure 2.12 : Les différents niveaux de détails (de LoD0 à LoD4) décrits dans le modèle de données CityGML 3.0[OGC18a]

Le niveau de complexité d'une représentation est originellement décrit en fonction de critères géométriques. Cependant, les maquettes numériques issues du CIM et du BIM permettent d'associer des données sémantiques aux données géométriques, qui peuvent correspondre ou non au degré de complexité des données géométriques. [SK07] nomme ce phénomène la cohérence spatio-sémantique. On peut observer dans la figure 2.13 qu'une même maquette CityGML visuellement identique peut avoir différents degrés de cohérence spatio-sémantique : une géométrie simple sans données sémantiques, une géométrie qui contient des informations de hiérarchisation des objets qui composent le bâtiment, et dans le dernier cas, des données sémantiques liées à chacun de ces objets.

25. <https://docs.ogc.org/is/20-010/20-010.html#toc24>

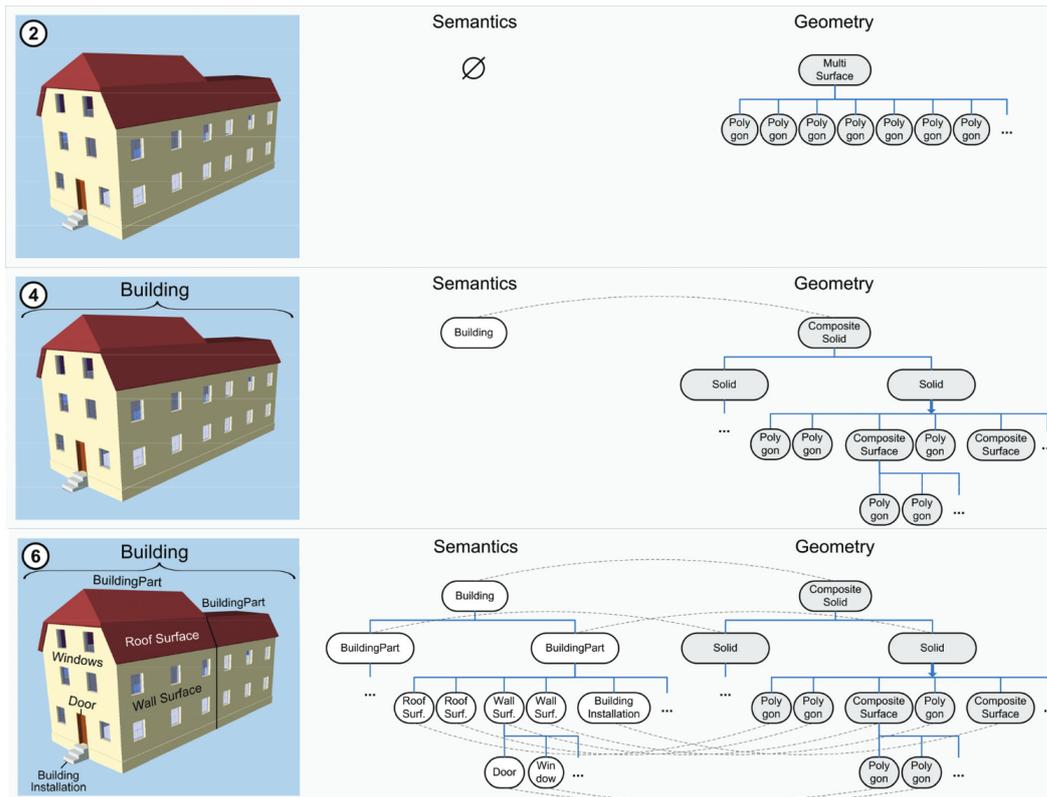


Figure 2.13 : Trois exemples de maquettes CityGML d'un bâtiment avec différentes cohérences spatio-sémantiques [SK07] : une maquette sans données sémantique, une maquette avec des données géométriques structurées et hiérarchisées et une maquette avec des données sémantiques et géométriques structurées et hiérarchisées, dites cohérentes.

Pour le modèle IFC, le sigle LOD a une signification différente de celle de CityGML. Il signifie Level Of Development, illustré dans la figure 2.14. Il est défini, pour chaque entité décrite, selon une complexité géométrique (Level of Geometry) et un ensemble d'informations nécessaires (Level Of Information). Selon le système de LoD utilisé, celui-ci peut s'appliquer à l'ensemble de la maquette d'un bâtiment, comme pour CityGML, ou bien à chacun des éléments qui le composent, comme pour IFC, ce qui pose problème lors de travaux d'alignement de concepts de LoD[AB22].

Identification		Level of Geometry	Level of Information (LOI)								
Element Type	IFC Class	LOG	Name	Core Material	Load-bearing Function	Surface Covering (texture)	Fire Protection Characteristics	Part of Escape Route?	Sound Insulation Characteristics	Is External?	Thermal Transmittance
Windows	IfcWindow	300	x	x			x	x	x	x	x
Walls	IfcWall	350	x	x	x	x	x	x	x	x	x
Curtain Walls	IfcCurtainWall	350	x	x	x	x	x	x	x	x	x
Stairs	IfcStair	200	x	x				x		x	
Ramps	IfcRamp	200	x	x				x		x	
Doors	IfcDoor	200	x	x				x		x	
Ceiling	IfcCeiling	300	x	x	x		x				
Sanitary	IfcSanitary	100	x								
Rooms	IfcSpace	-	x					x		x	
Slabs	IfcSlab	300	x	x	x		x		x	x	
Roofs	IfcRoof	300	x	x	x		x		x		x
Beams	IfcBeam	200	x	x	x					x	
Columns	IfcColumn	200	x	x	x		x	x		x	
Structral Truss	IfcAssembly	200	x	x	x						
Foundation	IfcFooting	350	x	x	x		x				x
Framing	IfcBuildingElementProxy	300	x	x	x		x		x		

Figure 2.14 : Exemple de description de niveau de développement (LOD) de certains classes BIM [AB22], constitués de niveau de géométrie (LoG) et niveau d'information (LoI).

Usuellement, les niveaux de détails d'une maquette ou d'une géométrie sont dérivés à partir du niveau le plus complexe créé, comme l'illustre la figure 2.15. Cette dérivation peut être effectuée manuellement [FR14] ou automatiquement [VLA15; GPG18; FM09].

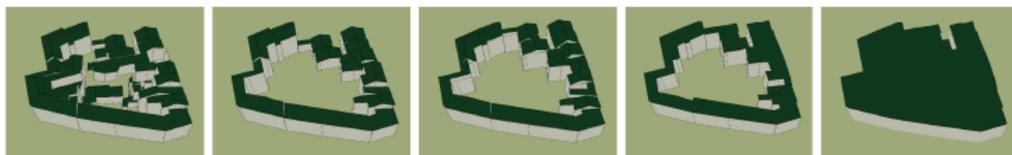


Figure 2.15 : Création de niveaux de détails d'un quartier à partir de données existantes, adapté de [GPG18]

[Bil+14; BLS16] étend le concept de LoD défini par CityGML pour ajouter des critères sémantiques au LoD existant. Une autre approche consiste à séparer les LoD entre LoD pour l'intérieur des bâtiments et l'extérieur des bâtiments [Tan+18]. [CDD13] propose un alignement entre les LoD CityGML et les LoG IFC. Grâce à ce genre d'alignement, il est alors possible de générer automatiquement des maquettes BIM depuis des maquettes CIM [Don+16].

Les travaux menés par [La +22] illustrent que l'on peut définir un système de niveaux de détails en fonction de modèles de données différents utilisés pour représenter les données géométriques (figure 2.16). Par exemple, ils définissent le LOD 0 comme étant des plans 2D, le LOD 1 et 2 des modèles 3D simples et les derniers LOD sont des modèles 3D texturés très précis. Ces représentations d'une même entité peuvent être créées à partir de différentes techniques d'acquisitions.

LoD	Description	Visualization	LoD	Description	Visualization
0	2D shape buildings visualization		IBM	3D textured mesh buildings visualization	
1	2.5D shape buildings visualization				
2	3D shape buildings visualization				
			PCBM	3D point cloud building navigation	

Figure 2.16 : Exemple d'un système de niveaux de détails (LoD) [La +22] défini selon différents modèles de données géométriques.

Le modèle CityGML et différents acteurs du BIM^{26 27 28} ont associé une échelle à chaque LoD CityGML ou IFC, ce qui permet alors de les associer à un ou plusieurs niveaux d'échelles OpenStreetMap (figure 2.17).

26. https://www.corenet.gov.sg/media/586132/Singapore-BIM-Guide_V2.pdf

27. https://changeagents.blogs.com/Linked_Documents/BIPS%203D%20Working%20Method.pdf

28. https://www.deges.de/wp-content/uploads/2020/02/BIM-LOD-Konzept_V16.pdf

Level	m/pixel	Width (m)	Area (m ²)	Examples of areas to represent	Scale	CityGML	Singapore BIM Guide v 2.0	3D Working Method 2006	BIM LOD Konzept
0	156 543	40 075 008	1,61E+15	Whole World	1:500 Million				
1	78 272	20 037 504	4,02E+14		1:250 Million				
2	39 136	10 018 752	1,00E+14	Subcontinental area	1:150 Million				
3	19 568	5 009 376	2,51E+13	Largest Country	1:70 Million				
4	9 784	2 504 688	6,27E+12		1:35 Million				
5	4 892	1 252 344	1,57E+12	Large African Country	1:15 Million				
6	2 446	626 172	3,92E+11	Large European Country	1:10 Million				
7	1 223	313 086	9,80E+10	Small Country, US State	1:4 Million				
8	611	156 543	2,45E+10	Administrative region	1:2 Million				
9	306	78 272	6,13E+09	Wide area, large metropolitan area	1:1 Million				
10	153	39 136	1,53E+09	Metropolitan area	1:500 000				
11	76	19 568	3,83E+08	City	1:250 000				
12	38	9 784	9,57E+07	Town, city district	1:150 000				
13	19	4 892	2,39E+07	Village, suburb	1:70 000				
14	9,55	2 446	5,98E+06		1:35 000				
15	4,78	1 223	1,50E+06	Small road	1:15 000	LoD1			
16	2,39	611	373 927	Street	1:8 000	LoD2			
17	1,19	306	93 482	Block, park	1:4 000	LoD2			LOD100
18	0,60	153	23 370	Site (f/cSite), trees	1:2 000	LoD3			LOD100
19	0,30	76	5 843	Local highway, crossing	1:1 000	LoD3	LOD100		LOD100
20	0,15	38	1 461	Mid-sized building	1:500	LoD4	LOD100	Information Level 1	LOD200
21	0,07	19	365	Floor	1:250		LOD200 - LOD300	Information Level 2	LOD200
22	0,04	9,55	91	Room	1:125		LOD300 - LOD350 - LOD400	Information Level 3	LOD200
23	0,0187	4,78	23		1:60		LOD 400 - LOD 500	Information Level 4 - 5	LOD300
24	0,0093	2,39	5,71	Table	1:30			Information Level 4 - 5	
25	0,0047	1,19	1,43	Door, Window	1:15				
26	0,0023	0,60	0,36	Air Conditioning	1:8				

Figure 2.17 : Alignement des LoD CityGML et IFC sur le niveau d'échelle de OpenStreetMap

2.2.3 Discussion

L'un des principaux enjeux, motivé par la multiplication de la création de données sur un même territoire, est de pouvoir concilier et fusionner plusieurs maquettes numériques 3D afin de mettre en commun les informations qu'elles contiennent pour permettre à l'utilisateur de réfléchir aux villes de demain [Bat21]. Il apparaît nécessaire de trouver des solutions pour fournir un moyen efficace de relier et visualiser les représentations d'une même entité géospatiale existant dans différentes sources de données, et de spécifier la manière dont les liens entre les ensembles de données et les représentations des entités doivent être créés et gérés [PZL17]. Il est nécessaire d'étudier plus en avant les représentations multiples existantes d'une même entité géospatiale afin de résoudre ce problème fondamental [Bre+20]. Nous proposons dans le chapitre 5 une conceptualisation des représentations hétérogènes d'une même entité, afin de pouvoir proposer une solution pour les lier à l'entité qu'elles représentent. Ces travaux permettent de fournir à un utilisateur ou un système de naviguer dans l'ensemble des représentations existantes d'une entité pour récupérer le plus grand nombre d'informations possible.

L'existence de plusieurs représentations d'une même entité implique qu'il existe différentes descriptions et simplifications d'une même entité [Bat18a ; Bat21]. Les

modèles de données proposent des systèmes hétérogènes de niveaux de détails pour définir la complexité d'une maquette numérique. Une même entité peut disposer de différentes représentations 3D, apportées par des données 3D structurées par un modèle, ou non. Nous proposons dans le chapitre 6 une étude de la conceptualisation générique d'entités par CityGML et IFC afin de proposer une méthodologie pour définir la complexité d'une représentation, indépendamment de son modèle de données d'origine, et de l'échelle de l'entité représentée. Cette contribution permet d'adapter la représentation d'une entité à visualiser selon le contexte de l'utilisateur, parmi un ensemble de représentations déjà existantes.

Création de vues unifiées et navigables de données géospatiales

Les travaux de ce chapitre ont été présentés aux Journées Françaises de l'Informatique Graphique [Col+21] et publiés dans l'annale du journal ISPRS Photogrammetry, Remote Sensing and Spatial Information Sciences [Col+22 ; Mar+22].

Dans ce chapitre, nous nous intéressons à la première problématique énoncée : comment créer une vue unifiée de données sémantiques et géométriques géospatiales hétérogènes dans un même contexte web ?

Permettre à un utilisateur de créer une vue unifiée de données implique de lui donner la capacité de consulter et visualiser un ensemble de données hétérogènes dans un même contexte web. L'affichage de données géométriques requiert un viewer spécifique, capable d'interpréter les données pour leur donner du sens, c.-à-d. construire et afficher les géométries en 2D ou en 3D. Dans le cas de données géospatiales, il est nécessaire de gérer les questions associées aux systèmes de projections utilisés puis de positionner correctement les géométries dans l'espace.

Nous détaillons dans ce chapitre les contributions suivantes, présentées dans la section 1.3 :

- C1 : Méthodologie d'intégration de données géospatiales hétérogènes pour la création de vues contextuelles.
- C5 : Développement d'un outil open-source pour intégrer des données géométriques issues de modèles de données géospatiales hétérogènes afin de les visualiser.

3.1 Vue contextuelle d'une entité géospatiale

Notre objectif est de pouvoir permettre à l'utilisateur de créer, dans un navigateur, une vue composée de multiples représentations d'une même entité. Celles-ci permettent d'apporter différentes informations sur la situation d'une entité : ce

qu'elle est, où elle est, comment elle évolue ... Le contexte d'une entité étant défini comme toute information qui peut être utilisée pour caractériser la situation d'entité [Abo+99], nous proposons de définir une vue contextuelle comme la visualisation simultanée d'une ou plusieurs représentations d'une entité et de son contexte.

Dans notre contexte d'étude, constitué de données CIM, BIM et GMAO, il est possible de distinguer deux types de représentations : thématique ou spatiale. La première, basée sur des données sémantiques, regroupe toute information permettant de caractériser une entité représentée par des données textuelles, en utilisant des formulaires, documents ou champs. La seconde, spatiale, regroupe des informations liées à la localisation, la topologie ou l'aspect physique d'une entité. Cette représentation est constituée de données géométriques 2D ou 3D. Cette représentation spatiale peut-être réaliste ou symbolique, à l'aide d'une icône ou d'un symbole représentant l'entité. Dans le cas où seule la localisation d'une entité est intéressante, il est possible de choisir une représentation abstraite. Par exemple, les églises sont représentées par une croix et les mairies par un rectangle rouge dans une carte topographique IGN¹.

Si les représentations thématiques sont facilement visualisables car constituées de données textuelles ou multimédia, il est nécessaire de trouver une solution permettant d'utiliser efficacement, dans un contexte 3D, les données géométriques hétérogènes géospatiales sous-jacentes aux représentations spatiales.

3.1.1 Méthodologie pour la création d'une vue contextuelle

Pour permettre la création d'une vue contextuelle, il est nécessaire de permettre à l'utilisateur de naviguer, interagir et visualiser différentes représentations d'une même entité. Nous proposons une méthodologie, illustrée dans la figure 3.1, pour naviguer dans des représentations thématiques et spatiales d'une même entité, issues de données géospatiales hétérogènes, et permettre à l'utilisateur de créer des vues contextuelles de cette entité.

1. <https://www.geoportail.gouv.fr/carte>

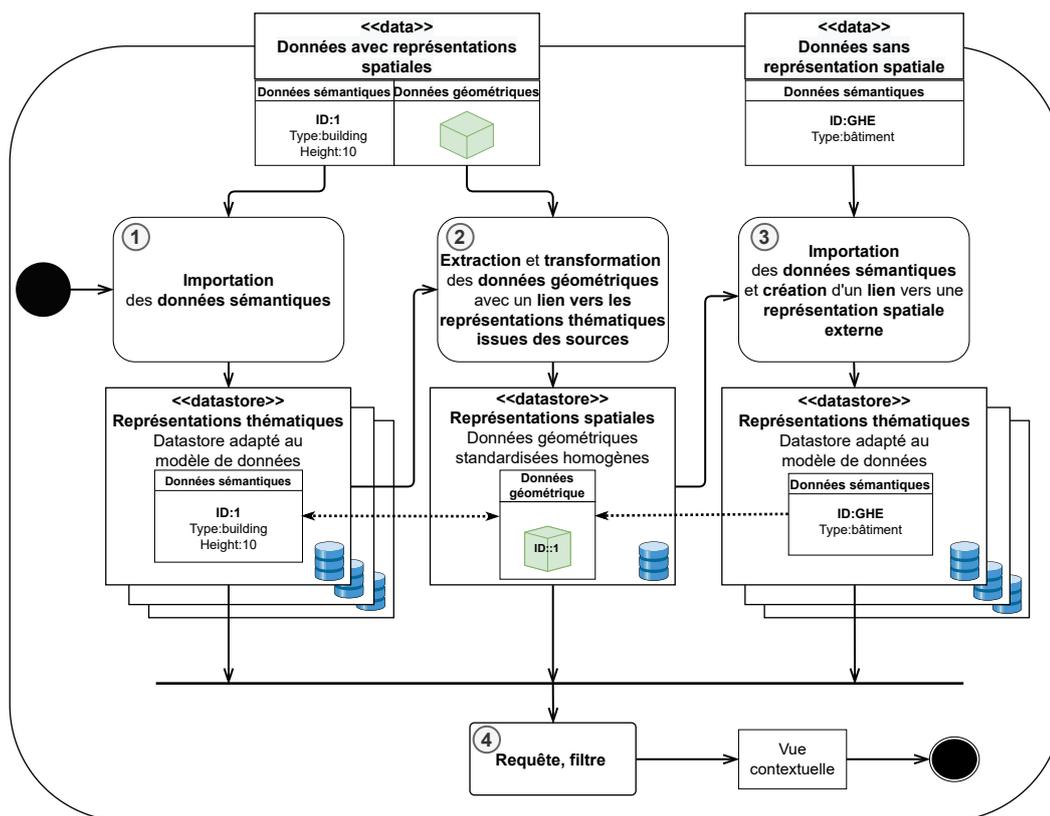


Figure 3.1 : Méthodologie de création de vues contextuelles proposée, adaptée de [Col+22]

Nous proposons de distinguer les données géospatiales qui ont une représentation spatiale de celles qui n'en ont pas (en haut de la figure 3.1) car les données géospatiales qui ne disposent pas de représentation spatiale ne sont, par défaut, pas accessible depuis un environnement géospatial 2D ou 3D. Ces données peuvent être stockées dans des bases de données ou des fichiers. Elles peuvent représenter des entités physiques ou abstraite, par exemple des évènements de la GMAO, comme une déclaration de panne, ou une demande d'intervention.

Etape 1 : Importation des représentations thématiques associées à des représentations spatiales : Il existe des solutions pour héberger et requêter des données sémantiques adaptées à chaque modèle de données. Nous proposons d'importer les données sémantiques sous-jacentes à chaque représentation spatiale dans un datastore adapté au modèle de données d'origine. Cela implique l'utilisation d'un datastore unique par modèle de données rencontré.

Etape 2 : Extraction et transformation des données géométriques de représentations spatiales vers un format pivot : Les modèles de données géospatiaux ont été pensés

pour structurer et échanger des données mais la visualisation des données géométriques nécessite d'interpréter celles-ci pour les visualiser dans un contexte 3D. Nous proposons d'extraire et de transformer les données géométriques sous-jacentes aux représentations spatiales vers un format pivot homogène et standardisé. Les données géométriques sont alors structurées et stockées uniformément indépendamment de leur modèle d'origine, ce qui permet au viewer de visualiser toute représentation spatiale. Lors de cette étape de transformation, il est nécessaire de garder un lien entre les représentations spatiales et leurs sources, car elles seront séparées d'éventuelles représentations additionnelles présentes dans ces sources. Cela assure qu'il est toujours possible d'accéder aux représentations sémantiques originellement associées aux représentations spatiales. Pour donner un exemple concret, cela permet de visualiser, dans un contexte 3D géospatial, une représentation spatiale issue de données CIM représentant un bâtiment, et d'accéder ensuite à des informations présentes dans ces données, comme l'adresse ou la hauteur du bâtiment. Il est important de préciser qu'une source de donnée géospatiale décrit usuellement plusieurs entités. Ce lien ne doit pas être simplement entre l'ensemble des données géométriques présent dans la source et la source, mais entre chaque représentation spatiale d'une entité et son pendant thématique. Cela peut être fait en conservant l'identifiant de l'entité (par exemple, son GMLID pour des données CityGML, son GUID pour des données IFC) lors de la transformation. Comme cet identifiant peut être retrouvé dans d'autres sources pour désigner une autre entité, il est aussi nécessaire de conserver une description de la source d'origine. Ces informations devant être conservées avec les représentations spatiales, le format pivot choisi pour structurer les données géométriques doit permettre d'associer des données sémantiques aux représentations spatiales des entités.

Etape 3 : Importation des représentations thématiques sans représentations spatiales : Les données qui ne comprennent pas de représentations spatiales sont importées dans un datastore adapté au modèle de données d'origine. Nous proposons de créer un lien dans ce datastore entre ces représentations thématiques et les représentations spatiales issues d'autres données. Cela permet d'accéder depuis une représentations spatiale à différentes représentations additionnelles issues d'autres sources d'une même entité, comme des documents, des données sémantiques GMAO ...

Etape 4 : Création de vues contextuelles par l'utilisateur : L'utilisateur a accès à l'ensemble des représentations thématiques associées à une représentation spatiale d'une entité grâce aux liens créés dans l'étape 2 et 3. En filtrant les représentations spatiales à visualiser et en naviguant dans les représentations additionnelles, l'utilisateur peut créer ses propres vues contextuelles.

Cette méthodologie assure dans un premier temps de pouvoir visualiser n'importe quelle représentation spatiale indépendamment de son modèle de donnée d'origine grâce à l'utilisation d'un format pivot. Dans un second temps, grâce aux liens entre différentes représentations hétérogènes d'une entité issues de multiples sources de données, elle permet à un utilisateur de créer sa propre vue contextuelle d'une entité en naviguant dans les représentations existantes et en sélectionnant celles qu'il souhaite visualiser. Il peut alors visualiser entièrement les données d'origines, en utilisant l'ensemble des représentations thématiques et spatiales existantes.

Les données sémantiques sont hébergées à l'aide de solutions adaptées à chaque modèle de données d'origine pour en faciliter la requête et le stockage. Seules les données sémantiques correspondant aux liens avec les données sources sont conservées avec les données géométriques. Ce choix permet de conserver l'ensemble des informations présentes dans les données d'origine. Le choix de conserver les données sémantiques avec les données géométriques lors de leur transformation vers un format pivot adapté n'a pas été retenu. Aucun standard n'ayant été créé pour les conserver optimalement, cela nécessiterait de définir une nouvelle structure homogène pour ces données, ce qui limite l'interopérabilité de la solution. Cela peut avoir un impact fort sur la taille des données créées et sur la fluidité de transfert et de visualisation des données géométriques. Une telle solution doit faire l'objet d'études approfondies afin d'être potentiellement retenue.

Afin d'appliquer la méthodologie présentée, il est nécessaire de sélectionner un format pivot adapté à l'utilisation des² données géométriques géospatiales. Les données géométriques doivent être structurées de façon à être interprétées par différents viewers. Pour garder le lien entre les représentations thématiques d'origines associées aux représentations spatiales transformées, le format pivot doit permettre de d'associer des données sémantiques aux données géométriques.

3.2 Homogénéisation de données géométriques géospatiales à l'aide d'un format pivot : 3DTiles

La visualisation de données géométriques géospatiales hétérogènes sur le web est une problématique qui dispose d'ores et déjà de solutions, comme l'utilisation d'un format pivot. S'il existe quelques outils open source [Che+18 ; Xu+20] pour transformer des données géométriques géospatiales vers un format pivot, il existe un réel besoin d'un outil open-source pour effectuer cette transformation à partir de données hétérogènes. Nous devons nous assurer que le format pivot choisi

traite correctement les données géospatiales (implicitement géoréférencées), afin de faciliter leur visualisation et de résoudre des problèmes tels que la manipulation du système de projection. Nous proposons un outil, Py3DTilers, pour transformer des données géométriques géospatiales en 3DTiles [OGC19], format standardisé populaire open-source utilisé par la communauté scientifique et industrielle, décrit dans le chapitre précédent 2.1.2.

Le choix du format 3DTiles comme format pivot pour uniformiser et homogénéiser la structure des données géométriques a été motivé par le fait que ce format est optimisé pour la visualisation de données géométriques géospatiales. Il est possible à l'aide de ce format d'associer de la sémantique à chaque représentation spatiale.

Py3DTilers permet de créer des 3dtiles pour visualiser différentes représentations spatiales dans un même contexte. Cet outil laisse la possibilité d'ajouter des données sémantiques, associées aux représentations spatiales. Cela permet de créer le lien entre les données extraites et d'origines pour accéder à des représentations thématiques additionnelles, comme indiqué dans l'étape 2 de la méthodologie de création de vue contextuelle 3.1. La création de cet outil a été pensée pour permettre l'étude de différents sujets tels que la fluidité de visualisation de données hétérogènes, l'impact de l'organisation spatiale des entités sur leur visualisation, la création de niveaux de détails à l'aide de représentations hétérogènes d'une même entité ou l'apport de données sémantiques associées à des données géométriques.

Cet outil est basé sur Py3dTiles², un outil open-source développé par la société Oslandia³ qui permet de lire des nuages de points et de les transformer en 3DTiles. Cette librairie permet de représenter les 3DTiles en mémoire et de les modifier. Cet outil a été originellement développé dans la plateforme UD-SV dans le cadre des travaux de [GPG18].

Un ensemble de notions spécifiques au standard 3DTiles a été ajouté. Par exemple, il est possible de retrouver la notion de tile (tuile), tileset (ensemble de tuiles) et extension. Les Feature et Batch Table, permettant de lier des données sémantiques à des représentations spatiales dans une tuile, ont été entièrement intégrées. Cette spécificité permet l'étude de l'ajout de données sémantiques à des représentations spatiales, tel que leur structuration ou l'impact de leur présence totale ou partielle sur le poids du fichier produit ou sur la fluidité de la visualisation.

Afin de garantir le bon respect de la spécification du standard, un système de validation par JSON Schema a été mis en place. Celui-ci permet d'assurer que les 3DTiles

2. <https://gitlab.com/py3dtiles/py3dtiles>

3. <https://oslandia.com>

créées seront utilisables sur différents viewers. Cette validation par schéma permet de créer des extensions à la spécification originelle. Deux extensions sont disponibles. La première sert à créer une hiérarchie spatiale entre les entités représentées, nommée Batch Table Hierarchy, créée par Cesium⁴. Avec cette extension, il est possible de distinguer les différents éléments constituant une entité, par exemple de représenter un bâtiment, les étages de ce bâtiment, et les murs de chaque étage. La deuxième a été créée dans le cadre de travaux [Jai20b] pour représenter et visualiser l'évolution de la ville dans le temps. Cette extension décrit comment structurer les données géométriques pour leur associer des temporalités et comment elles ont évolué entre deux versions.

Cet outil a été pensé de façon à pouvoir travailler sur les représentations géométriques des entités, indépendamment de leurs modèles de données d'origines. Une fois les géométries extraites des sources de données, les différents traitements seront appliqués de façons identiques pour chaque représentation. La mise à disposition de cet outil et la réflexion autour de son architecture permettra l'étude d'autres sujets autour de la visualisation de données 3D géospatiales hétérogènes par la communauté scientifique.

Py3DTilers est découpé en un ensemble de Tiler, dont chacun est responsable de lire, extraire, et trianguler les données géométriques d'un modèle de données spécifique. Il existe des Tilers pour créer des 3DTiles à partir des standards suivants :

- IFC : en utilisant une librairie nommée IFCOpenShell⁵, responsable de la lecture du fichier et de la récupération de la géométrie de chaque entité. Chaque entité est géolocalisée, étant à l'origine positionnée relativement à son parent (une salle est placée par rapport à un étage, qui lui-même est placé relativement au bâtiment).
- CityGML : en utilisant une base de données 3DCityDB⁶, qui implémente l'ensemble du standard CityGML.
- GeoJson : utilisé de façon classique dans les SIG pour représenter des géométries 2D, il est possible d'extruder ces géométries à l'aide de données sémantiques associées à chaque entité dans le fichier ou d'une valeur arbitraire choisie par l'utilisateur.
- Obj : format classique et très populaire au sein de la communauté d'informatique graphique, il est possible de lire et d'exporter les représentations spatiales en .Obj. Cela permet de visualiser plus facilement le résultat obtenu après une

4. https://github.com/CesiumGS/3d-tiles/blob/main/extensions/3DTILES_batch_table_hierarchy

5. <http://ifcopenshell.org/>

6. <https://github.com/3dcitydb>

transformation, d'autant plus qu'il existe encore peu de viewers open-source capables de visualiser rapidement et simplement des 3DTiles.

- 3DTiles : il est possible de lire un fichier 3DTiles afin de l'analyser, par exemple pour connaître le nombre de tuiles, le nombre d'entité par tuile, le nombre de triangles, le nombre de vertex ... Il est aussi possible de reprojeter ou de déplacer le tileset dans l'espace, comme pour les autres formats.

Le processus de création d'un 3DTiles suit la méthodologie suivante, illustrée dans la figure 3.2 :

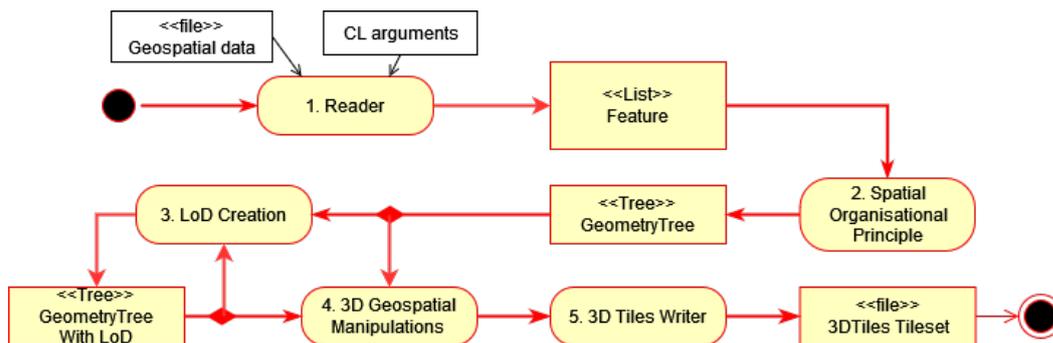


Figure 3.2 : Diagramme d'activités du process de création de 3DTiles à partir de données géospatiales from Marnat et al.[Mar+22]

Etape 1 : Lecture des géométries et création de features : Après avoir extrait et triangulé les données géométriques, chaque entité géospatiale distincte est stockée sous forme de *feature*, caractérisée par un identifiant, une représentation géométrique et d'éventuelles données sémantiques additionnelles. Cette abstraction en *feature* permet l'application d'un process unique de transformation en 3DTiles, indépendant du format d'entrée.

Etape 2 : Partitionnement des features dans l'espace : Une fois l'ensemble des *features* créé, elles sont distribuées dans un arbre où chaque noeud représente un sous-ensemble des features disponibles. Cette distribution dépend d'une méthode de partitionnement spatiale ou sémantique des entités. Il est possible de les organiser spatialement à l'aide d'un kd-tree⁷ classique ou bien à l'aide d'entités délimitantes comme des routes. Il est possible de les organiser sémantiquement, par type d'entité (mur, fenêtre, dalle, ...), ou en conservant la hiérarchie induite par le modèle de données d'origine.

Etape 3 : Création de LoD (niveaux de détails) : Il est possible de créer des niveaux de détails, ajoutant de la profondeur à l'arbre. Une *feature* peut se voir ajouter de

7. https://en.wikipedia.org/wiki/K-d_tree

nouvelles représentations plus ou moins détaillées. Un ensemble de features peut être regroupé en une seule feature parente qui les regroupe et les simplifie. Plus les noeuds sont profonds dans l'arbre et plus la représentation géométrique d'une feature est précise et détaillée.

Etape 4 : Transformation géospatiale : Les features peuvent être transformées par l'utilisateur dans l'espace : il peut changer le système de projection utilisé, appliquer une translation ou une rotation, ou encore changer l'échelle utilisée.

Etape 5 : Ecriture du 3D Tiles : L'arbre contenant les géométries est transformé en 3D Tiles, chaque noeud devenant une tuile du tileset final. Des données sémantiques, comme les identifiants des entités originaux, peuvent être préservées grâce à l'utilisation des batch et feature tables. Les géométries des tuiles sont écrites en GITF, puis encodées en binaire.

3.3 Démonstration et résultats

Dans cette section, nous présentons la démonstration mise en place pour valider la méthodologie de création de vue contextuelle, et les résultats obtenus après la transformation des données géométriques à l'aide de Py3DTilers et leur utilisation pour créer des vues contextuelles.

3.3.1 Architecture technique

L'architecture technique de la démonstration est présentée dans la figure 3.3. Les différents composants sont répartis suivant en plusieurs couches : les fichiers de données sources utilisés, les différents traitements appliqués sur ces données à l'aide de différents outils pour les importer dans des bases de données, puis les différents services web utilisés par le viewer pour récupérer ces données.

Pour cette démonstration, nous avons utilisé des multimédias et des données de la GMAO qui apportent des représentations thématiques de différentes entités. Les données CIM et BIM apportent des représentations spatiales 3D et thématiques de ces entités, et les données SIG et nuage de points apportent seulement des représentations spatiales.

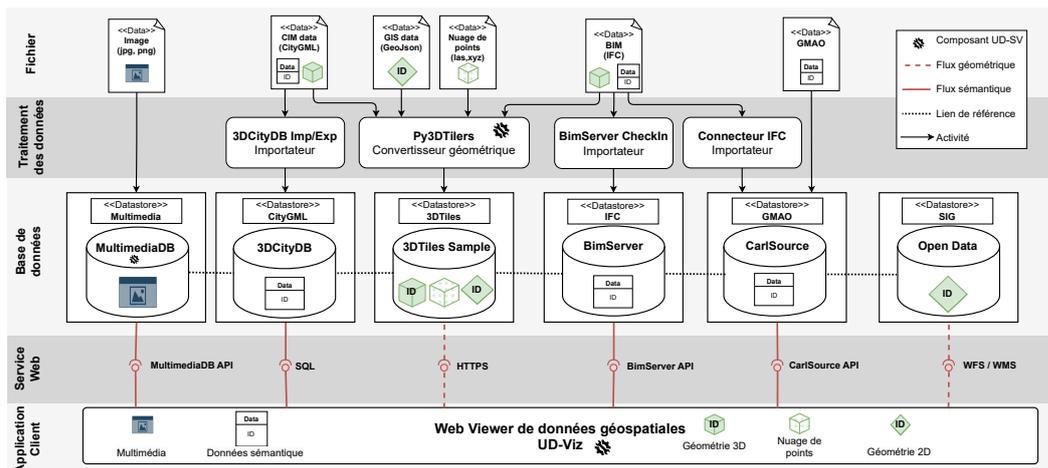


Figure 3.3 : Architecture technique, adapté de [Col+22]

Les représentations thématiques sont stockées dans différentes bases de données adaptées à leur modèle de données respectif : multimediaDB pour les multimédias, 3DCityDB pour les données CityGML, BimServer pour les données IFC, et 3DTiles sample, un serveur http, pour stocker les fichiers 3DTiles obtenus après transformation. Afin d’assurer une reproductibilité et réutilisabilité de cette transformation, l’ensemble des outils utilisés (en dehors de CarlSource, outil propriétaire de l’entreprise) sont des outils open-source.

Les liens entre les représentations thématiques additionnelles d’une entité sont permis grâce à l’extension des bases MultimediaDB et CarlSource, permettant d’associer une représentation de chaque base à une représentation IFC ou CityGML à l’aide de leurs identifiants.

Nous utilisons des données open-source provenant du Grand Lyon⁸ pour récupérer des fonds de plans cartographiques et de potentielles données SIG additionnelles à l’aide de flux de données standardisés comme WFS [OGC14] et WMS [OGC06]. Les 3DTiles peuvent être récupérées à l’aide de requête HTTP. Les données CityGML peuvent être récupérées à l’aide de requête SQL, et les autres serveurs de données disposent de leur propre API pour interroger les données.

Le viewer est basé sur la librairie UD-Viz. Il permet de visualiser et naviguer dans l’ensemble des représentations dans un contexte 3D géospatial. Nous avons développé une extension d’UD-Viz pour effectuer des requêtes vers BimServer en utilisant les identifiants de représentation IFC pour récupérer des informations sémantiques et les

8. <https://data.grandlyon.com/>

afficher dans la vue. Une autre extension permet de sélectionner une représentation spatiale et de visualiser le contenu sémantique associé à cette représentation.

3.3.2 Visualisation des représentations spatiales

Le choix du format pivot 3DTiles pour structurer les données géométriques permet une réutilisation aisée des représentations spatiales dans différents viewers. L'utilisation de JSON Schema renforce et valide la conformité des 3DTiles produites à l'aide de Py3DTilers en vérifiant que les champs nécessaires sont présents et correctement écrits. La vérification du contenu des tuiles en GLTF peut être effectuée à l'aide d'un outil mis à disposition par Cesium⁹. Cette conformité permet aux 3DTiles créées avec Py3DTilers d'assurer leur rôle de format pivot et d'être visualisées sur différents viewers, comme illustré dans la figure 3.4.

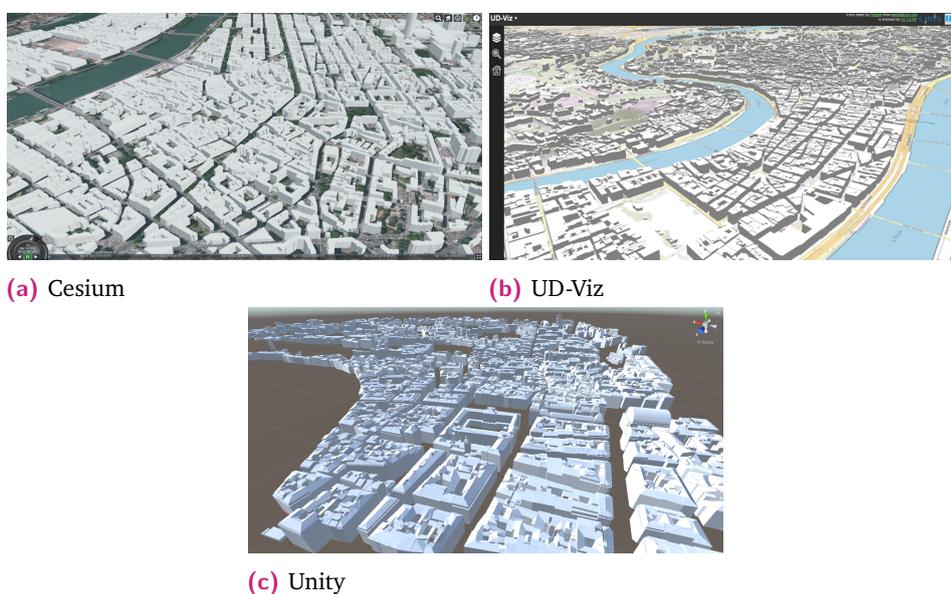
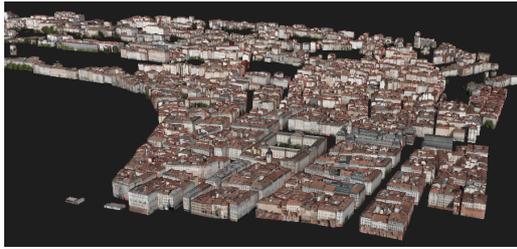


Figure 3.4 : Visualisation de 3D Tiles avec différents viewers [Mar+22]

Il est important de garantir que les représentations spatiales transformées en 3DTiles sont identiques à celles d'origine. Grâce à l'utilisation de bibliothèques open-source spécifique à chaque modèle, on peut observer dans la figure 3.5 que les représentations spatiales sont identiques malgré un changement de structure des données géométriques sous-jacentes.

9. <https://github.com/CesiumGS/3d-tiles-validator>

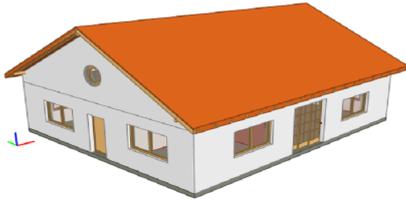


(a) Maquette CityGML dans FME^a

a. <https://fme.safe.com/>

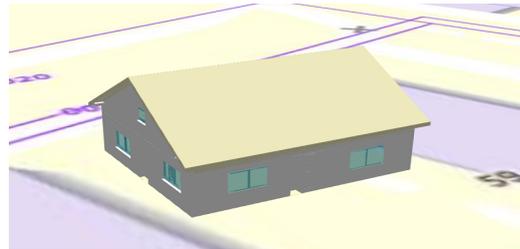


(b) 3D Tiles dans UD-Viz



(c) Maquette IFC dans Bim Vision^a

a. <https://bimvision.eu/>



(d) 3D Tiles dans UD-Viz

Figure 3.5 : Comparaison de géométries CityGML et IFC transformées en 3D Tiles [Mar+22]

3.3.3 Création de vue contextuelle d'une entité

Grâce à cette transformation en 3DTiles, nous assurons que l'ensemble des représentations spatiales d'une même entité peut être visualisé dans un même contexte géospatial 3D. L'utilisateur a accès à d'autres représentations thématiques de cette même entité grâce à l'ensemble des liens mis en place. Il est possible d'accéder dans un premier temps aux représentations thématiques originellement associées à celles spatiales visualisées à l'aide des références vers les sources stockées dans les 3DTiles, comme illustré dans le cadre A de la figure 3.6.



Figure 3.6 : Vue contextuelle d'un toit constituée de représentations thématiques issues d'un fichier IFC (dans le cadre A), de la géométrie (dans le cadre B), d'un document (dans le cadre C), et d'une représentation spatiale issue de la géométrie d'un fichier IFC (dans le cadre D) [Col+22]

Afin de pouvoir créer sa propre vue, l'utilisateur peut naviguer librement dans l'espace 3D et choisir les entités qu'il souhaite afficher. Cela lui permet de créer différentes vues d'une même entité. En croisant des représentations spatiales issues du BIM et du CIM, il est possible d'obtenir différentes vues d'une même entité à différentes échelles, comme illustré dans la figure 3.7. On peut observer dans cette figure différentes représentations spatiales d'un même bâtiment qui ont plus ou moins de détails géométriques et qui détaillent plus ou moins la composition de ce bâtiment.



Figure 3.7 : Création de vues contextuelles, depuis une vue de la ville d'un bâtiment à l'aide d'une représentation spatiale CityGML (en haut à gauche) jusqu'à une vue d'un objet dans ce bâtiment à l'aide d'une représentation géométrique IFC (en bas à droite), adapté de [Col+22]

Grâce aux liens mis en place entre des représentations thématiques à des représentations spatiales, il est possible d'accéder à d'autres représentations de la même entité, issues d'autres sources. On peut observer dans le cadre C de la figure 3.6 qu'il est possible de consulter des documents liés à une représentation spatiale d'une même entité dans une même vue.

Un autre exemple de cette navigation est illustré dans la figure 3.8, dans lequel il est possible de naviguer entre une représentation spatiale issue du BIM dans un contexte 3D d'une dalle du Pont de Anne-de-Bretagne à Nantes et une représentation thématique issue de la GMAO qui dispose d'autres informations sur cette dalle. Ce lien permet d'effectuer des opérations de maintenance classique, comme la déclaration d'un désordre, directement depuis la représentation spatiale.

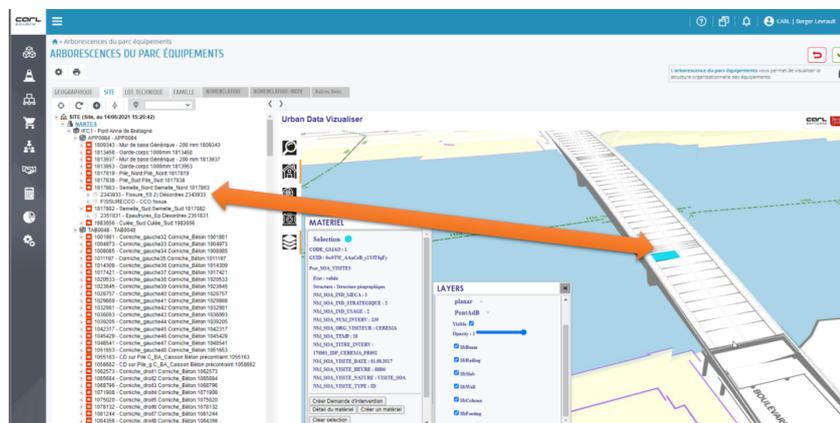


Figure 3.8 : Exemple de navigation possible entre une représentation thématique issue de la GMAO (à gauche) et une représentation spatiale issue d'un IFC (géométrie 3D en bleu à droite) [Col+22]

3.4 Discussion

Dans ce chapitre, nous avons proposé deux contributions pour permettre la création de vues unifiées de multiples représentations hétérogènes. Premièrement, nous avons présenté une méthodologie pour permettre à un utilisateur de naviguer dans un ensemble de représentations d'une même entité et de composer sa propre vue contextuelle en utilisant des représentations thématiques et spatiales issues de sources hétérogènes. Cette méthodologie s'appuie sur deux idées principales : l'extraction et la transformation des données géométriques sous-jacentes aux représentations spatiales vers un format pivot pour faciliter leur utilisation et la création de liens entre

ces représentations et des représentations thématiques pour permettre à l'utilisateur de naviguer entre ces représentations.

Notre deuxième contribution concerne la proposition d'utilisation d'un format pivot avec le standard 3DTiles et le développement de l'outil open source, Py3DTilers, afin d'assurer la transformation des données géométriques hétérogènes dans ce standard. L'utilisation de ce format pivot permet d'homogénéiser les données et de visualiser les représentations spatiales indépendamment de leur modèle de données. Ce format permet de conserver la sémantique associée aux données géométriques. Cette spécificité nous permet de créer le lien entre des représentations spatiales transformées et leurs sources. L'outil développé a été pensé pour traiter les données de manière uniforme et assurer le bon respect du standard 3DTiles. Pour évaluer ces contributions, nous avons présenté une démonstration, construite à base d'outils open-source, permettant à un utilisateur de créer sa propre vue contextuelle depuis des données BIM, CIM et de GMAO.

Ces contributions ouvrent la voie à de nouvelles recherches et de nouveaux développements. Py3DTilers a été spécifiquement pensé pour permettre l'étude d'autres problématiques associées aux données géométriques géospatiales, comme la répartition des géométries 3D dans l'espace ou la création de niveaux de détails. Py3DTilers peut permettre de suivre l'évolution du standard 3DTiles, qui, dans sa nouvelle version 1.1, s'intéresse à la compression des données.

Les données sémantiques nécessitent des solutions propres à chaque modèle de données les structurant pour les stocker et les requêter. Cela signifie que l'architecture technique d'une application utilisant un plus grand nombre de modèles de données différents peut devenir complexe car il est nécessaire d'avoir autant de datastore que de modèles de données utilisés. L'éditeur de l'application et les utilisateurs doivent avoir connaissance de l'ensemble de ces solutions et de leur système de requêtage associé, ce qui rend l'exploitation de ces données complexes. Enfin, il est nécessaire de modifier les solutions d'hébergements des sources de données qui ne disposent pas de représentation spatiale pour créer les liens entre les représentations thématiques qu'elles apportent et les représentations spatiales d'une même entité issues d'autres sources, ce qui n'est pas toujours possible.

Dans le prochain chapitre, nous nous intéresserons aux données sémantiques issues de données géospatiales afin de faciliter leur requêtage et stockage. Ainsi, nous irons plus loin dans la création d'une vue unifiée de l'ensemble des représentations disponibles dans des sources de données géospatiales hétérogènes grâce à une contribution permettant d'intégrer ces données sémantiques géospatiales.

Intégration des données sémantiques et géométriques de données géospatiales

Dans le chapitre précédent, nous nous sommes intéressés aux données géométriques pour garantir qu'il est possible de visualiser l'ensemble des représentations spatiales. Dans ce chapitre, nous explorons comment proposer une création de vue unifiée de l'ensemble des représentations spatiales et thématiques disponibles dans des sources de données géospatiales hétérogènes. Les données sémantiques nécessitent de multiples solutions adaptées à chaque modèle, ce qui rend la création d'une application et la navigation par un utilisateur dans des données issues de multiples sources complexes. Pour aider l'utilisateur à améliorer sa compréhension des entités, nous cherchons à aller plus loin que la visualisation des données 3D grâce à l'apport d'informations provenant de données sémantiques. Nous avons besoin d'une solution permettant d'intégrer l'ensemble des données géospatiales, c.-à-d., de créer une vue unifiée des données géométriques et les données sémantiques, en garantissant que l'entière des informations soit conservée. Intégrer les données sémantiques permet de requêter, structurer et héberger l'ensemble des données sémantiques de façon homogène.

Ce chapitre présente notre troisième contribution [1.3](#) : une méthodologie d'*intégration basée modèles* de données sémantiques géospatiales issues de modèles de données géospatiales hétérogènes pour étendre et enrichir la méthodologie précédente. Les travaux de ce chapitre ont été présentés lors de la 27^e conférence internationale AGILE [[Vin+24](#)].

4.1 Méthodologie d'intégration basée modèles de données géospatiales

Nous présentons une méthodologie d'intégration de données géospatiales (figure [4.1](#)) qui permet de traiter les données géométriques et sémantiques, applicable

génériquement sur toutes données structurées par un modèle. Nous proposons de conserver l'utilisation d'un format pivot homogène et standardisé pour structurer uniformément les données géométriques pour permettre au viewer de visualiser toute représentation spatiale. L'utilisation d'un format pivot pour structurer et héberger les données sémantiques peut permettre d'améliorer leur réutilisation et leur interopérabilité. La transformation des données sémantiques vers un format homogène et standardisé garantit qu'elles sont décrites à l'aide d'un langage commun et homogène ce qui facilite leur partage et leur interrogation. Les modèles de données donnent aux données une définition et une structure normalisées et partageables qui constituent une base essentielle pour l'interopérabilité. Nous proposons d'utiliser une méthodologie d'intégration de données basée modèles pour limiter d'une part la perte de données lors de leur transformation et de réduire d'autre part l'hétérogénéité sémantique entre des données provenant de différents domaines. L'utilisation de graphes sémantiques pour structurer les données semble pertinente pour intégrer des données sémantiques hétérogènes. Ils permettent une forte interopérabilité entre les données issues de différentes sources. Les relations entre les entités peuvent être clairement exprimées. La flexibilité des graphes permet de créer des données additionnelles, comme les liens entre différentes représentations, sans restructurer la solution employée. Les graphes permettent également de représenter des modèles de données, ce qui rend une intégration de données basée modèles possible.

Similairement à la méthodologie présentée dans le chapitre précédent, nous proposons de garder un lien entre les représentations issues d'une même source de données, et de créer des liens entre des représentations thématiques et des représentations spatiales externes.

Lors de la transformation des données sémantiques et géométriques vers un format pivot homogène, nous proposons de retirer les données redondantes inadaptées pour le format ciblé, c.-à-d. ne conserver que les données sémantiques dans les graphes et ne conserver que les données géométriques dans le format pivot adapté. Cela à plusieurs intérêts, dont la réduction importante de la taille des graphes générés et la garantie de conserver des données géométriques structurées de façon standard et optimale pour les visualiser dans un temps fini et acceptable par l'utilisateur.

Etape 1 : Intégration des modèles de données : Comme les données sémantiques vont être transformées à l'aide d'une méthode d'intégration basée modèles (pour garantir la validité et l'exhaustivité des données) dans un format de graphe de connaissances, la première étape de la méthodologie nécessite d'intégrer les modèles de données. Ces modèles sont ainsi transformés en ontologie, décrits avec un langage de graphe formel unique et commun. Ces ontologies sont ensuite importées dans une

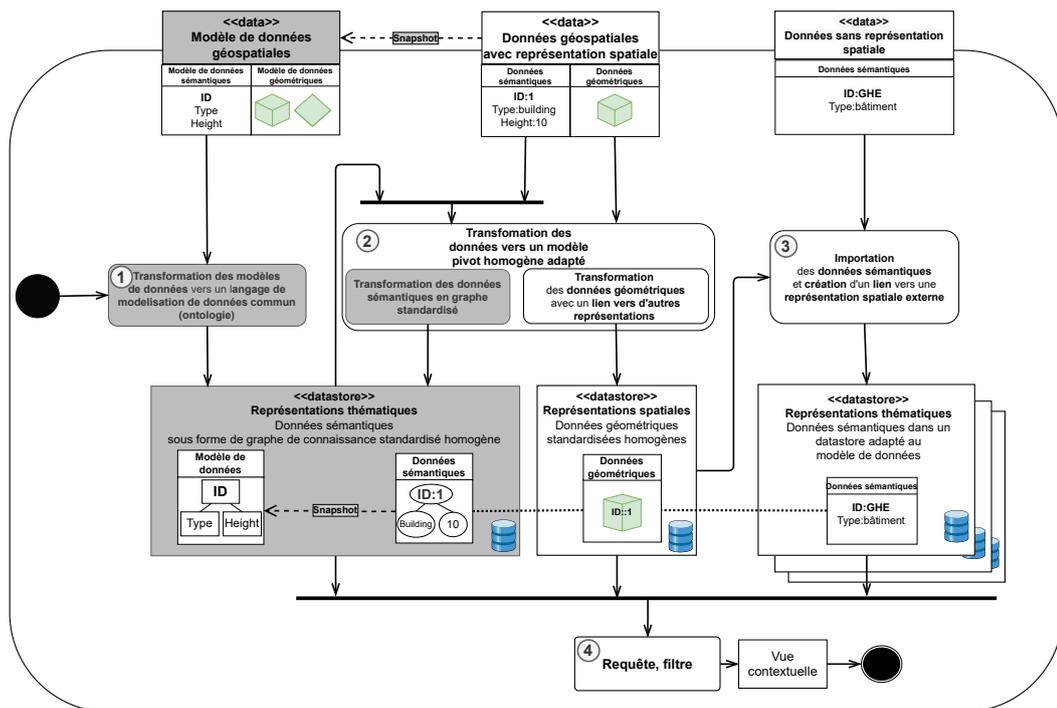


Figure 4.1 : Méthodologie d'intégration de données géospatiales proposée. Les éléments en gris ont été ajoutés par rapport à la précédente contribution.

base de connaissances. L'objectif est d'accroître l'interopérabilité entre les modèles de données et de permettre une réutilisation des données dans divers domaines d'information et applications. Certains modèles de données disposent d'ontologies existantes acceptées par la communauté ou officielles, d'autres doivent faire l'objet d'études pour créer ces ontologies afin de garantir que l'ensemble des concepts du modèle de données soit conservé correctement dans l'ontologie créée.

Etape 2 : Intégration des données sémantiques et géométriques vers des formats pivots : Les données sont extraites et transformées dans des formats spécifiques aux données spatiales et sémantiques. Les données sémantiques sont transformées en un graphe standardisé homogène. Les modèles de données intégrés à l'étape 1 sont utilisés pour effectuer la transformation de ces données sémantiques pour garantir la conservation de l'ensemble des informations. Les données géométriques sont transformées vers un format de données géométriques efficaces pour le partage, la visualisation et l'interaction avec celles-ci. Au cours de ce processus, un lien entre les données sémantiques et les données géométriques issues d'une même source doit être créé afin de garantir la possibilité de naviguer entre elles.

Etape 3 : Importation de représentations thématiques sans représentations spatiales : Certaines sources de données peuvent disposer seulement de représentations théma-

tiques, décrites à l'aide de données sémantiques. Ces données peuvent être importées dans un datastore adapté au modèle de données d'origine ou bien importées dans la base de connaissances, en utilisant leur modèle de donnée sous-jacent. Nous proposons de créer un lien dans la base de connaissances ou dans les datastores spécifiques avec des représentations spatiales des entités décrites par ces représentations thématiques.

Etape 4 : Création de vue contextuelle par l'utilisateur : Notre méthodologie offre un accès aux données homogénéisées à l'aide d'un langage de requête unique depuis une même base de connaissances qui regroupe l'ensemble des représentations thématiques issues de données géospatiales. Elle permet aux utilisateurs d'explorer, de filtrer et de visualiser des représentations thématiques et sémantiques d'entités pour créer des vues contextuelles répondant à leur besoin.

Une fois ces étapes réalisées, deux datastores ont été créés et sont disponibles : une base de connaissances contenant les modèles de données et les instances de données sémantiques, et une base de données géométriques. Des opérations sur les données peuvent être effectuées une fois cette méthodologie appliquée, comme l'illustre la figure 4.2. Des méthodes d'annotation sémantique (entity linking) sémantique, géométrique ou hybride peuvent être utilisées pour identifier les représentations existantes d'une même entité ou assurer qu'elles soient consistantes, c.-à-d. qu'elles représentent la même entité (cadre 1). Les modèles de données transformés peuvent être alignés en utilisant des approches discutées dans la section 2 pour minimiser l'hétérogénéité sémantique entre les domaines d'information BIM et CIM (cadre 2). Les résultats de ces opérations peuvent être conservés dans la base de connaissances, sous forme de relation entre les données. Lorsque ces relations entre les données sont établies, la nature formelle des ontologies permet d'utiliser des outils tels que des raisonneurs pour vérifier que les concepts et relations définis par ces modèles de données traduits sont logiquement cohérents. Ces raisonneurs permettent de déduire logiquement de nouvelles relations entre les concepts dans les modèles de données intégrés et entre les représentations des entités. Il est possible de s'intéresser à la correction et à la validation des données à l'aide de leur modèle ou à l'aide d'autres données représentant la même entité (cadre 3). L'homogénéisation des données géométriques permet différentes études, comme l'organisation spatiale des représentations spatiales pour améliorer la fluidité de la visualisation, la compression des données 3D, etc. (cadre 4).

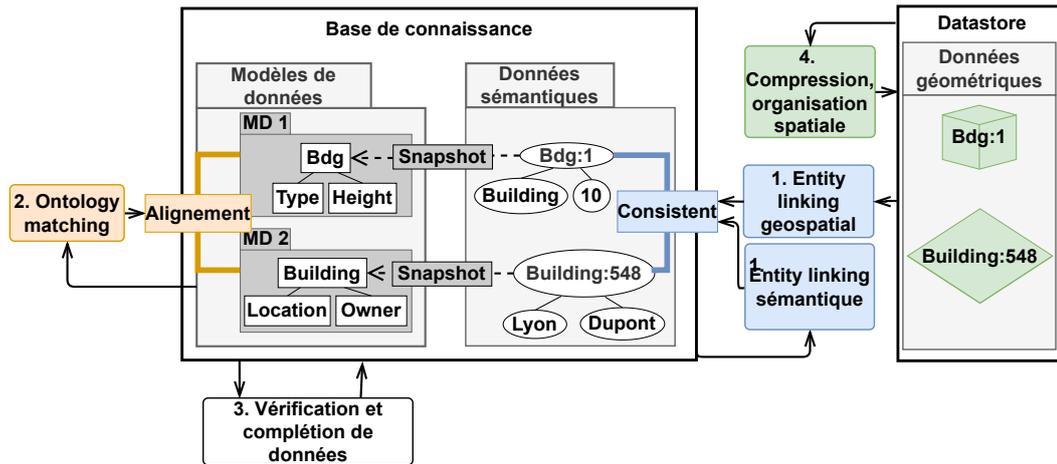


Figure 4.2 : Architecture logique obtenue après application de la méthodologie sur les représentations identifiées par 'Bdg :1' et 'Building :548', qui représentent la même entité du monde réel.

4.2 Démonstration et résultats

Dans cette section, nous présentons la démonstration mise en oeuvre pour valider la méthodologie d'intégration de données sémantiques et géométriques géospatiales. Nous commençons par présenter l'architecture technique utilisée, puis les résultats obtenus après transformation des données géométriques et sémantiques, et enfin leur utilisation pour créer des vues contextuelles.

4.2.1 Architecture technique

L'architecture mise en oeuvre pour cette démonstration est illustrée dans la figure 4.3. Nous illustrons l'intégration des données sémantiques à l'aide de données IFC et CityGML.

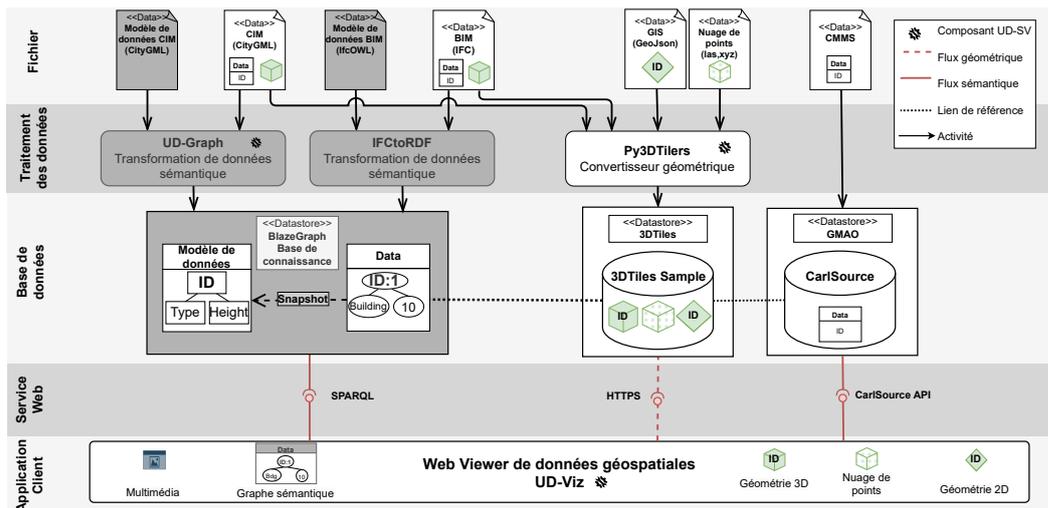


Figure 4.3 : Architecture technique. La base de connaissances permettant d’héberger des données sémantiques (modèles de données et données sémantiques) en graphes (en gris) a été ajoutée par rapport à la précédente contribution.

Etape 1 : Intégration des modèles de données : L’objectif de cette première étape est de rajouter dans la base de connaissances les ontologies des modèles de données utilisées : IFC et CityGML. La base de connaissances utilisée ici est Blazegraph¹, une base de données graphe open source, supportant des données RDF et des requêtes SPARQL. L’ontologie du modèle IFC est disponible officiellement en RDF². L’ontologie de CityGML a été créée en s’inspirant des travaux de [Vin22 ; Vin23], à l’aide de l’outil ShapeChange³ pour transformer les modèles de données CityGML 2.0 et 3.0 en RDF.

Etape 2 : Intégration des données sémantiques et géométriques : L’intégration des données géométriques a été effectuée à l’aide de Py3DTilers, présenté dans le chapitre précédent ???. Les données géométriques sont transformées en 3DTiles, en gardant l’identifiant de chaque représentation pour garantir le lien entre la géométrie et les données sources. Celles-ci sont stockées dans un simple serveur HTTP. Les données sémantiques sont intégrées à l’aide de leurs ontologies. Nous avons utilisé l’outil UD-Graph⁴ pour transformer les données CityGML en RDF. Cet outil est développé dans le cadre de la plateforme UD-SV et est open-source. L’outil IfctoRdf⁵ a été utilisé pour transformer les données IFC en RDF.

1. <https://blazegraph.com/>
2. <https://technical.buildingsmart.org/standards/ifc/ifc-formats/ifcowl/>
3. <https://shapechange.net/>
4. <https://github.com/VCityTeam/UD-Graph>
5. <https://github.com/pipauwel/IFctoRDF>

Etape 3 : Représentations thématiques sans représentations spatiales : Dans cette démonstration, nous utilisons des représentations thématiques issues de données de la GMAO, hébergées dans une base de données externe. Celle-ci conserve des liens entre les représentations thématiques qu'elles apportent et des représentations spatiales issues de l'étape 2.

Etape 4 : Création de vue contextuelle : Une application web a été développée en utilisant la librairie UD-Viz, développé au sein de la plateforme UD-SV. Les données RDF peuvent être récupérées à l'aide de requêtes SPARQL. L'affichage sous forme de graphe est possible grâce à une extension d'UD-Viz, développée dans le cadre des travaux de [Vin22]. Les données 3DTiles peuvent être récupérées à l'aide de requêtes HTTP.

4.2.2 Résultats

L'un des principaux inconvénients de l'intégration de données géospatiales sous forme de graphes est la taille des graphes obtenus. Dans le cas de données IFC, buildingSmart estime une augmentation de la taille d'un fichier selon le format de graphe choisi entre 800% (RDF) et 1350 (Turtle)%⁶. Afin de pallier ce problème, seules les données sémantiques sont gardées sous forme de graphe, les données 3D ayant été transformées en 3DTiles. Étant par nature des données complexes, répétitives et lourdes dans des formats géospatiaux, on peut observer dans le tableau 4.1 qu'en ne conservant que les données sémantiques, on obtient des graphes pouvant gagner jusqu'à 70 % de leur volume, sans perte d'information. On peut observer que le poids des géométries transformées dans un format adapté est réduit par rapport à leur poids en graphe.

6. <https://technical.buildingsmart.org/standards/ifc/ifc-formats/>

File	Format	Original	TTL	TTL sans données géométrique	3DTiles
FZK-LOD4	CityGML	15.9 mb	7.4 mb	39 kb	39.3 kb
Doua district	CityGML	23.2 mb	20.7 mb	8.1 mb	5.5 mb
Limonest	CityGML	56.6 mb	59.2 mb	15 mb	8.8 mb
FZK	IFC	2.4 mb	20.4 mb	15.9 mb	2.07 mb
AC20 Institute	IFC	10.4 mb	27.9 mb	24.4 mb	5.30 mb
Doua building	IFC	17.5mb	133.7 mb	71 mb	24 mb
Schependomlaan	IFC	47 mb	216 mb	78 mb	26.2 mb

Table 4.1 : Taille des fichiers avant et après transformation des données sémantiques en graphes (RDF) et des données géométriques en un format homogène 3D (3DTiles).

La démonstration permet de naviguer dans un ensemble de représentations spatiales et thématiques d'une même entité, créé à partir de données géométriques et sémantiques issues de sources hétérogènes dans un environnement géospatial 3D. La visualisation est adaptée en fonction du type de donnée : les données géométriques peuvent être visualisées dans l'environnement géospatial, comme le montre la figure 4.4, et les données sémantiques peuvent être explorées au moyen d'une visualisation graphique (figure 4.5). La figure 4.4 montre deux représentations du même bâtiment. Elles ont été transformées à partir d'un fichier CityGML et d'un fichier IFC, respectivement.



Figure 4.4 : Visualisation de deux représentations spatiales d’une même entité (ici un bâtiment) issues de données IFC et CityGML, transformées en 3DTiles

Les données sémantiques transformées en graphe peuvent être parcourues (figure 4.5). Le noeud parent correspondant au bâtiment de la figure 4.4 est coloré en bleu. Chaque arête représente une relation spécifique avec cette entité. Des informations supplémentaires concernant chaque noeud et les relations qu’il peut avoir avec d’autres noeuds sont décrites par leur modèle de données intégré (par exemple, une instance de `bldg:Building` est liée à la classe `Building` du modèle CityGML). Grâce à ces interfaces de graphe, les données sémantiques peuvent être explorées librement par l’utilisateur.

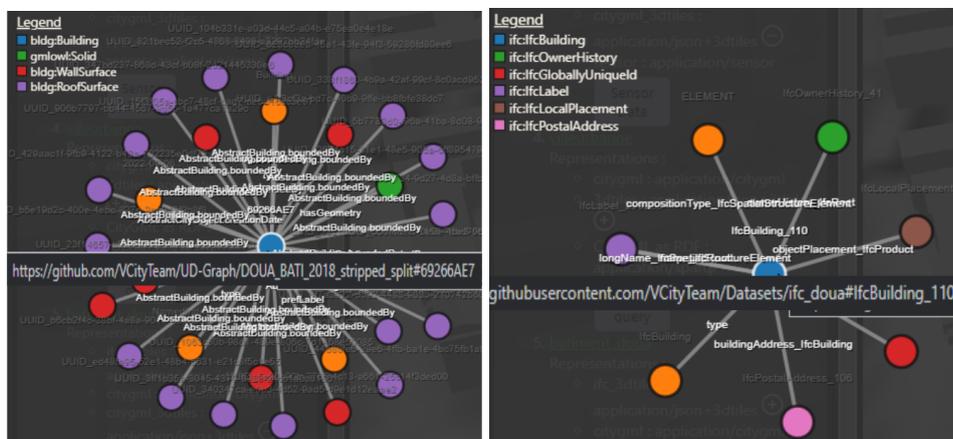


Figure 4.5 : Visualisation de deux représentations thématiques d’un même bâtiment issues de données IFC et CityGML, transformées en RDF

Un lien entre les représentations issues d’une même source est créé pendant leur intégration. Comme illustré dans la 4.6, on peut observer qu’il est possible d’interagir entre ces représentations. La représentation spatiale correspondant au noeud survolé

par la souris est mise en évidence en la colorant en bleu. En cliquant sur ce noeud, la caméra s'oriente et se positionne automatiquement pour visualiser l'ensemble de la représentation spatiale. Dans notre experimentation, ce lien est permis grâce à la conservation des identifiants des objets CityGML et IFC durant la transformation des données géométriques en 3DTiles.



Figure 4.6 : Visualisation d'un bâtiment à l'aide d'une représentation thématique, sous forme de graphe, et d'une représentation spatiale, sous forme de géométrie 3D, issues de données CityGML. Ces représentations sont liées, permettant d'interagir entre elles.

Grâce aux données sémantiques, il est possible d'obtenir de nouvelles informations, comme les relations entre entités. Dans la figure 4.7, le graphe visualisé correspond à la hiérarchie des entités que l'on peut retrouver dans un fichier IFC. Le noeud bleu correspond au bâtiment affiché. Les noeuds orange correspondent aux étages de celui-ci, et les noeuds rouges sont les fenêtres appartenant à chaque étage. Il est possible d'interagir entre les représentations thématiques sous forme de graphe et les représentations spatiales issues de la même source grâce aux liens établis entre ces représentations. Lors de la sélection d'un noeud, ici un étage, les représentations spatiales de celui-ci et de l'ensemble des entités le constituant sont mises en évidence. Dans cet exemple, l'utilisateur peut choisir de visualiser un bâtiment, un des étages le constituant ou bien une entité unique distincte, comme un mur ou une fenêtre.

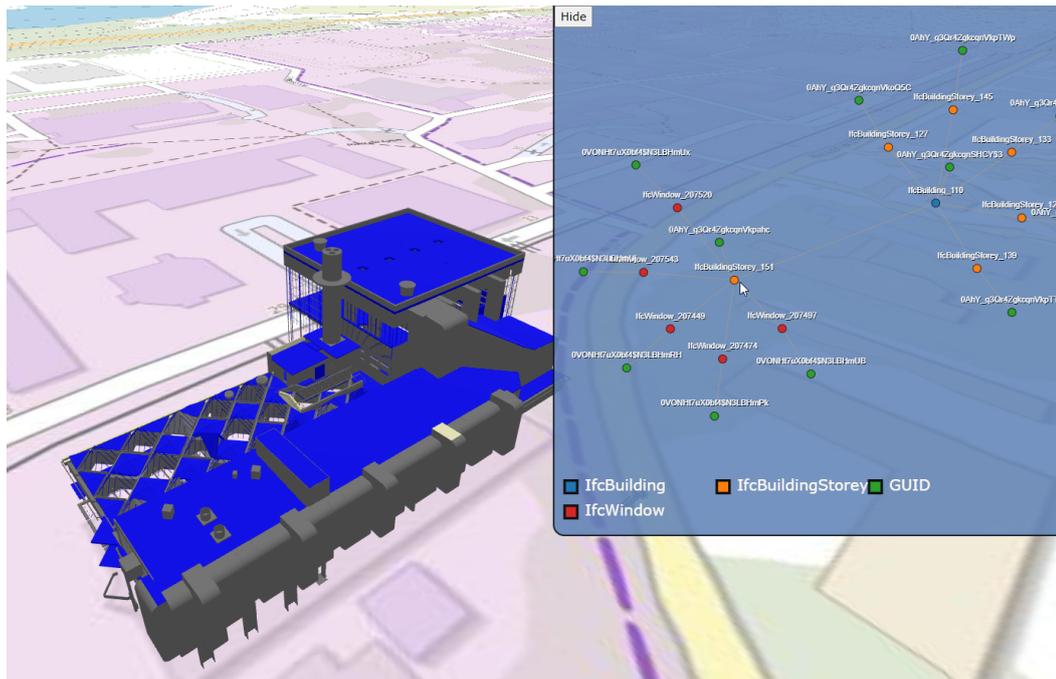


Figure 4.7 : Visualisation d'un bâtiment à l'aide d'une représentation thématique, sous forme de graphe, et d'une représentation spatiale, sous forme de géométrie 3D, issues de données IFC. Le graphe représente les entités qui en composent une autre (ici, un bâtiment). La géométrie de l'étage correspondant au noeud survolé est coloré en bleu.

4.3 Discussion

La méthodologie présentée permet une intégration des données géométriques et sémantiques géospatiales. L'intégration utilise des technologies et des formats de données conçus pour structurer, interroger et visualiser des données géométriques et sémantiques. Elle permet de disposer d'une solution générique et unifiée pour stocker et requêter les données sémantiques, indépendamment de leur modèle de données. L'intégration assure que les informations présentes originellement sont conservées, grâce à l'utilisation d'une méthodologie d'intégration des données sémantiques basée modèles. Nous avons démontré qu'il est possible de limiter le poids des graphes résultants en supprimant les données géométriques, qui reste l'une des principales problématiques associées à l'utilisation de graphes pour intégrer des données sémantiques. L'intégration des modèles de données et des données sous forme de graphes permet de stocker le résultat d'analyse et de traitement de ces données, tel que l'entity linking pour retrouver les différentes représentations d'une

même entité, ou l'exploitation de la base de connaissances pour créer ou vérifier de l'information.

L'édition et la synchronisation des données nécessitent que l'ensemble des acteurs d'un même projet CIM ou BIM travaillent avec les données transformées résultantes. L'actualisation des données depuis de nouvelles acquisitions pose question. Dans le cas de temps long, par exemple la création de nouvelles maquettes CIM tous les 3 ans de la métropole de Lyon par le GrandLyon, il est envisageable de simplement effectuer à nouveau la méthodologie pour ajouter les nouvelles données. Dans le cas d'acquisition de données sur un temps court, par exemple à l'aide de capteurs, des solutions semblables au dynamizer [CK16 ; KCK20] sont à envisager pour permettre de référencer ces données depuis leurs sources. Par exemple, grâce à cette solution, la valeur du champ "température" d'une pièce devient une référence vers une source externe contenant l'historique des valeurs, plutôt qu'une valeur unique.

Il existe des standards de graphe de connaissance formels, comme GeoSPARQL, stRDF ou encore BimSPARQL, qui fournissent des structures efficaces pour stocker, combiner et interroger des données sémantiques hétérogènes [Mal+20]. Transformer les données BIM et CIM intégrées dans la base de connaissances vers ces formats peut permettre de simplifier leur interrogation et analyse en fournissant un vocabulaire et un langage commun à l'ensemble des données.

La contribution présentée dans ce chapitre permet de répondre à la problématique P1 présentée dans la section 1.2 et de créer une vue contextuelle de données sémantiques et géométriques géospatiales hétérogènes dans un même contexte web. Il est possible de visualiser l'ensemble des représentations spatiales et thématiques apportées par des données géospatiales. La navigation est possible entre les représentations issues d'une même source et d'éventuelles représentations externes associées. Cependant, la navigation entre l'ensemble des représentations d'une même entité est limitée. Par exemple, il est impossible d'accéder à une représentation thématique issue de données CIM depuis une représentation spatiale d'une même entité issue de données BIM. L'utilisateur n'a donc pas accès à l'ensemble des informations disponibles dans différentes sources à propos d'une même entité. L'étude de cette problématique met en avant la nécessité d'une conceptualisation des différentes représentations d'une même entité pour permettre de les décrire génériquement et de travailler avec de manière efficace dans le but d'obtenir une navigation centrée sur les entités. Nous présentons dans le prochain chapitre cette conceptualisation ainsi qu'une solution pour référencer, décrire et naviguer dans les différentes représentations d'une même entité issues de différentes sources.

Gestion des multiples représentations d'une même entité

De plus en plus de données sont créées pour représenter la ville et le territoire par différents acteurs. Il est même possible de trouver plusieurs jumeaux numériques d'une même ville et d'objets urbains dans des univers professionnels distincts. Une même source de données peut aussi apporter plusieurs représentations thématiques ou spatiales d'une même entité, par exemple en utilisant plusieurs sémiologies graphiques différentes. Grâce aux contributions précédentes, ces représentations peuvent être visualisées dans un même contexte pour créer une vue unifiée de ces données. Répertorier et décrire l'ensemble des entités ainsi qu'identifier la variété de leurs représentations présentes dans de multiples sources de données peut aider à la compréhension de la ville et d'objets urbains et faciliter la navigation de l'utilisateur dans les données.

Dans ce chapitre, nous répondons à la deuxième problématique énoncée dans la partie 1.2 : Comment peut-on référencer et naviguer dans un ensemble de représentations existantes d'une même entité issues de sources hétérogènes ? Nous présentons notre contribution C3 : une formalisation plus complète du concept de **Variant** -instance ou ensemble d'instances représentant la même entité- et de **Variant Identifiant** -solution pour décrire la localisation et l'accès à un Variant-.

Nous présentons dans un premier temps le concept de **Variant** défini par [Bec+21] pour formaliser les représentations d'une même entité et ses limites. Nous proposons ensuite une définition plus précise et une formalisation plus complète de ce concept. Dans un second temps, nous présentons le concept de **Variant Identifiant**, permettant de référencer et naviguer dans un ensemble de variants d'une entité. Enfin, nous illustrons ces concepts à l'aide d'une démonstration.

5.1 Variant : Définition et formalisation des concepts

Pour étudier les différentes représentations d'une même entité que l'on peut retrouver dans des données géospatiales, nous avons besoin de les formaliser en détaillant ce qui les constitue. On retrouve le concept de Variant, proposé par [Bec+21], qui établit une première conceptualisation de ces représentations. Nous présentons ce concept, ainsi que des observations relatives à des différentes représentations que l'on peut retrouver dans des données géospatiales. Nous proposons une définition plus précise, basée sur ces observations.

5.1.1 Définition initiale

De part la multitude de données créées par un nombre croissant d'acteurs, on observe l'apparition de multiples représentations d'une même entité dans différentes sources de données. Par exemple, on retrouve usuellement plusieurs maquettes BIM d'un même bâtiment, créées par les différents acteurs participants à la création de ce bâtiment. Cet exemple est illustré dans la figure 5.1 : un même bâtiment est représenté par trois instances de classe building du modèle CityGML. Ces trois instances sont hébergées dans un même datastore, ici, une base de donnée 3DCityDB.

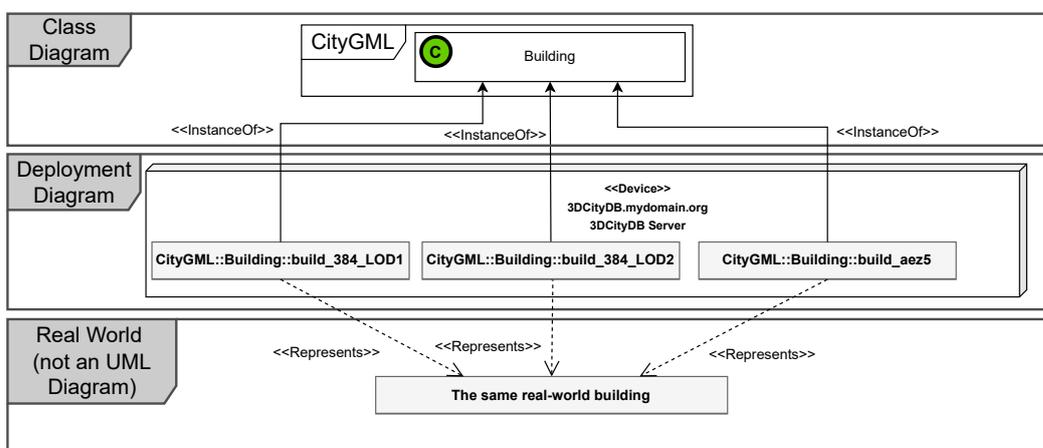


Figure 5.1 : Exemple 1 : Multiples instances (Deployment Diagram) issues d'un même modèle (Class Diagram) et représentant une même entité géospatiale (Real World).

[Bec+21] ont représenté ce phénomène grâce au concept de variants, illustré dans la figure 5.2.

Définition initiale de variants : ensemble d'instances représentant la même entité

géospatiale.

Cette définition est basée sur l'observation que différentes instances d'un même modèle peuvent représenter la même entité.

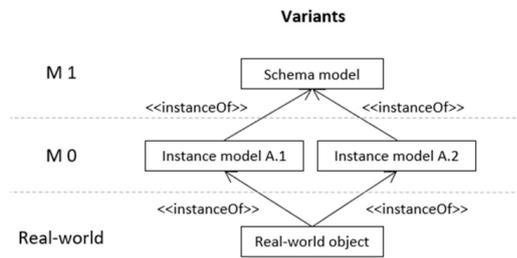


Figure 5.2 : Modèle logique de variants selon [Bec+21]

Nous proposons d'élargir la définition du concept de Variant en observant qu'une même entité peut être représentée par plusieurs instances structurées mais également selon différents modèles. Dans l'illustration que nous proposons 5.3, on peut retrouver différentes représentations d'un même bâtiment, qui sont des instances de modèles : deux instances de la classe Building du modèle CityGML dans une base 3DCityDB, une instance (image) dans une multimédiaDB, et une instance de la classe IFCBuilding du modèle IFC dans un fichier IFC.

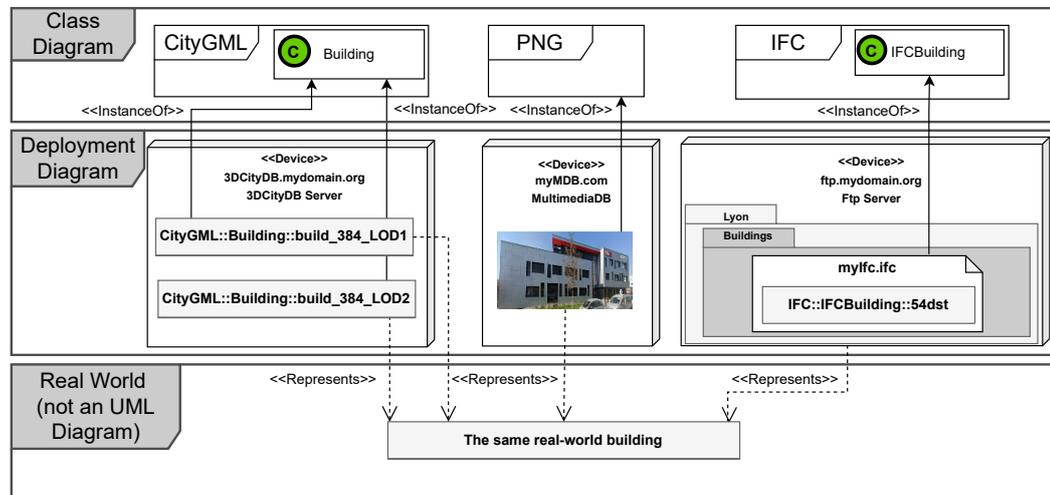


Figure 5.3 : Exemple 2 : Multiples instances (Deployment Diagram) issues de modèles hétérogènes (Class Diagram) provenant de diverses sources de données et représentant une même entité géospatiale (Real-World Diagram) Les instances de gauche sont des instances de bâtiment CityGML LOD1 et LOD2. Celle du centre est une image. Celle de droite est une instance de la classe IFCBuilding.

Dans le monde géospatial, une entité peut être elle-même représentée par un ensemble d'entités, c.-à-d. des ensembles d'instances, sans que le modèle de données ne spécifie obligatoirement cette relation. Une ville peut être caractérisée par un ensemble de bâtiments CityGML, un bâtiment peut être un ensemble de points dans un nuage de points ou un ensemble de triangles dans une géométrie triangulée. Le regroupement de ces instances peut être considéré comme la composition d'une des représentations de cette entité. Dans la figure 5.4, deux instances de la classe Building et une instance de la classe Bridge du modèle CityGML hébergées dans une même base de données 3DCityDB représentent les entités composants la ville de Lyon, et donc, représentent la ville de Lyon.

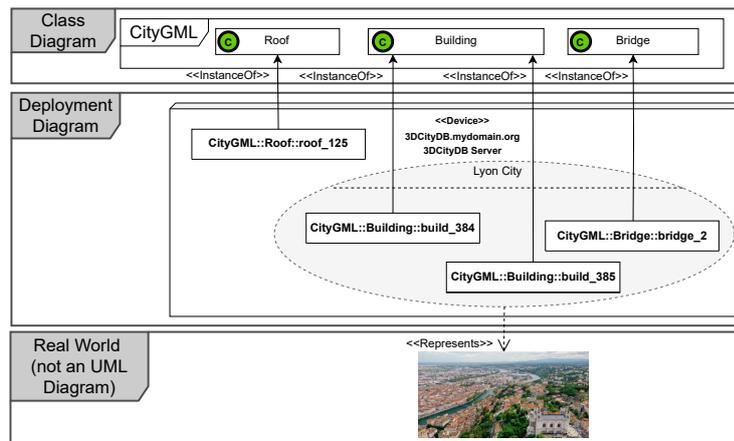


Figure 5.4 : Example 3 : Ensemble d'instances représentant une entité géospatiale par composition. Les instances sont hébergées par un même serveur et sont décrites par le modèle CityGML. L'ensemble des instances représentant des bâtiments et un pont composent une représentation de la ville de Lyon.

Le concept de Variant a pour but d'aider à travailler avec les différentes représentations disponibles d'une entité géospatiale afin de fournir davantage d'informations à son sujet. Ces représentations sont des instances décrites par des modèles de données. Connaître les modèles de données qui les décrivent n'est pas suffisant pour travailler efficacement (liaison, intégration, navigation) avec eux au niveau de l'instance car ces instances peuvent être hébergées sur différents supports de stockage informatique, sous différents formats et dans différents langages numériques. La source de données (par exemple, une base de données ou un fichier) dans laquelle les instances sont hébergées joue un rôle clé dans leur utilisation.

5.1.2 Définition proposée

Compte tenu des observations précédentes, nous proposons une nouvelle définition du concept de Variant pour exprimer plus précisément son périmètre sémantique.

Définition proposée de Variant : Instance ou ensemble d'instances de modèle hébergées dans une source de données (il peut s'agir d'une base de données, d'un fichier, d'un service, etc.) représentant différemment la même entité géospatiale. Les variants d'une même entité géospatiale composent donc différentes représentations de cette même entité.

Cette définition est illustrée par le modèle logique de la figure 5.5 : les deux instances i1 et i2, décrites par le modèle A, représentent la même entité géospatiale que l'instance i1 du modèle B.

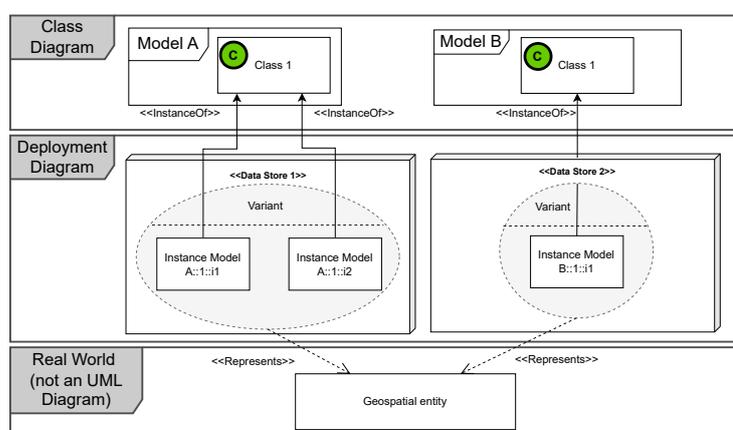


Figure 5.5 : Modèle logique de Variant proposé : Instance ou ensemble d'instances hébergée dans une source de données (base de donnée, fichier, service etc.) représentant la même entité géospatiale différemment.

Il peut arriver qu'une maquette numérique d'une entité soit hébergée dans plusieurs sources. Par exemple, les projets BIM sont développés par plusieurs parties prenantes qui travailleront sur différentes parties du projet. Même si elles partagent des similitudes, telles que l'identifiant ou l'architecture (même bâtiment, nombre d'étages et de pièces), chaque instance de maquette d'une des sources doit être considérée comme un variant de l'entité. Il s'agit toujours d'instances indépendantes qui représentent la même entité, mais qui vivent dans des sources différentes, bien qu'il soit toujours possible de synchroniser les attributs entre ces variants.

En observant les variants issus de données géospatiales, on retrouve couramment des entités qui sont représentées par les entités qui les composent. Nous proposons la définition suivante :

Définition proposée de Variant Complexe : Ensemble d'instances représentant des entités différentes qui composent une entité parente (figure 5.6). Ces instances doivent être toutes issues d'un même modèle de donnée et d'une même source.

Chaque instance doit être issue d'une classe décrite explicitement par un modèle de données comme une représentation d'une entité géospatiale, comme par exemple des classes IFC (IfcDoor, IfcWall) ou CityGML (building, wall).

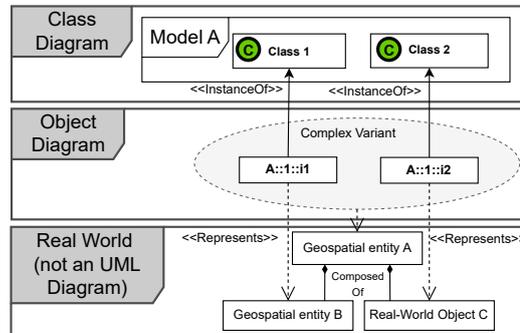


Figure 5.6 : Modèle logique de Variant Complexe proposé : Ensemble d'instances représentant des entités différentes qui composent une entité parente. Deux instances de deux classes du même modèle A qui représentent respectivement les entités B et C représentent l'entité A par composition.

L'ensemble de ces instances, implicitement des entités représentées par celle-ci, peuvent représenter une entité, phénomène décrit comme *complex feature* [ISO15b]. On nommera l'ensemble de ces instances comme *variant complexe* de l'entité parente représentée. On peut retrouver les informations de composition dans les données. Par exemple, les modèles IFC et CityGML décrivent des entités de façon hiérarchique : un bâtiment est un ensemble d'étages, puis de salles, ou un bâtiment est un ensemble de murs et d'espaces. Cette composition des entités ne doit pas être obligatoirement décrite par le modèle de donnée : par exemple, la notion de ville n'existe pas dans le modèle CityGML, pourtant, un ensemble de bâtiments décrit avec ces modèles peuvent former une ville. Un ensemble de triangles ou un ensemble de points dans un nuage de points ne peuvent pas être définis comme variant complexe car chaque instance de point ou de triangle ne représente pas d'entité en particulier.

5.2 Navigation dans les variants d'une entité géospatiale

Notre objectif est de pouvoir naviguer dans un ensemble de variants d'une entité. Il est nécessaire de proposer des solutions afin d'accéder à ces variants pour les afficher ou interagir avec. Une solution générique à tout variant est requise afin de permettre de les identifier, les référencer et de les lier aux entités qu'ils représentent. Les variants pouvant être hébergés dans des sources hétérogènes, cette solution doit décrire les sources et les méthodes d'accès à un variant, par exemple, l'identifiant ou la requête nécessaire pour récupérer le variant. Nous proposons le concept de **Variant Identifieur**, composé d'un **Variant Accessor** et d'un **Variant Locator**, pour décrire et accéder à un variant donné afin de permettre une navigation entre différentes représentations d'une même entité.

5.2.1 Modèle conceptuel de Variant Identifieur

Référencer un variant, qui est une instance ou un ensemble d'instances hébergées dans une source de donnée, requiert plusieurs informations pour y avoir accès, l'afficher et interagir avec. Premièrement, la source de données doit être spécifiée pour savoir où récupérer le variant. Deuxièmement, l'accès au variant doit être décrit, à l'aide d'identifiants, de requêtes ou méthodes pré-définies. Nous proposons le concept de *Variant Identifieur*, illustré dans la figure 5.7, qui rassemble l'ensemble des informations nécessaires pour décrire et référencer un variant. Il est composé d'un *Variant Locator*, qui décrit la source de donnée qui héberge le variant. Un *Variant Locator* peut être partagé entre plusieurs *Variant Identifieurs*, car une même source de donnée peut contenir différents variants de différentes entités. Il est composé d'un *Variant Accessor*, qui décrit le variant afin de récupérer l'instance ou les instances le composant depuis sa source. Celui-ci aussi peut être partagé entre différents *Variant Identifieurs*. Il peut s'agir d'une requête permettant d'extraire des instances spécifiques d'une base de données. Il peut s'agir d'un identifiant ou d'un ensemble d'identifiants avec le nom de classe de cet identifiant pour extraire les instances nécessaires d'un fichier. Un *Variant Accessor* dépendra souvent du modèle de donnée sous-jacent et du type de donnée (géométrique ou sémantique). Un variant peut être référencé par un *Variant Identifieur*.

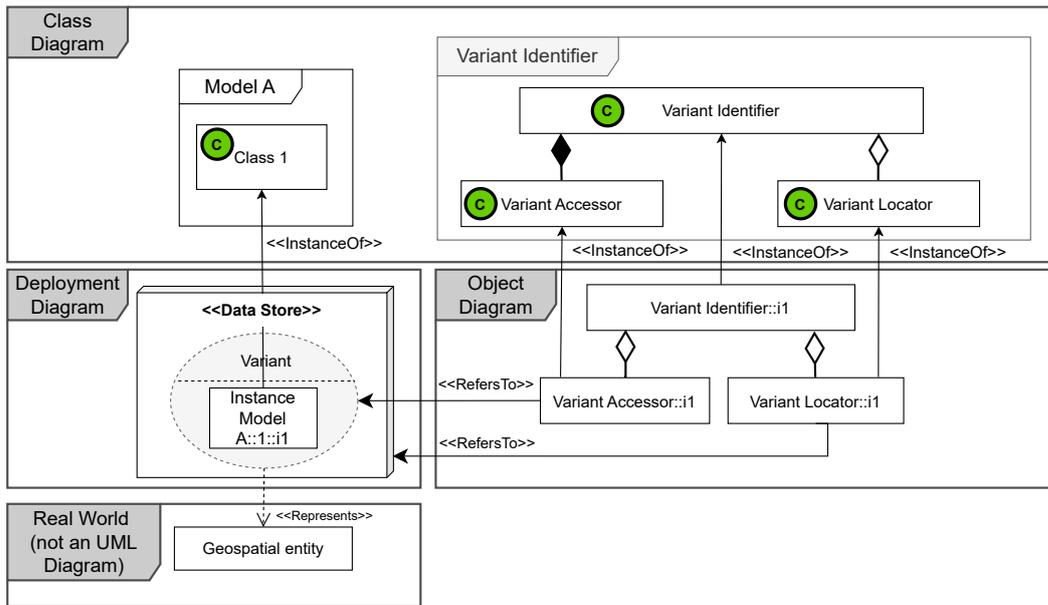


Figure 5.7 : Modèle logique de Variant Identifier : Un Variant Identifier permet de référencer et décrire un variant. Il se compose d'un *Variant Locator* qui décrit la source de données et d'un *Variant Accessor* qui décrit comment récupérer l'instance ou l'ensemble d'instances à partir de cette source.

Puisqu'un Variant peut être une représentation de différents types (sémantique, géométrique, multimédia), le Variant Identifier peut spécifier le type de représentation qu'il contient. Certains variants peuvent être une représentation spatiale, parfois géolocalisée, utilisant des données géométriques qui ont besoin d'être interprétées pour être correctement affichées. L'ajout d'informations spécifiant la géolocalisation d'une entité et le type de représentations de ses variants peut aider à les afficher. Cela permettrait à un utilisateur ou à une application de rechercher des entités géospatiales situées dans une zone spécifique. La consultation et visualisation des variants dépend de la capacité des applications utilisant la solution proposée. Par exemple, une géométrie 3D identifiée comme variant d'un bâtiment ne peut être visualisée que si l'application utilisant la solution proposée a la capacité de visualiser.

5.2.2 Modèle logique pour naviguer dans les variants

Le concept de Variant Identifier permet de lier un ensemble de variants d'une entité. Nous proposons d'utiliser une abstraction de cette entité pour réunir ces références afin de naviguer dans l'ensemble des représentations d'une entité. Cela facilitera la navigation en donnant la possibilité d'ajouter des informations sur l'entité

géospatiale, telles que son nom, son emplacement, etc. Cela permet de créer des relations entre les entités qui peuvent être extraites des variants de ces entités. Cela rend possible de créer des liens entre variants lorsque leurs modèles sous-jacents ne permettent pas de les créer. Par exemple, le modèle .Obj, traditionnellement utilisé pour représenter des données 3D ne permet pas l'ajout de données sémantiques.

Grâce aux Variants Identifiers, il est possible de référencer les variants d'entités existants issus de données géospatiales intégrées avec la méthode précédemment présentées, comme illustré dans la figure 5.8. L'instance "Building" de la classe "Abstraction d'Entité" du modèle proposé de "Variant Identifier" représente le bâtiment de Carl-Software. Cette abstraction de ce bâtiment contient l'ensemble des variants identifiants qui référencent les variants disponibles de cette entité : les instances Bdg :1 et Building :548 des modèles MD1 et MD2, les instances de données géométriques 1 et 548 hébergées dans le datastore des représentations spatiales, et un multimédia hébergé dans un datastore externe. Pour une entité donnée, il est possible de trouver des variants dans les graphes créés à partir de données sémantiques géospatiales, dans les données géométriques géospatiales et dans des sources de données externes. Seule l'abstraction de l'entité étant responsable de garder les liens avec l'ensemble des variants, il n'est plus nécessaire de modifier les sources externes non spatiales pour y ajouter des liens vers des représentations spatiales, celle-ci peuvent vivre indépendamment de l'application finale. En référençant les variants existants, ceux-ci peuvent évoluer dans leur contexte initial et être consultés sans ingérence. Cela permet également d'éviter l'ajouts de données dans des formats non-optimaux pour celles-ci : les données géométriques et les données sémantiques peuvent être complètement séparées, structurées par des modèles optimaux. Les liens entre les variants permettent de naviguer entre ces représentations hétérogènes.

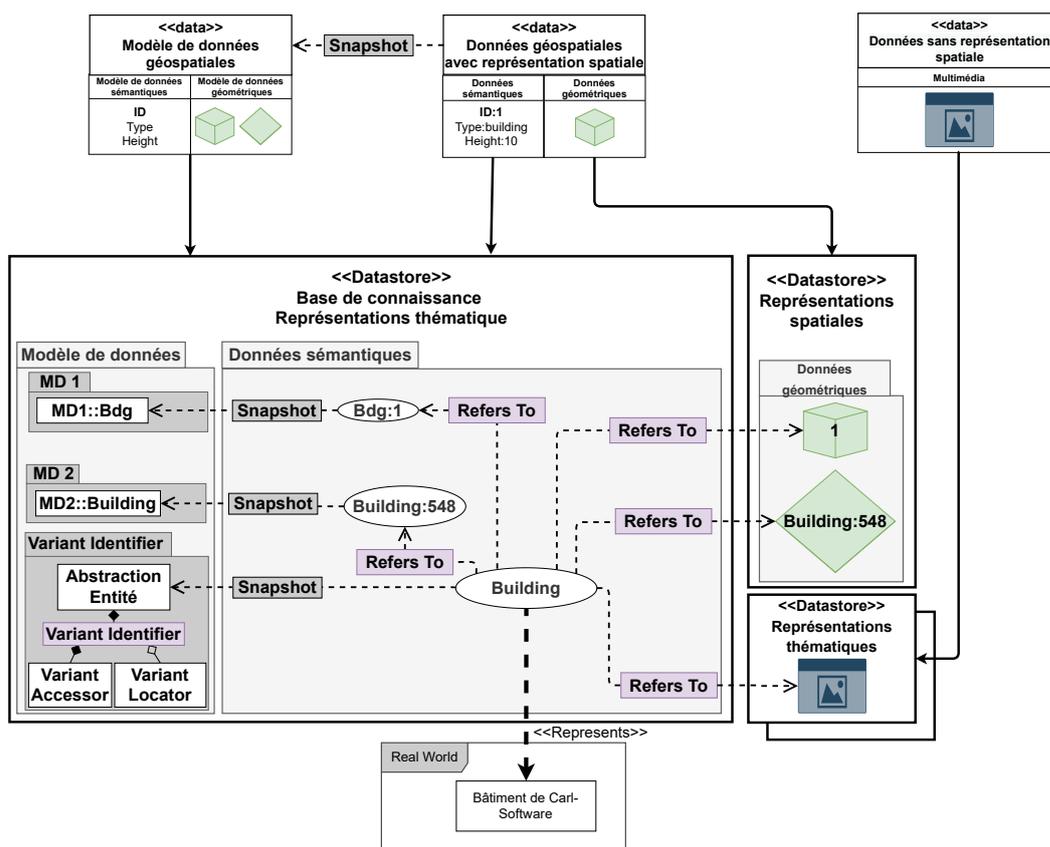


Figure 5.8 : Modèle logique d'abstraction d'une entité géospatiale pour naviguer entre ses variants à l'aide de Variant Identifiers

Le fait de rendre possible la navigation entre les variants signifie qu'ils doivent être liés. Les utilisateurs, les systèmes et les applications peuvent utiliser ces liens pour explorer et accéder aux informations apportées par les variants d'une entité.

Il incombe au gestionnaire des *Variants Identifiers* de garantir la consistance des variants, c'est-à-dire de s'assurer qu'ils représentent tous la même entité. La solution la plus simple consiste à laisser un utilisateur, disposant de préférence des droits et de l'expertise nécessaires, de spécifier les variants d'une entité à la main. Toutefois, la consistance des variants peut être assurée par des méthodes automatiques.

5.3 Démonstration et résultats

Dans cette section, nous présentons la démonstration mise en place pour illustrer le concept de **Variant** et de **Variant Identifier** pour permettre une navigation entre

les différentes représentations d'une même entité. Nous commençons par présenter l'implémentation du modèle abstrait de Variant Identifier à l'aide d'une extension au modèle 3D-Container de l'OGC, puis l'architecture technique utilisée, et enfin, nous présentons les résultats obtenus à l'aide de cette démonstration.

5.3.1 Implémentation à l'aide d'une extension au modèle OGC 3D-Container

Afin de démontrer le concept de Variant et de l'utilisation de Variant Identifier pour permettre une navigation centrée entité, nous proposons une extension au modèle 3D-Container de l'OGC [MTL20], présenté dans le chapitre 2. Le modèle de données 3D-Container est présenté en blanc dans la figure 5.9. La classe **3DContainer** correspond à l'abstraction d'une entité géospatiale. Elle possède un identifiant ainsi qu'un extant spatial qui nous permettra de localiser l'entité représentée dans un contexte géospatial. Chaque 3DContainer peut avoir plusieurs **content**, décrits par la classe **LinkToVariant** (en gris dans la figure 5.9). Cette nouvelle classe correspond au **Variant Identifier** : l'attribut **href** correspond au **Variant Locator** et nous avons ajouté un attribut **accessor** qui correspond au **Variant Accessor**. Ce modèle logique de données implémente le modèle conceptuel décrit précédemment dans la figure 5.7.

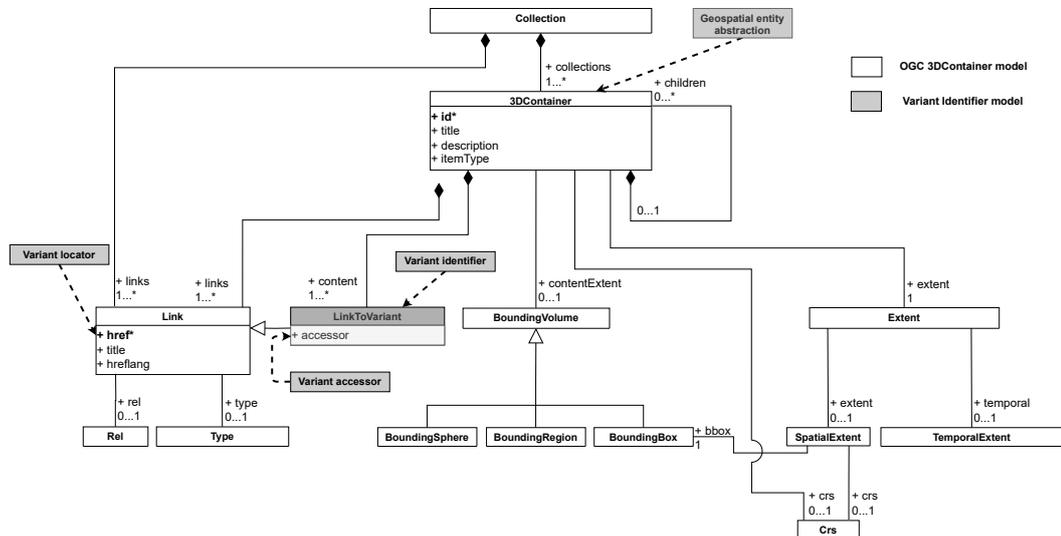


Figure 5.9 : Extension du modèle 3D-Container (en blanc) à l'aide du modèle de Variant Identifier (en gris). L'attribut *href* correspond au *Variant locator*. La nouvelle classe *LinkToVariant* hérite de la classe *link* et correspond au *Variant Accessor*. La classe *3DContainer* correspond à une abstraction d'une entité géospatiale.

5.3.2 Architecture technique

La figure 5.10 illustre l'architecture technique utilisée pour la démonstration. Les modèles de données IFC et CityGML et les données sémantiques de ces modèles sont transformés en graphe et importés dans une base de connaissances BlazeGraph à l'aide des outils UD-Graph et IFctoRDF. Les données géométriques issues de données IFC, CityGML, GeoJSON et LAS/LAZ (formats de fichier de nuage de points) sont transformées vers le format pivot 3DTiles grâce à l'outil Py3DTilers. Par rapport à la démonstration précédente, on peut noter l'ajout de données 3D-Container (en gris) qui permettent de décrire les entités représentées par les données utilisées, et de référencer les variants que l'on y retrouve. Ces données sont ajoutées et accessibles dans la base de connaissances sous forme de graphe.

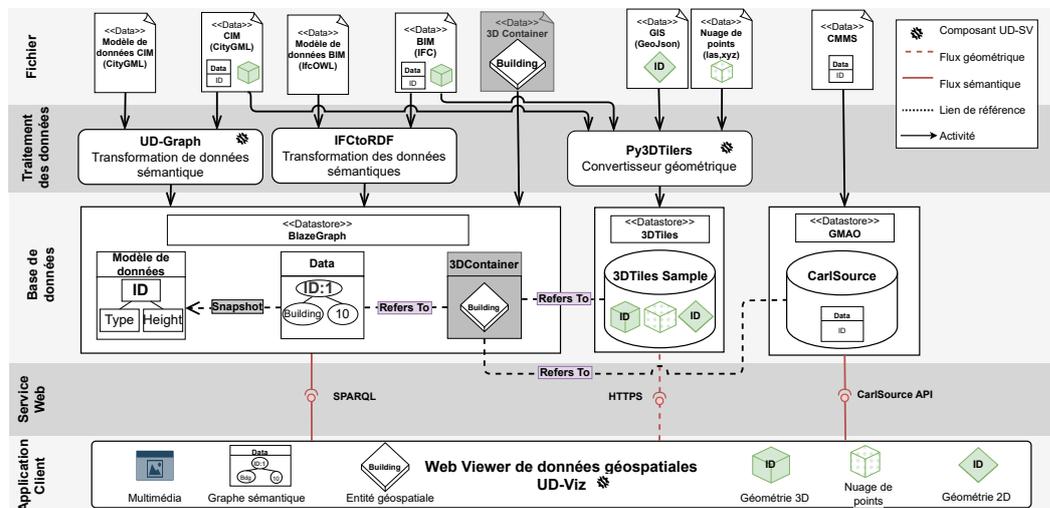


Figure 5.10 : Architecture technique. Les données 3DContainer permettant de référencer les variants d'entités (en gris) ont été ajoutés par rapport à la précédente contribution.

A l'aide de cette démonstration, nous cherchons à illustrer la navigation centrée entité, c.-à-d. la capacité à identifier et référencer un ensemble de représentations d'une même entité. Pour ce faire, les entités utilisées et leurs variants sont listés dans la figure 5.11. On retrouve les entités suivantes : la métropole de Lyon, les villes de Limonest et Villeurbanne et deux bâtiments de ces villes. Les Variants Identifiants utilisés permettent d'accéder à des représentations spatiales ou thématiques, hébergées dans des sources hétérogènes de données. Les Variants Accessors utilisés sont décrits comme : un fichier, un ensemble de fichiers, une instance, un ensemble d'instances ou une zone spatiale correspondant à l'entité. Les variants ont été sélectionnés et décrits manuellement dans cette démonstration.

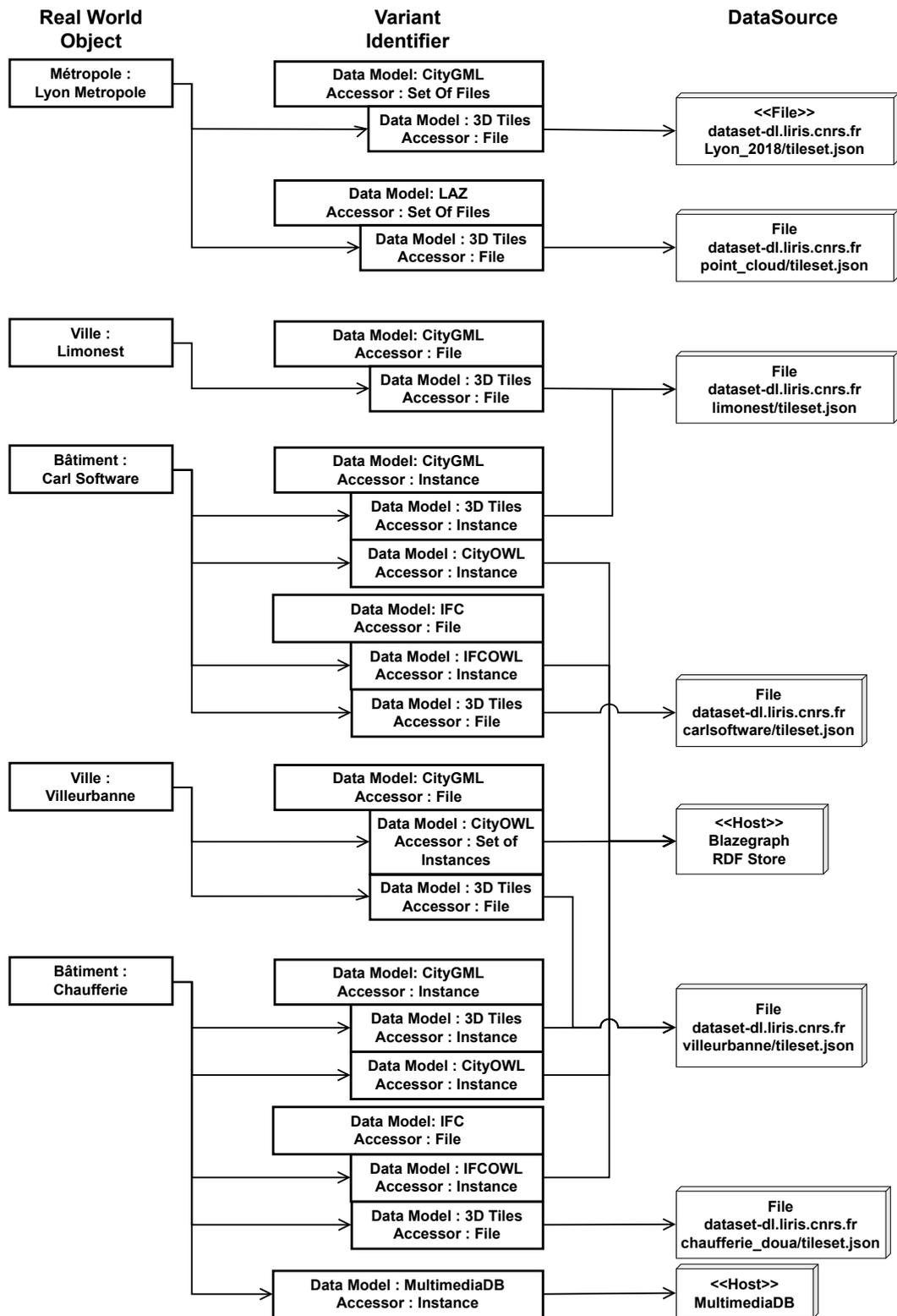
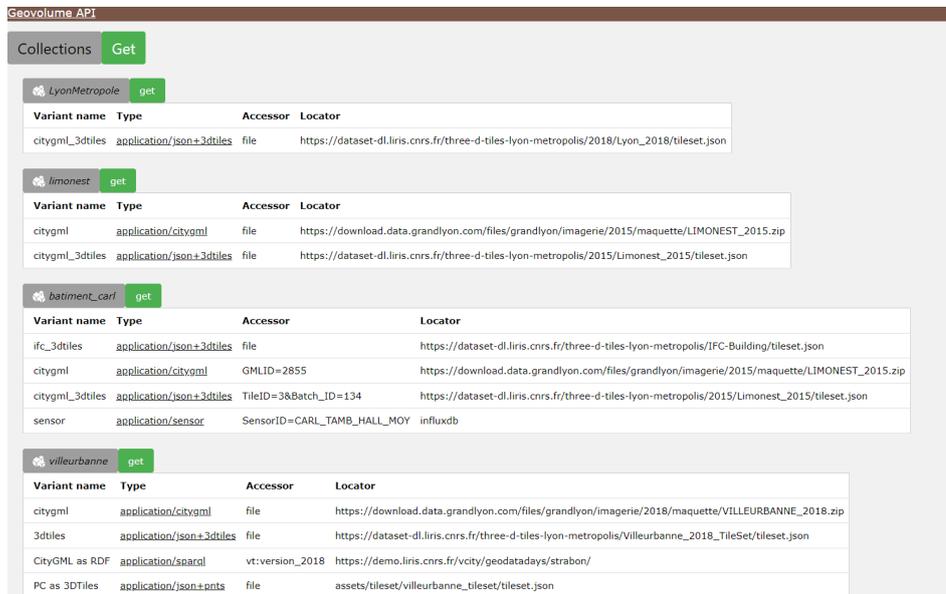


Figure 5.11 : Descriptions des entités géospatiales et leurs variants utilisés dans la démonstration.

5.3.3 Résultat

Il est possible de demander et de consulter les entités géospatiales répertoriées et leurs variants à l'aide des données 3D-Container. Toutes ces entités et leurs variants identifiants peuvent être visualisés dans le même contexte web, comme le montre la figure 5.12.



The screenshot shows the Geovolume API interface with a 'Collections' tab selected. It displays four collections: LyonMetropole, limonest, batiment_carl, and villeurbanne. Each collection has a 'get' button and a table of variants with columns for Variant name, Type, Accessor, and Locator.

Collection	Variant name	Type	Accessor	Locator
LyonMetropole	citygml_3dtiles	application/json+3dtiles	file	https://dataset-dl.liris.cnrs.fr/three-d-tiles-lyon-metropolis/2018/Lyon_2018/tileset.json
limonest	citygml	application/citygml	file	https://download.data.grandlyon.com/files/grandlyon/imagerie/2015/maquette/LIMONEST_2015.zip
	citygml_3dtiles	application/json+3dtiles	file	https://dataset-dl.liris.cnrs.fr/three-d-tiles-lyon-metropolis/2015/Limonest_2015/tileset.json
batiment_carl	ifc_3dtiles	application/json+3dtiles	file	https://dataset-dl.liris.cnrs.fr/three-d-tiles-lyon-metropolis/IFC-Building/tileset.json
	citygml	application/citygml	GMLID=2855	https://download.data.grandlyon.com/files/grandlyon/imagerie/2015/maquette/LIMONEST_2015.zip
	citygml_3dtiles	application/json+3dtiles	TileID=3&Batch_ID=134	https://dataset-dl.liris.cnrs.fr/three-d-tiles-lyon-metropolis/2015/Limonest_2015/tileset.json
	sensor	application/sensor	SensorID=CARL_TAMB_HALL_MOY	influxdb
villeurbanne	citygml	application/citygml	file	https://download.data.grandlyon.com/files/grandlyon/imagerie/2018/maquette/VILLEURBANNE_2018.zip
	3dtiles	application/json+3dtiles	file	https://dataset-dl.liris.cnrs.fr/three-d-tiles-lyon-metropolis/Villeurbanne_2018_TileSet/tileset.json
	CityGML as RDF	application/sparql	vt:version_2018	https://demo.liris.cnrs.fr/vcity/geodatadays/strabon/
	PC as 3DTiles	application/json+3dtiles	file	assets/tileset/villeurbanne_tileset/tileset.json

Figure 5.12 : Page d'accueil de l'API Geovolume. Les entités géospatiales et leurs variants sont listés grâce à leurs représentations sous forme de 3D-Container dans le serveur.

Les différents variants accessors utilisés sont les suivants : la valeur **file** signifie que le fichier entier référencé par le locator est le variant. On accède à une instance CityGML ou IFC en utilisant respectivement un **GMLID** ou un **GUID**. Le format 3DTiles permet d'associer des informations thématiques supplémentaires à chaque géométrie à l'aide d'une feature table. Il est possible de décrire un identifiant de Variant à l'aide de divers accessor pour les variants 3DTiles : en spécifiant une tuile à l'aide d'un **TileID**, ou une géométrie dans cette tuile en ajoutant avec son identifiant dans la tuile, le **BatchID**. Il est aussi possible de spécifier un identifiant à retrouver dans la feature table. Le dernier accessor utilisé, comme le montre la figure 5.16, est l'**extent** de cette entité, qui est utilisé sur les nuages de points pour spécifier que nous n'avons besoin que des points situés à l'intérieur de la boîte de délimitation.

La navigation de l'utilisateur dans le viewer UD-Viz est centrée entité grâce aux données 3D-Container, comme illustré dans la figure 5.13. L'ensemble des entités

présentes dans la zone visible par l'utilisateur est récupéré depuis la base de connaissances et affiché à l'aide de leurs noms (à gauche) et chaque cube (dans la vue à droite) correspond à une entité. Depuis cette interface, l'utilisateur peut choisir les variants qu'il souhaite afficher, issues de 3DTiles, de données graphes ou de données capteurs.

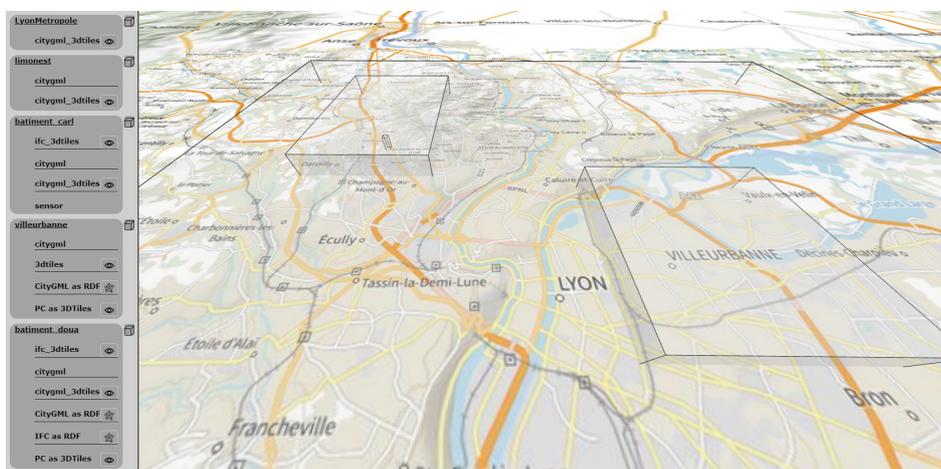


Figure 5.13 : Vue des entités sous UD-Viz. 5 entités géospatiales sont visibles : la métropole de Lyon, deux villes : Villeurbanne et Limonest, et deux bâtiments à l'intérieur de ces villes. Elles sont affichées sous forme de simples cubes géolocalisés dans la scène 3D (à droite) et sous forme de liste avec les variants disponibles pour chacune d'entre elles (à gauche).

La navigation est centrée entité, car l'utilisateur visualise dans un premier temps les entités existantes, puis choisit les représentations qui lui semble pertinentes pour lui apporter les informations qu'il requiert sur cette entité. Il peut composer sa vue contextuelle en choisissant quelles entités visualiser et comment les représenter. On peut voir dans la figure 5.14 qu'il est possible de visualiser des représentations spatiales et thématiques en simultanément. Les éléments 3D (au milieu) représentent la ville de Villeurbanne, de même que le graphe présenté à droite. Elles peuvent provenir de sources hétérogènes car celles-ci sont identifiées comme variants de l'entité qu'elles représentent.

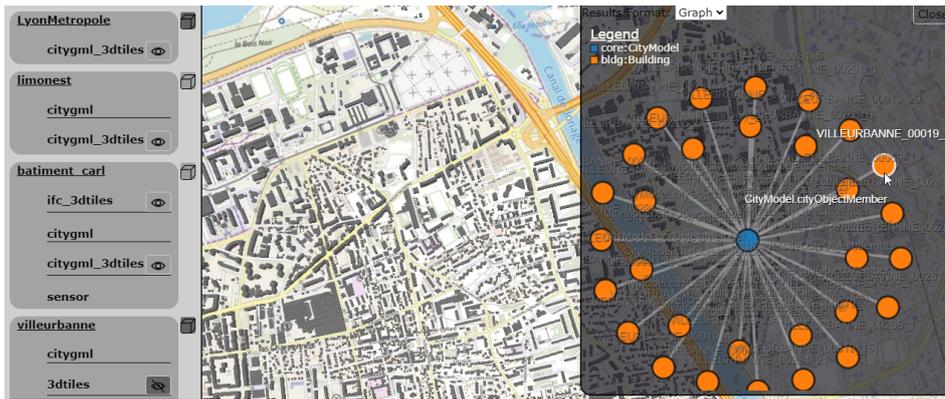


Figure 5.14 : Vue composée de multiples variants d’une même entité. Deux variants sont affichés dans la même vue de la ville de Villeurbanne : une représentation géométrique (au centre) et un graphe (sur la droite).

Il peut exister différentes représentations de différentes entités dans une même source de données. C’est le rôle du Variant Accessor de donner l’accès à un variant spécifique dans une source de données. La figure 5.15 illustre cette possibilité : depuis un même fichier 3DTiles, on peut accéder à un variant de la ville de Limonest (à gauche), l’entiereté de ce fichier, et accéder à un variant du bâtiment de CarlSoftware (à droite) à l’aide de son identifiant, étant une seule géométrie 3D.

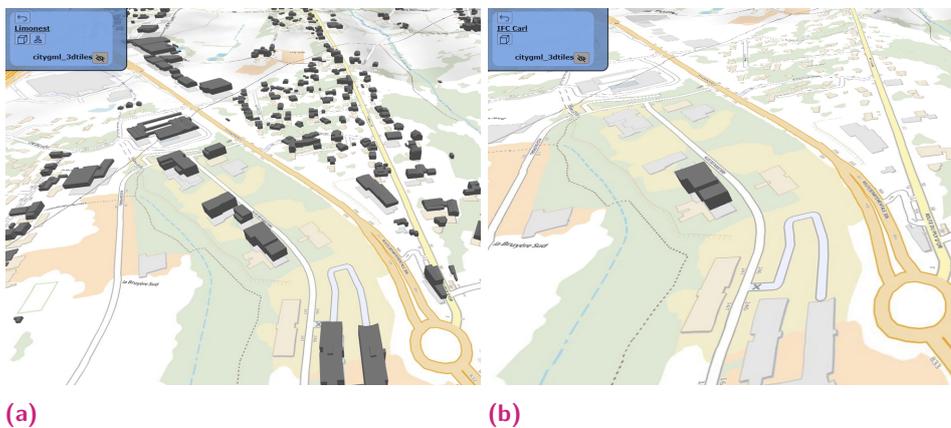


Figure 5.15 : Visualisation de deux variants de deux entités géospatiales distinctes à partir de la même source de données. Le variant de la ville de Limonest est un fichier 3DTiles entier (a). Le variant du bâtiment Carl Software est un objet géométrique (b) dans ce fichier, visualisé en utilisant son identifiant.

Un autre exemple de l’utilisation de Variant Accessor pour identifier différentes entités dans une même source est illustré dans la figure 5.16. Dans cet exemple, la ville de Villeurbanne dispose d’un variant qui est un fichier 3DTiles. La chaufferie

de la Doua est représentée par un sous-ensemble de ce 3DTiles limité à l'extent de l'entité représentée.

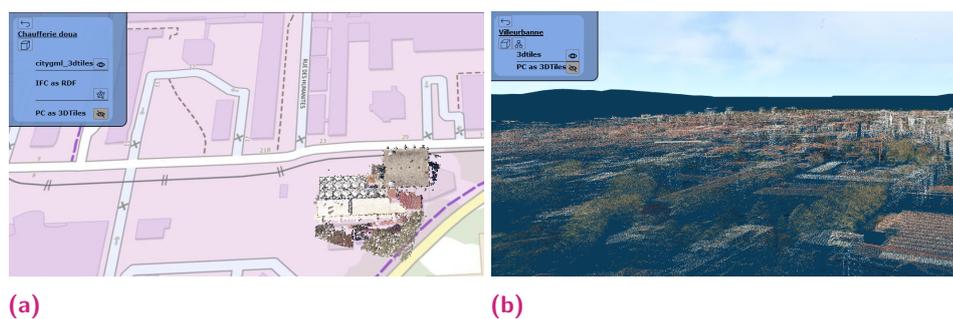


Figure 5.16 : Visualisation de deux variants de deux entités géospatiales distinctes à partir de la même source de données. Le variant de la ville de Villeurbanne est un fichier complet de nuage de points 3DTiles (a). Le variant d'un bâtiment à l'intérieur de la ville de Villeurbanne est l'ensemble de points contenu dans sa bounding box issu du même fichier 3DTiles (b).

Grâce aux Variants Identifiers, il est possible de référencer des variants existants dans des sources externes, sans modifier ces sources. Cela permet l'accès à de nouvelles sources, qu'il aurait été impossible de modifier pour créer un lien entre une entité et la représentation qu'elle apporte.

5.4 Discussion

Dans ce chapitre, nous avons proposé une définition plus précise et une formalisation plus complète du concept de *Variant* et proposé le concept de *Variant Identifier* pour référencer et naviguer dans un ensemble de représentations existantes d'une même entité issues de sources hétérogènes. L'élaboration de ces concepts vise à permettre l'étude générique des variants d'une entité. Une première illustration de cette étude est la définition possible de types de variant rencontré usuellement, comme un *Variant Complexe*. Les *Variants Identifiers* apportent une solution générique pour référencer et naviguer dans les variants d'une entité. Cette solution peut être appliquée dans toute application visant à naviguer dans des représentations hétérogènes d'une même entité existantes dans de multiples sources. Elle permet de référencer des variants existants dans tous types de sources et de les laisser perdurer si nécessaire dans leur contexte.

Des méthodes d'*entity linking* peuvent être appliquées pour identifier automatiquement les variants d'entités ou pour valider leur consistance. Grâce à la méthodologie

d'intégration de données géospatiales employée, les résultats de ces méthodes peuvent être conservés dans la base de connaissances.

Une source de données géospatiale contient usuellement des représentations d'un grand nombre d'entité. Par exemple, dans un IFC, pour un bâtiment moyen, on peut retrouver cinq à dix mille entités distinctes (mur, tuyau, dalle, ...). Une problématique apparaît alors : est-ce que toutes les entités décrites par une source doivent disposer d'une abstraction pour les lier à leurs variants ? Cela impliquerait que le nombre d'entité décrites et listées -ici par des 3D-Container- serait très grand. Pour l'instant, le choix a été fait de ne lister et décrire que les entités qui disposent de plusieurs variants distincts identifiés. Lors de l'ajout d'une nouvelle source de données, une méthode automatique pourrait être appliquée pour identifier les entités qu'elle représente et reconnaître si elles sont déjà représentées dans d'autres sources de données. Alors, on peut créer une abstraction de cette entité pour lister l'ensemble de ses variants.

La solution présentée permet une navigation centrée entité, dans laquelle l'utilisateur peut avoir accès aux informations disponibles sur une entité en ayant connaissance des diverses représentations existantes de cette entité. Celui-ci peut alors composer sa propre vue contextuelle en choisissant les représentations qu'il souhaite afficher. A l'aide du contexte de l'utilisateur, on pourrait choisir automatiquement la ou les représentations pertinentes pour lui, qui pourrait permettre de répondre à son besoin. Certains variants sont complémentaires, lorsqu'ils fournissent des représentations hétérogènes qui peuvent être visualisées en simultanée. D'autres variants sont exclusifs parce que leur visualisation simultanée serait source de confusion pour l'utilisateur. Par exemple, la visualisation simultanée de deux modèles 3D d'un même bâtiment les ferait se chevaucher et ne serait pas pertinente. Des méthodes pour permettre aux utilisateurs de visualiser efficacement des ensembles de variants d'une entité doivent être étudiées afin de proposer à l'utilisateur le plus grand nombre d'informations possible à propos de l'entité ciblée et de son contexte. Dans le prochain chapitre, nous allons nous intéresser à cette problématique en prenant en compte le contexte 3D de l'utilisateur : sa position dans l'espace et ce qu'il voit et souhaite voir.

Navigation multiscalaire

Nos contributions précédentes permettent de décrire les entités présentes dans des sources de données hétérogènes et de visualiser leurs variants, qui correspondent à diverses représentations spatiales et thématiques. Afin de répondre adéquatement aux besoins de l'utilisateur, son contexte doit être pris en compte en permettant ainsi d'identifier les entités qui l'intéressent. Ensuite, il est nécessaire de choisir comment représenter ces entités, c.-à-d., choisir parmi les représentations existantes celles qui sont pertinentes selon son contexte.

Dans ce chapitre, nous nous intéressons au contexte 3D lié à l'utilisateur : sa position dans la scène, ce qu'il voit et souhaite voir. Nous apportons des réponses à la problématique P3 : comment sélectionner puis représenter les entités à visualiser dans un environnement 3D en tenant compte du contexte de l'utilisateur ?

Dans le domaine de l'informatique graphique, pour une même entité, il est courant de créer différentes représentations spatiales 3D à différents niveaux de détails (LoD - Level of Detail) géométriques afin de pouvoir faire varier la représentation de l'entité en fonction du point de vue de la caméra. Plus l'entité est loin de la caméra, moins la représentation visualisée sera détaillée, et ce, afin de conserver une visualisation fluide. Les niveaux de détails (LoD) d'un modèle 3D sont dérivés à partir du niveau le plus complexe au sein d'une même source et sont sémantiquement annotés, ce qui crée plusieurs représentations spatiales ordonnées de la plus complexe à la plus simplifiée pour une même entité (voir 2.2.2). Dans notre contexte d'étude, les représentations d'une même entité pré-existent au sein de sources différentes, selon des modèles de données hétérogènes, qui disposent de leur propre définition de niveau de détail. Par exemple, un bâtiment représenté par un modèle 3D en LoD 1 d'une source peut être bien plus précis qu'un modèle 3D en LoD 10 d'une autre source. Les modèles de données géospatiales apportent des informations sur la composition d'une entité, par exemple, en définissant un bâtiment comme la composition de quatre murs et un toit, définie dans le chapitre 5 comme *variant complexe* d'une entité. Une représentation spatiale d'une entité est dans ce cas une composition de représentations spatiales d'autres entités, rendant la définition de son niveau de détail problématique. Un modèle 3D d'un bâtiment extrêmement

détaillé est-il plus complexe qu'un ensemble de modèles 3D simples représentant l'ensemble des murs, fenêtres et dalles de ce bâtiment ?

Afin de répondre à notre problématique, il est nécessaire de comprendre comment les entités sont pensées et décrites par les différents modèles de données géospatiales et de trouver des méthodes pour analyser et décrire l'ensemble des représentations spatiales d'une même entité à l'aide d'un système de niveau de détail indépendant du modèle d'origine et de l'échelle de l'entité représentée. Ce chapitre présente notre contribution C4 : dans la section 6.1, une étude des conceptualisations d'une entité selon les modèles IFC et CityGML et dans la section 6.2 la proposition d'une méthode permettant de sélectionner les entités à visualiser selon le contexte utilisateur et de décrire le niveau de complexité d'une représentation spatiale. La section 6.3 présente l'expérimentation mise en oeuvre pour illustrer cette méthode, les premiers résultats obtenus et une discussion autour de ces travaux.

6.1 Conceptualisation d'une entité : étude de cas des modèles CityGML et IFC

Pour rappel, une entité est définie par l'ISO 19119 [ISO16] comme quelque chose qui a une existence séparée et distincte et une réalité objective ou conceptuelle. Un objet est défini comme une entité dont les limites physiques et l'identité sont bien définies [ISO19a]. Pour représenter conceptuellement la ville et les entités urbaines, les modèles de données géospatiaux utilisent des descriptions génériques conceptuelles différentes, notamment grâce à la notion d'espace. L'objectif de cette étude est de pouvoir décrire génériquement les entités et les variants qui comprennent une représentation spatiale pour proposer une méthode de sélection et d'affichage en fonction de leur pertinence selon un contexte utilisateur donné. Pour ce faire, il est donc nécessaire de comprendre et comparer les différents concepts apportés par des modèles de données géospatiaux pour décrire une entité.

6.1.1 Modèle CityGML 3.0

Dans cette section, nous allons décrire le modèle CityGML 3.0 plus en détail afin de comprendre comment une entité est formalisée dans ce modèle. Le modèle CityGML 3.0¹ [OGC18a ; KGP05 ; KCK20] décrit un espace (AbstractSpace) comme une entité

1. <https://docs.ogc.org/is/20-010/20-010.html>

disposant d'une étendue volumétrique dans le monde réel. Les bâtiments, les plans d'eau, les arbres, les pièces et les espaces de circulation sont des exemples d'entités ayant une étendue volumétrique. Cette notion d'espace est décomposée en quatre ensembles (figure 6.1)² : les espaces logiques dans lesquels on retrouve les espaces physiques, constitués d'espaces occupés et non occupés.

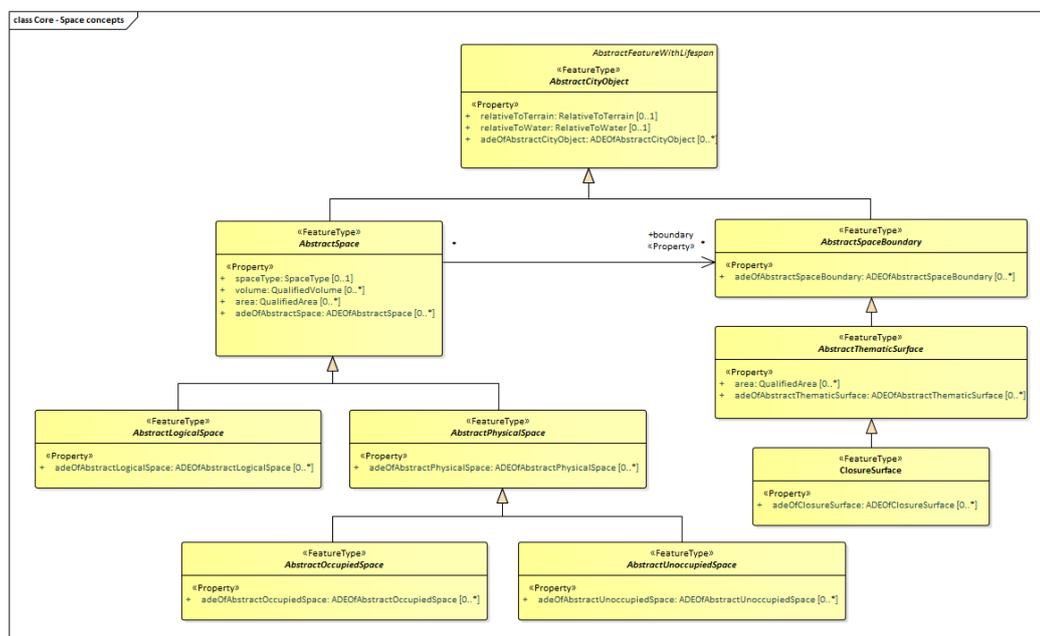


Figure 6.1 : Diagramme UML des concepts d'espaces dans CityGML 3.0 [OGC18a]

Un espace logique (figure 6.2) est défini en fonction de considérations thématiques. Il n'est pas nécessairement délimité par des objets physiques. Les délimitations de ces espaces sont définies par l'homme. Un quartier est un espace logique, car il est délimité par des frontières administratives.

2. <https://docs.ogc.org/is/20-010/20-010.html#toc30>

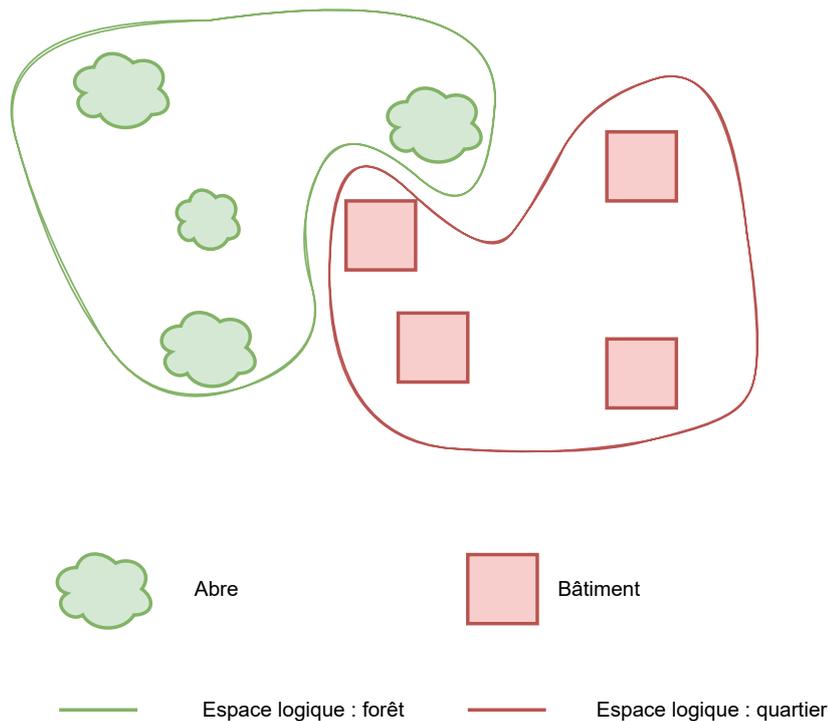


Figure 6.2 : Exemple de deux espaces logiques CityGML 3.0, définis selon le type des entités : arbre ou bâtiment

Un *espace physique* (figure 6.3) est un espace entièrement ou partiellement délimité par des objets physiques. Un bâtiment est un espace physique, car il est délimité par les murs qui le composent. Les objets physiques qui délimitent un espace sont appelés *frontières de l'espace* (*space boundary*). Un même objet physique appartenir à la frontière de plusieurs espaces physiques, elles réalisent ainsi le contact entre des espaces adjacents. Un même mur peut délimiter deux pièces distinctes. La surface de l'eau est la limite entre le plan d'eau et l'air, la surface de la route est la limite entre le sol et l'espace de circulation. Les objets physiques qui font partie des frontières d'un espace sont appelés *objets externes*.

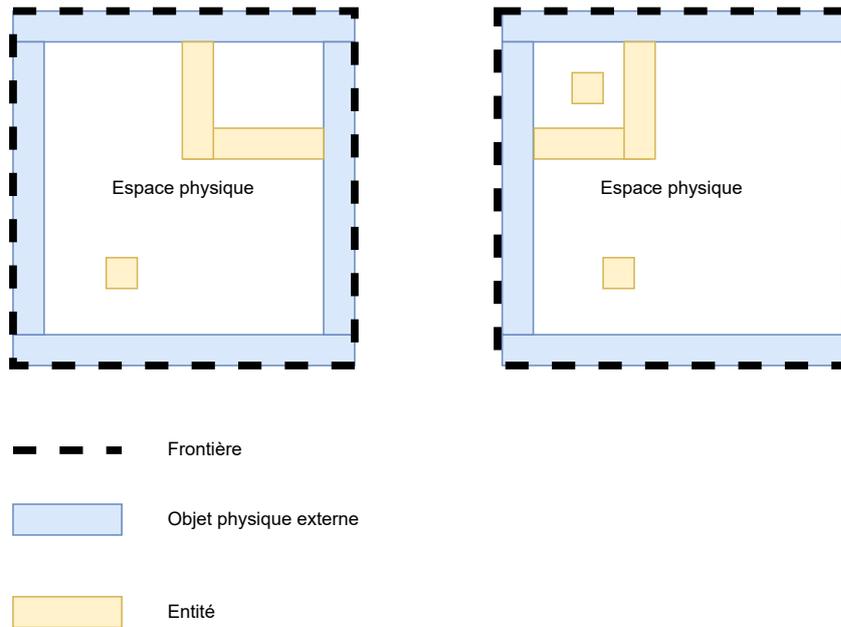


Figure 6.3 : Exemple de deux espaces physiques CityGML 3.0 : le premier est entièrement délimité par des objets physiques, le second est partiellement délimité par des objets physiques

Les espaces physiques sont séparés en deux concepts distincts : les espaces occupés et les espaces non occupés (6.4). La notion d'occupation est définie comme relative à l'occupation d'un espace dans l'environnement urbain. Les *espaces occupés* sont des espaces représentant des **objets physiques** volumétriques qui occupent l'espace dans l'environnement urbain, comme un bâtiment ou le mobilier urbain. Malgré la restriction à l'environnement urbain, un meuble est défini comme espace occupé, car il occupe de l'espace dans une salle, définie comme espace non occupé. Les *espaces non occupés* sont des espaces représentant des **entités** volumétriques qui n'occupent pas d'espace dans l'environnement urbain. Cette notion d'occupation est relative à la liberté de circulation dans cet espace, à priori en ne prenant seulement en considération les véhicules à l'échelle de la ville et l'humain à l'échelle de l'intérieur de bâtiment. Un espace non-occupé est considéré comme libre pour la circulation. Ainsi, bien que semblant occuper un espace dans l'environnement, une route est cependant définie par CityGML comme espace non occupé. Cette observation est expliquée et détaillée dans le paragraphe suivant.

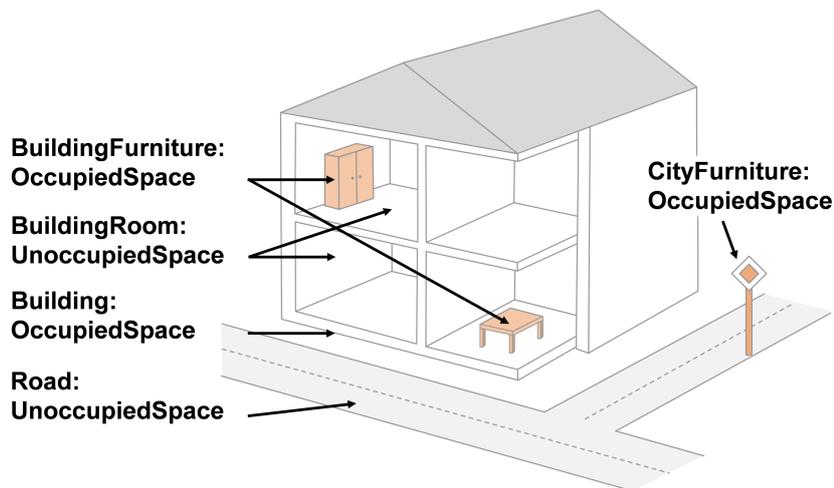


Figure 6.4 : Classification en espace occupé et non occupé d'entités urbaines selon CityGML 3.0 [OGC18a]

La classification d'espaces de circulations (TrafficSpace) en tant qu'espaces non occupés peut être ambiguë car ils sont constitués d'objets physiques (la chaussée ou les pavés d'un trottoir) qui occupent de l'espace. Les espaces de circulations (TrafficSpace), comme les routes, sont définis comme la composition de deux espaces (figure 6.5) : un espace libre de déplacement (ClearanceSpace) défini comme non occupé et une aire de circulation (TrafficArea) considérée comme occupée.

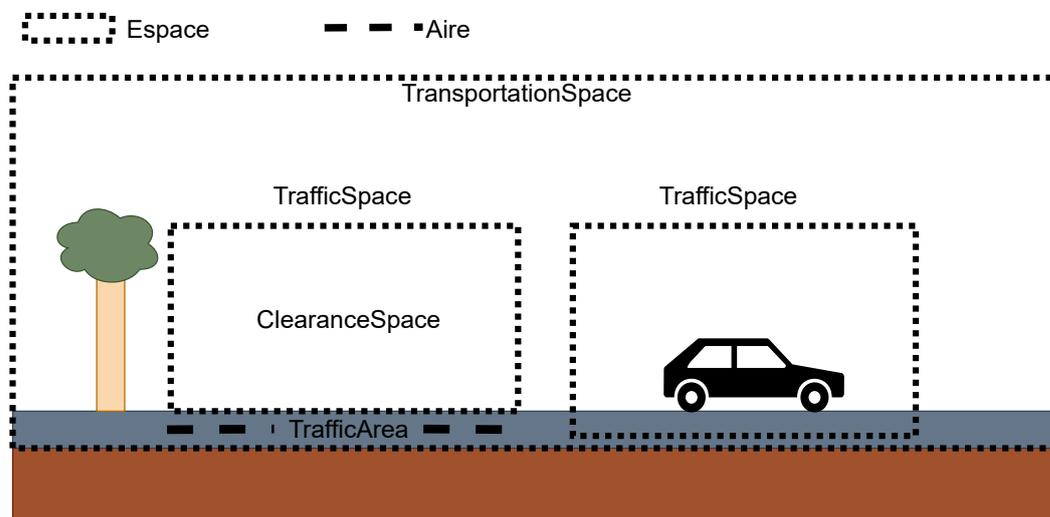


Figure 6.5 : Illustration des concepts CityGML 3.0 d'espaces de circulations (TrafficSpace), composé d'un espace de déplacement (ClearanceSpace) et d'une aire de circulation (TrafficArea)

Une route est définie comme un espace non occupé et comme une composition d'un espace non occupé (l'espace circulaire) et d'un espace occupé (la chaussée). Un bâtiment est défini comme espace occupé et comme une composition d'espaces non occupés (les pièces) et occupés (les murs). La notion d'espace occupé n'est pas stricte et ambiguë, fait observé par l'OGC : "There is a risk of misunderstanding the term OccupiedSpace"³. Elle est relative à l'*occupation d'un espace dans l'environnement urbain* qui n'est pas définie formellement par le modèle de données. Cette définition n'est donc pas interprétable nativement par la machine et rend difficile la réutilisation de ces concepts dans CityGML à différentes échelles (territoire entier, espace intérieur, objet technique, ...), ou dans d'autres modèles.

6.1.2 Modèle IFC

Le modèle IFC propose une formalisation d'entité qui diffère du modèle CityGML. La classe *IfcObject*⁴ généralise toute chose ou processus traité sémantiquement par le modèle : les espaces ou les objets physiques tangibles tels que les murs, les poutres ou les revêtements, mais aussi les processus tels que les tâches de travail, des contrôles ... Le sous-type *produit* (*IfcProduct*)⁵ est défini comme la représentation abstraite de tout entité liée à un contexte spatial qui se situe à un endroit spécifique dans l'espace. On peut alors décrire les espaces à l'aide d'*éléments spatiaux* (*IfcSpatialElement*)⁶ et les objets physiques à l'aide de la notion d'*élément* (*IfcElement*)⁷.

3. <https://docs.ogc.org/is/20-010/20-010.html>

4. https://standards.buildingsmart.org/IFC/DEV/IFC4_2/FINAL/HTML/schema/ifckernel/lexical/ifcobject.htm

5. https://standards.buildingsmart.org/IFC/DEV/IFC4_2/FINAL/HTML/schema/ifckernel/lexical/ifcproduct.htm

6. https://standards.buildingsmart.org/IFC/DEV/IFC4_2/FINAL/HTML/schema/ifcproductextension/lexical/ifcspatialelement.htm

7. https://standards.buildingsmart.org/IFC/DEV/IFC4_2/FINAL/HTML/schema/ifcproductextension/lexical/ifcelement.htm



Figure 6.6 : Héritages des types *IfcElement* et *IfcSpatialElement* selon le modèle IFC.

La notion d'espace est adressée à l'aide d'*éléments spatiaux* : une *zone spatiale* (*IfcSpatialZone*) ou un *élément de structure spatiale* (*IfcSpatialStructureElement*). Une *zone spatiale* est une décomposition non hiérarchique et potentiellement chevauchante répondant à des considérations fonctionnelles (zone thermique, électrique, etc.). Une *structure spatiale* est une décomposition hiérarchique d'une entité constituée d'*éléments* ou d'*éléments spatiaux*. Une *structure spatiale* pour un bâtiment est composée des *éléments de structure spatiale*⁸ suivants : un site, un bâtiment, un étage et une salle. La figure 6.7 présente un exemple d'une structure spatiale d'un bâtiment constitué de deux sections de deux et trois étages chacune. Pour un pont, on retrouve la structure spatiale suivante : un site, un pont et une partie de pont.

8. <https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD1/HTML/schema/ifcproductextension/lexical/ifcspatialstructureelement.htm>

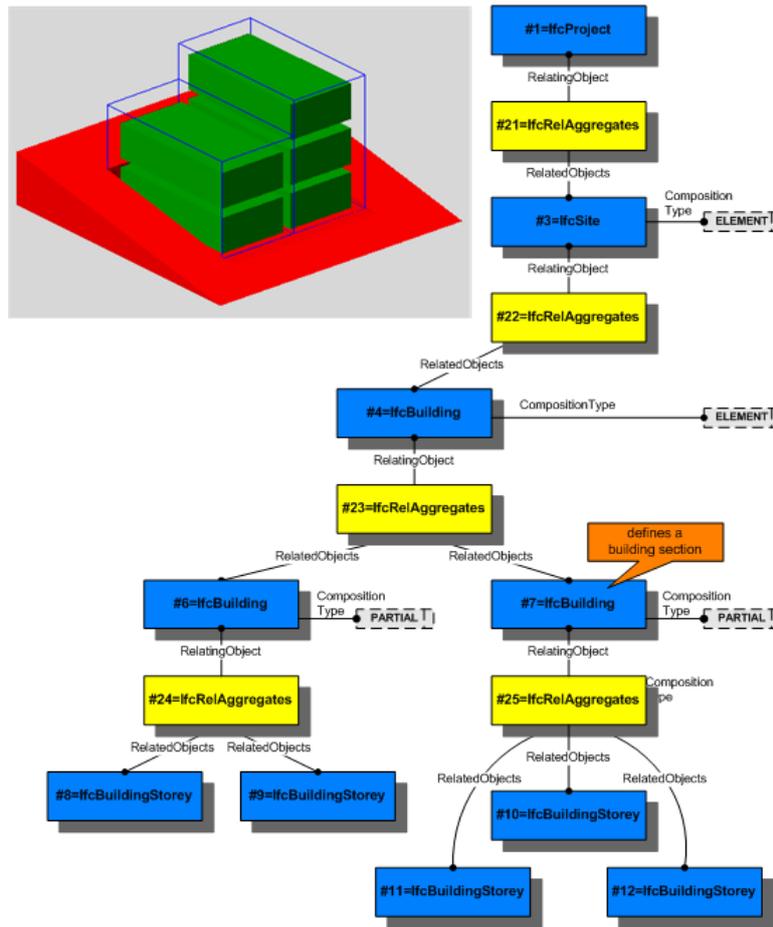


Figure 6.7 : Exemple d'une structure spatiale d'un bâtiment constitué de deux sections de deux et trois étages chacune, selon le modèle IFC

Les objets physiques tels que des murs, des poutres ou des revêtements sont conceptualisés sous la notion d'*élément*, décrit comme la généralisation de tous les composants qui constituent un produit issu des domaines de l'architecture, de l'ingénierie et de la construction (AEC). Ces *éléments* disposent de représentations géométriques décrites à l'aide de données géométriques. Chaque *élément* d'un site est associé à au moins un *élément spatial* de ce site situé à n'importe quel niveau de sa structure spatiale. Dans la figure 6.8, un bâtiment est divisé en deux sections de bâtiments distinctes. Un escalier est associé à la section de gauche. Cette section de bâtiment est une nouvelle fois divisée en deux étages, dont l'un contient deux murs.

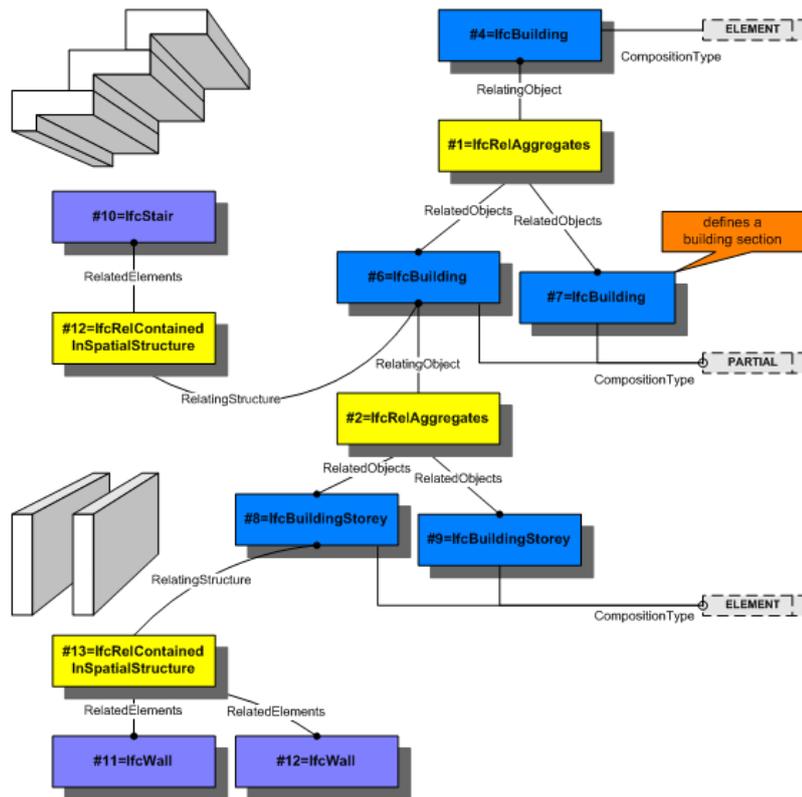


Figure 6.8 : Exemple d’une structure spatiale d’un bâtiment constituée de deux sections, d’un escalier et de deux murs, selon le modèle IFC

On retrouve la notion d’*élément externe* à un espace dans ce modèle via la notion d’*élément de structure spatiale externe* (IfcExternalSpatialElement⁹). Une propriété (IsExternal) permet d’indiquer si l’élément est externe au bâtiment entier. La relation IfcRelBoundedBy permet de relier un élément spatial à l’ensemble des éléments qui le délimite.

6.1.3 Alignement des modèles IFC et CityGML 3.0

En prenant en compte l’ensemble de ces observations, nous proposons une correspondance entre les concepts associés aux espaces et entités de CityGML et IFC dans la table 6.1.

9. <https://standards.buildingsmart.org/IFC/RELEASE/IFC4/ADD1/HTML/schema/ifcproductextension/lexical/ifcexternalspatialstructureelement.htm>

CityGML	IFC
Espace abstrait (Abstract space)	Produit (IfcProduct)
Espace logique (Logical space)	Zone spatiale (IfcSpatialZone)
Espace physique (Physical space)	Elément de structure spatiale (IfcSpatialStructureElement)
Espace occupé (Occupied space)	Elément (IfcElement)
Espace non occupé (Non-occupied space)	Elément d'ouverture / Zone spatiale (IfcOpeningElement / IfcSpatialZone)
Frontières d'un espace (Space boundary)	Elément de structure spatiale externe (IfcExternalSpatialElement)

Table 6.1 : Proposition d'alignement des concepts d'espaces et d'entités par CityGML et IFC

Une même entité peut ainsi être modélisée différemment : comme un espace composé d'espaces et d'entités pour IFC ou un objet physique pour CityGML. Dans le modèle IFC, un bâtiment est un élément de structure spatiale (IfcSpatialStructureElement), composé d'espaces et d'objets physique. Un bâtiment est décrit comme un espace qui peut être représenté et décrit seulement par les objets physiques qui le délimitent et le composent. Visualiser un bâtiment revient à visualiser l'ensemble des entités qui le compose : les murs, les fenêtres, ...

Dans le modèle CityGML, un bâtiment est considéré comme un espace occupé, donc un objet physique, soit un élément (IfcElement) dans le modèle IFC. Un bâtiment est décrit comme un objet qui dispose de sa propre description et de ses propres représentations. Cet exemple montre que selon les concepts définis par les modèles de données, une même entité peut être représentée différemment : soit par une représentation spatiale, soit par les entités qui composent cette entité.

Cette différence de formalisation peut s'expliquer par le point de vue adopté par ces deux modèles lors de leur définition. Le modèle IFC est défini du point de vue du bâtiment. La modélisation d'un bâtiment suit alors une approche "bottom-up" : l'ensemble des objets physiques constituant un bâtiment sont d'abord décrits, puis regroupés spatialement dans des espaces alors décrits. Un espace est ainsi défini par l'ensemble des éléments qui le délimite spatialement et des zones qui le décomposent.

Les espaces ainsi formés sont ensuite regroupés dans une structure spatiale formant le bâtiment. Une pièce est définie par la composition des murs qui l'entourent. Un étage est défini par la composition des pièces comprises dans cet espace. Le modèle CityGML 3.0 est défini du point de vue de la ville et suit une approche "top-down" : l'espace urbain est divisé en espaces physiques, puis en bâtiment, qui sont occupés par des objets physiques.

Cependant, on peut remarquer que dans les modèles CityGML et IFC, une entité est pensée et décrite en tant qu'espace physique. Les entités peuvent disposer d'une ou plusieurs représentations spatiales, plus ou moins détaillées géométriquement. Les entités peuvent disposer d'informations sur leur composition, à l'aide de relations hiérarchiques ou de compositions avec d'autres entités enfants, par exemple en décrivant l'ensemble des murs qui constituent un bâtiment. En utilisant ces relations pour identifier les entités enfants et leurs représentations spatiales, on dispose alors de nouveaux modes de représentations spatiales pour une entité :

- les représentations spatiales des entités enfants peuvent **compléter** la représentation spatiale du parent et être visualisées en même temps.
- les représentations spatiales des entités enfants peuvent **remplacer** celle du parent.

Pour une entité donnée, un variant peut apporter différentes modes de représentations spatiales en utilisant les relations présentes dans les données, illustrée dans la figure 6.9. Dans le premier cas, le variant apporte une représentation spatiale de l'entité, par exemple un cube pour un bâtiment, mais ne dispose pas de relations hiérarchiques ou de compositions. Dans le second cas, le variant n'apporte pas de représentation spatiale de l'entité, mais dispose de relations hiérarchiques ou de compositions avec des entités enfants qui, elles, disposent de représentations spatiales. L'ensemble de ces représentations spatiales forment une représentation spatiale de l'entité parente. Par exemple, en décrivant un bâtiment comme une composition de murs, visualiser les quatre géométries représentant ces murs offrent également une représentation du bâtiment. Dans le troisième et dernier cas, le variant apporte une représentation spatiale et dispose de relations hiérarchiques ou de compositions avec des entités enfants. Dans ce cas, les représentations des entités enfants peuvent **compléter** ou **remplacer** la représentation de l'entité parente pour apporter plus de précision. Par exemple, un prisme à base triangulaire représentant un toit peut être ajouté à un cube représentant le bâtiment pour apporter plus de détails, ou bien les géométries de murs et de toit peuvent remplacer le cube du bâtiment.

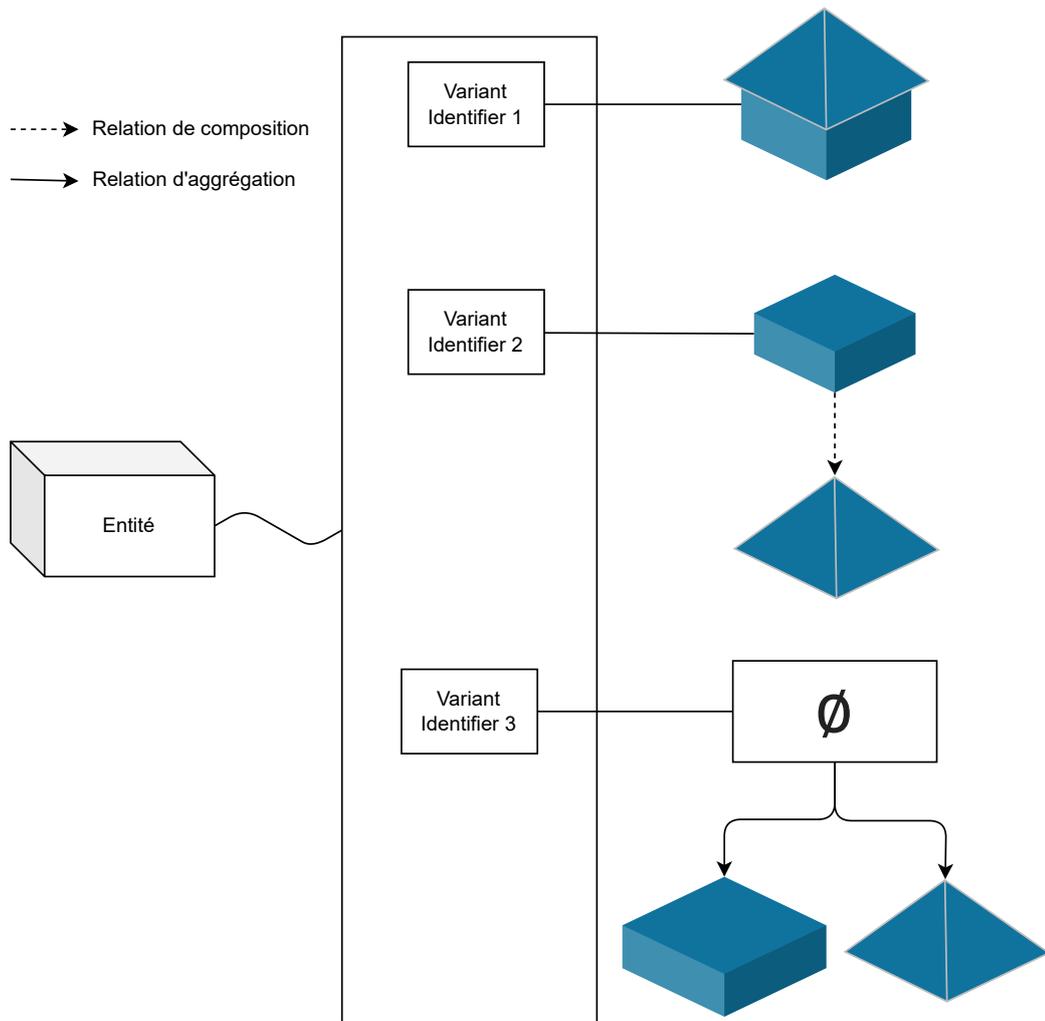


Figure 6.9 : Illustration des trois différents modes de représentations identifiés : une représentation spatiale directe, une représentation spatiale créée par composition et une représentation spatiale créée par agrégation.

Dans le domaine de l'informatique graphique, le niveau de détail d'un objet géométrique s'exprime par la complexité de la géométrie : le nombre de points, le nombre de faces, le poids de la texture associée ... Dans le cas des représentations spatiales issues de données géospatiales, l'apport d'information sur la composition d'une entité rend la gestion du niveau de détail plus complexe, car une représentation peut alors être une composition de représentations.

6.2 Visualisation multiscalaire d'entités et de leurs représentations spatiales

Nous présentons un ensemble de propositions afin de pouvoir décrire et trier les entités et leurs représentations spatiales en fonction de leur taille et complexité, en prenant en compte les concepts d'espaces et informations de relations présentés et généralisés à partir des modèles IFC et CityGML. L'objectif de ces propositions est de pouvoir qualifier et quantifier la complexité géométrique d'une représentation spatiale provenant de n'importe quelle source, afin de pouvoir proposer à l'utilisateur les représentations spatiales d'une entité pertinentes à visualiser en fonction de son contexte dans un environnement 3D.

Proposition 1 : Une entité peut être abstraite -dont les limites sont définies par des considérations thématiques- ou physique -dont les limites sont définies par des considérations physiques-. Elle peut être représentée par un espace (une étendue volumétrique). Cet espace permet de situer l'entité et d'apporter une représentation spatiale abstraite de l'entité.

Proposition 2 : Une entité peut être constituée d'autres entités, physiques ou abstraites, appelées entités enfants. Les entités enfants peuvent être annotées comme *externes*, ce qui signifie qu'elles font partie des entités physiques qui définissent la limite physique de leurs ancêtres. Une entité qui a au moins un enfant externe implique qu'elle peut être observée de l'extérieur en utilisant l'ensemble des enfants externes. Une même entité peut appartenir à la frontière de plusieurs entités distinctes. Par exemple, un même mur peut appartenir à la limite extérieure d'une salle ainsi qu'au bâtiment qui contient cette salle. La figure 6.10 illustre cette proposition. Une entité physique EP est décomposée en un ensemble d'entités abstraites et physiques. Cette entité est délimitée par les entités physiques EP1,2, 4 et 5, et contient également les entités abstraites EA1 et EA2 ainsi que l'entité physique EP3. L'entité abstraite EA1 est entièrement entourée par les entités physiques EP1, 2 et 5. L'entité physique EP5 appartient donc aux frontières de l'entité physique parente, mais aussi de l'entité abstraite EA1.

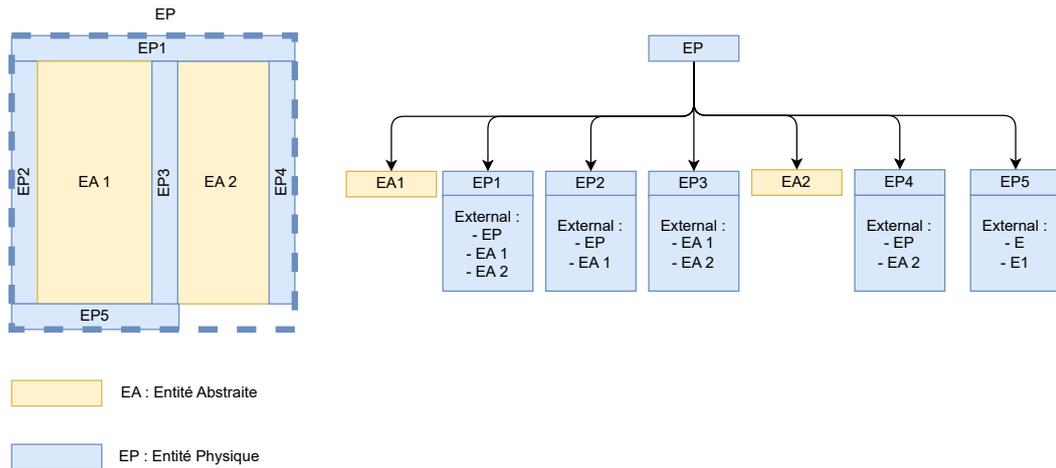


Figure 6.10 : Exemple de décomposition d’une entité physique en entités physiques et abstraites. Les entités enfants qui font parties des limites physiques d’une autre entité sont annotées comme *externe*.

Proposition 3 : On peut retrouver deux types de relations entre une entité et ses enfants, la composition ou l’agrégation. Dans le cas de la composition, les enfants s’ajoutent à la représentation du parent pour ajouter de l’information. Dans le cas de l’agrégation, les enfants remplacent entièrement le parent. Un même variant peut apporter différentes représentations spatiales : celles de l’entité, celles de l’entité avec les entités qui la compose, ou l’ensemble des entités qui forment le parent par agrégation. Dans le cas d’une agrégation, l’ensemble des entités enfants externes (si elles sont identifiées) d’une entité peut être considéré comme une représentation spatiale de cette entité vue de l’extérieur. Nous proposons d’appeler *représentation spatiale directe* une représentation spatiale d’une entité qui ne dispose pas d’informations sur les entités enfants d’une entité.

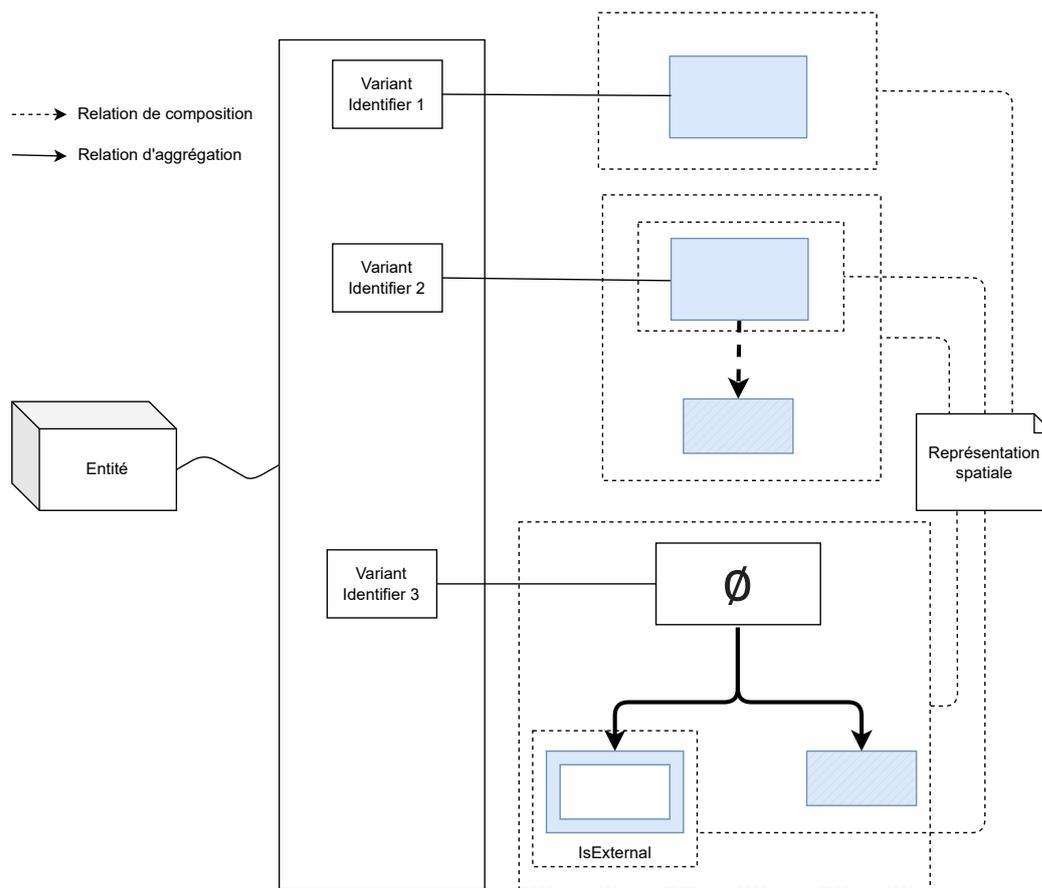


Figure 6.11 : Illustration des différentes représentations spatiales que peuvent apporter des variants issus de données géospatiales.

Proposition 4 : Nous proposons d'associer un niveau d'échelle à chaque entité, nommé Level Of Scale (LoS), et d'associer un niveau de complexité, nommé Level Of Complexity (LoC), aux représentations spatiales d'une entité. Le LoS permet de répondre à la question : quelles sont les entités à visualiser selon le contexte 3D de l'utilisateur ? Nous proposons d'adapter et d'utiliser les niveaux d'échelles que l'on retrouve dans les systèmes SIG classiques. Il s'agit d'une échelle, partant du niveau 0 qui correspond au monde entier. Plus le niveau est élevé et plus la zone observée est réduite. Pour chaque niveau, une aire visible est calculée à partir d'un nombre de pixel (usuellement 256) et un rapport de mètre par pixel à respecter. On retrouve quelques travaux permettant d'associer à chaque LoD CityGML et IFC un niveau d'échelle (figure 6.12), indiquant la représentation d'un bâtiment est requise en fonction de l'échelle. On peut donc associer à un bâtiment différents niveaux d'échelles pour lesquels le bâtiment doit être visible. Nous proposons d'associer des niveaux d'échelles aux entités en fonction de leur volume. Par exemple, un bâtiment

de 300m² serait alors associé au niveau 21. Au niveau 15, 300² représentent environ un carré de 3 pixels par 3. Il ne serait alors pas pertinent de l’visualiser en dessous. Au niveau 23, 300² représentent environ un carré de 1000 pixels par 1000. Il ne serait alors pas pertinent de le visualiser au-delà. L’utilisation des niveaux d’échelles SIG permet d’imiter et de profiter des mécanismes de visualisation existants des SIG. Par exemple, les changements dans la vue s’effectueront au même moment pour le fond de plan que pour les entités à visualiser.

Level	m/pixel	Width (m)	Area (m ²)	Examples of areas to represent	Scale	CityGML	Singapore BIM Guide v	3D Working Method	BIM LOD Konzept
15	4,777	1222	1495710	Small road	1:15 000	LoD1			
16	2,389	611	373927	Street	1:8 000	LoD2			
17	1,194	306	93482	Block, park	1:4 000	LoD2			LOD100
18	0,597	153	23370	Site (IfcSite), trees	1:2 000	LoD3			LOD100
19	0,299	76	5843	Local highway, crossing	1:1 000	LoD3	LOD100		LOD100
20	0,149	38	1461	Mid-sized building	1:500	LoD4	LOD100	Information Level 1	LOD200
21	0,075	19	365	Floor	1:250		LOD200 LOD300	Information Level 2	LOD200
22	0,037	10	91	Room	1:125		LOD300 LOD350 LOD400	Information Level 3	LOD200
23	0,019	5	23		1:60		LOD 400 LOD 500	Information Level 4 - 5	LOD300
24	0,009	2	6	Table	1:30			Information Level 4 - 5	

Figure 6.12 : Association des LoD (Level Of Detail) CityGML et LOD (Level Of Development) IFC à des niveaux d’échelles. L’ensemble du tableau est proposé dans la figure 2.17

Le LoC permet de répondre à la question : quelles sont les représentations d’une entité à visualiser selon le contexte 3D de l’utilisateur ? Plus une représentation spatiale est complexe, plus elle apporte d’information sur l’entité. Cette complexité sera utilisée comme un critère pour proposer à l’utilisateur des représentations spatiales pertinentes selon son contexte de visualisation. Cette complexité peut dépendre de la complexité géométrique de la représentation spatiale, mais aussi de la complexité des données sémantiques associées : les relations avec d’autres entités, le nombre d’entités utilisées pour décrire l’entité parente . . . Pour un même variant, une *représentation spatiale directe* devient plus complexe lorsque l’on y ajoute par composition les représentations de ces enfants. Par exemple, l’ajout de la représentation d’un toit sur une *représentation spatiale directe* rend la représentation du bâtiment plus complexe. Pour un même variant, la représentation formée par l’agrégation des entités enfants d’une entité est plus complexe que la représentation formée par l’agrégation des entités enfants externes, elle-même plus complexe qu’une *représentation spatiale directe* de l’entité. Par exemple, l’agrégation de représentations de quatre murs rend

la représentation du bâtiment plus complexe qu'une représentation directe de ce bâtiment.

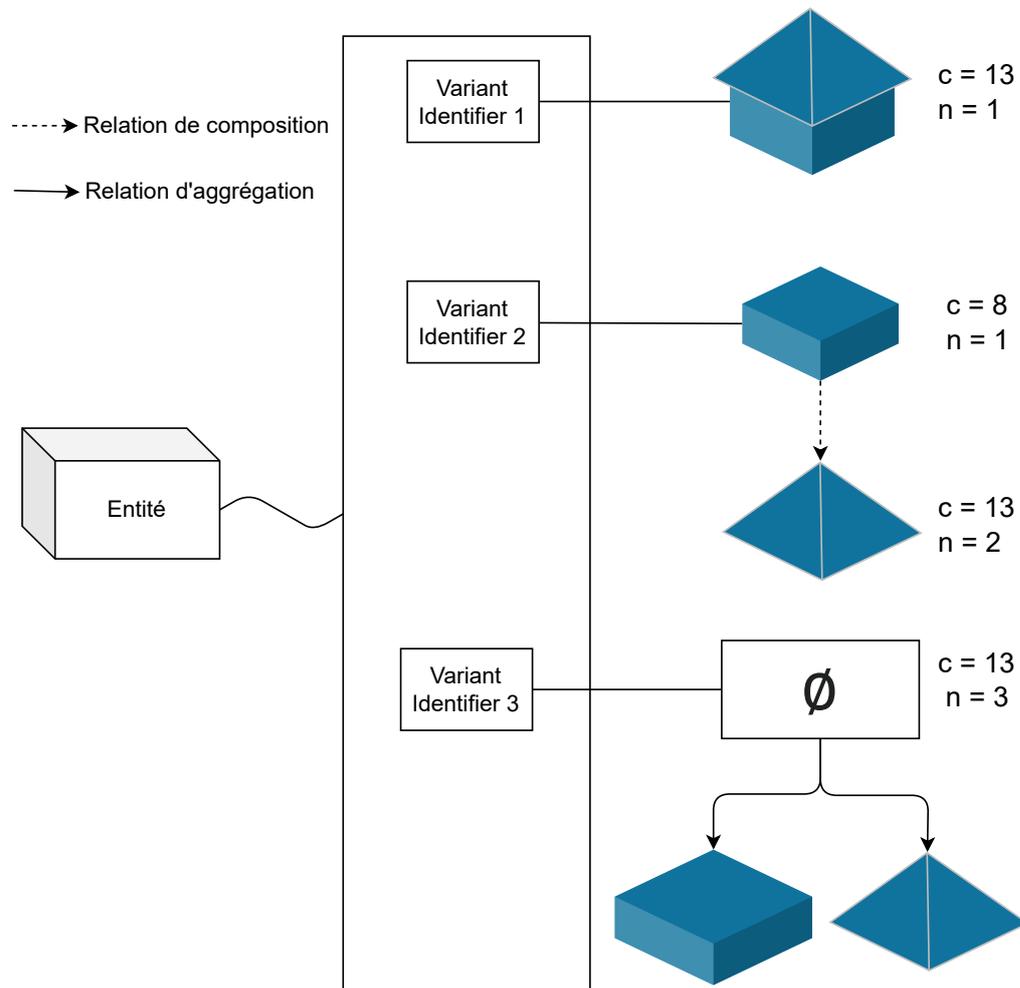


Figure 6.13 : Illustration d'une complexité de variants basée sur le nombre de points (ici, des sommets) (C) de la représentation et le nombre d'entités distinctes (N)

Ce LoC doit prendre en compte des critères géométriques, comme le nombre de points utilisés ou la résolution de la représentation, qui peut être calculée à partir de distance minimum, maximum ou moyenne entre deux points. La résolution peut être définie comme relative au LoS afin de fonctionner à n'importe quelle échelle et comprendre intuitivement le niveau de précision apportée par une représentation. Il est possible de prendre en compte d'autres informations, comme la dimension de la représentation, 2D ou 3D, ou encore la quantité d'information associée à la représentation, comme la granularité de celle-ci, c.-à-d., le nombre d'entités représentées.

Le calcul du LoC d'une représentation spatiale reste encore à approfondir. Nous proposons une expérimentation permettant de choisir différents critères permettant de le calculer afin d'observer l'impact de ces critères sur le résultat obtenu.

6.3 Expérimentation, premiers résultats et conclusion

6.3.1 Expérimentation

Afin de pouvoir calculer le LoC d'un variant, nous avons ajouté des métadonnées aux variants dans le modèle geovolume (figure 6.14). Dans cette expérimentation, nous avons calculé, à l'aide de Py3tilers, le nombre de sommets et les distances minimum et maximum entre deux sommets ainsi que le nombre d'entités distinctes présentes dans chaque variant. La bounding box de chaque 3DContainer correspond à l'espace occupé par une entité, et servira à calculer le LoS de l'entité.

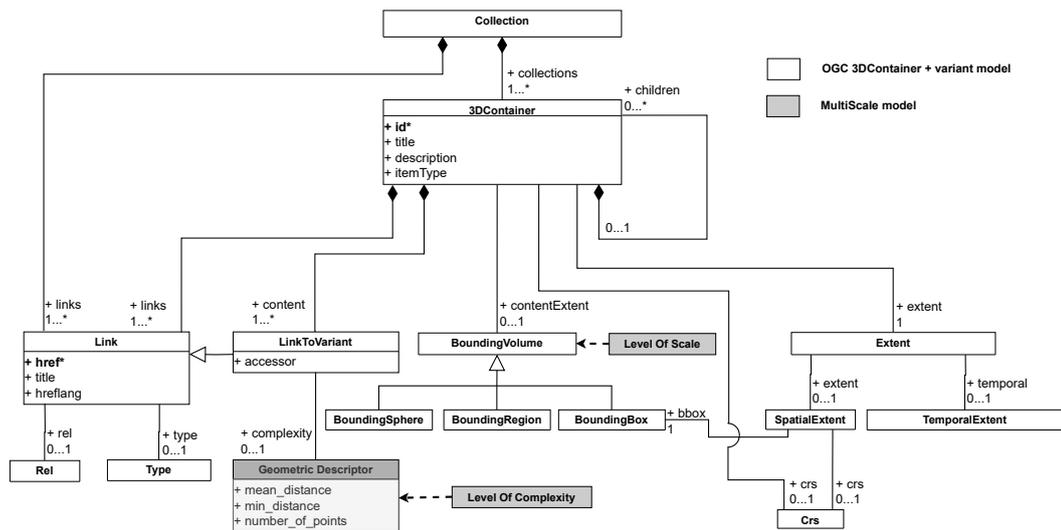


Figure 6.14 : Extension du modèle 3D-Container (en blanc) avec la classe Geometry Descriptor (en gris) qui permet d'associer des métadonnées aux variants correspondant à sa complexité.

Les informations liées aux relations de composition ou d'agrégation peuvent être récupérées depuis les données sémantiques en graphe. Dans le cadre de cette expérimentation, nous avons choisi d'utiliser une extension à 3DTiles pour conserver

ces informations dans les représentations spatiales extraites des données géospatiales¹⁰.

Cette expérimentation permet de choisir le critère géométrique à utiliser pour définir le LoC des représentations spatiales afin de les trier et sélectionner celles qui semblent le plus pertinentes. L'interface de cette expérimentation est illustrée dans la figure 6.15. Dans le cadre rouge A, on retrouve les informations associées à l'entité visualisée (le bâtiment gris) : ses variants disponibles, le LoS de l'entité calculé à partir de la bounding box B visualisée et les niveaux d'échelles minimum et maximum à partir desquels l'entité s'affiche ou non. Il est indiqué que la taille de l'entité correspond à un LoS de 19, et ne sera visualisée qu'entre les niveaux 14 et 21. À ces niveaux, l'entité occupera moins de 10 pixels, ou plus de 3840 pixels, soit deux fois la résolution de l'écran utilisé pour l'expérimentation (voir figure 6.12). Il est possible de choisir différents critères pour définir le LoC des représentations de cette entité : par distance minimum ou maximum entre deux vertex, le nombre de vertex ou d'entités décrites. Le LoC impact l'ordre d'affichage des représentations en fonction de la distance entre la caméra et l'entité : plus on est proche de l'entité, plus la représentation sera complexe.

On retrouve dans le cadre C le niveau d'échelle de la vue actuelle et l'échelle globale, avec pour chaque niveau un exemple de ce que l'on peut retrouver à cette échelle. Usuellement, les librairies de visualisation de données géospatiales mettent à disposition une méthode pour récupérer le rapport de mètres par pixel d'une vue, ce qui permet de calculer le niveau actuel de l'échelle.

10. https://github.com/CesiumGS/3d-tiles/tree/main/extensions/3DTILES_batch_table_hierarchy



Figure 6.15 : Interface de l'expérimentation mise en place pour comparer différents LoC. Dans le cadre A, on retrouve le LoS de l'entité, le LoS minimum et maximum et le critère utilisé pour calculer le LoC. La bounding box B représente l'espace qu'occupe l'entité. Le cadre C représente les différents niveaux d'échelles et le niveau d'échelle courant.

6.3.2 Premiers résultats

Cette section présente les premiers résultats obtenus avec l'implémentation du critère basé sur le nombre de vertex d'une représentation. Selon un point de vue et un niveau d'échelle associé, on choisit l'entité à visualiser, puis la représentation adéquate. Les représentations des entités de l'expérimentation sont sélectionnées puis visualisées de la plus complexe à la moins complexe. Pour l'instant, le choix a été fait de les visualiser les unes à la suite des autres en fonction du niveau d'échelle de la vue. Par exemple : pour une entité qui dispose de deux représentations et qui peut s'afficher à des niveaux d'échelles 5 à 10, en fonction du critère utilisé, l'une des deux représentations s'affichera à des niveaux entre 5 à 7, et l'autre à des niveaux entre 8 à 10. Dans la figure 6.16, l'entité représentée est la ville de Villeurbanne. Deux variants sont disponibles : un 3DTiles provenant d'un ensemble d'instances de bâtiments CityGML, l'autre est un nuage de points. La représentation en CityGML dispose de moins de vertex que la seconde, elle est donc affichée en premier et de plus loin. Lorsque le niveau d'échelle augmente, la représentation change pour afficher le nuage de points.



(a) Zoom 1



(b) Zoom 2

Figure 6.16 : Enchainement de deux représentations de Villeurbanne en fonction d'un LoC défini par le nombre de vertex dans la représentation

On retrouve ce même fonctionnement à un autre niveau d'échelle dans la figure 6.17. Dans cet exemple, le bâtiment de Carl Software est sélectionné. On dispose de trois représentations : une maquette numérique CityGML, la maquette IFC, et la représentation spatiale formée par l'ensemble des entités externes du bâtiment dans la maquette IFC. Elles sont également triées en fonction de leur nombre de vertex. En premier, la représentation CityGML est affichée. Au zoom, l'extérieur de la maquette IFC s'affiche, puis la maquette IFC.



(a) Zoom 1



(b) Zoom 2



(c) Zoom 3

Figure 6.17 : Enchainement de trois représentations du bâtiment de CarlSoftware en fonction d'un LoC défini par le nombre de vertex dans la représentation

Cette première expérimentation démontre qu'il est possible d'associer des niveaux de complexité à des entités, indépendamment de leur échelle, et qu'il est possible de naviguer entre elles pour apporter plus ou moins de détails en fonction du contexte 3D de l'utilisateur. D'autres critères pourront être pris en compte dans de futurs travaux et expérimentations afin d'identifier les plus pertinents.

Les représentations sont automatiquement alternées en fonction du niveau d'échelle. Il peut être pertinent de proposer à l'utilisateur celles qui semblent les plus pertinentes en fonction de ce niveau au lieu de n'en afficher qu'une seule à la fois pour lui laisser le contrôle sur sa vue.

6.3.3 Discussion

Dans ce chapitre, nous avons étudié les conceptualisations d'une entité définies par différents modèles de données géospatiales dans le but de comprendre comment les représentations spatiales d'entités peuvent être généralisées. Nous avons proposé une mise en correspondance entre les différents concepts d'espaces et d'entités décrits par les modèles IFC et CityGML. Grâce à cette étude, nous avons détaillés quatre propositions pour décrire génériquement une entité comme un espace, décrire les relations qu'il est possible de retrouver dans les représentations spatiales et une méthode pour choisir, selon un contexte 3D, les entités à afficher et les représentations à utiliser. Nous apportons une solution permettant de sélectionner automatiquement les entités à afficher selon le point de vue de l'utilisateur, quelle que soit l'échelle, basée sur l'espace occupé par une entité et le système d'échelle utilisé dans les SIG classiques. Les différentes représentations spatiales que l'on peut retrouver dans les données géospatiales apportées par des variants complexes ont été généralisées et décrites.

L'expérimentation proposée est une première base de travail pour définir plus précisément les critères géométriques et sémantiques à utiliser pour décrire la complexité (LoC) d'une représentation spatiale, qu'il sera nécessaire de poursuivre dans le futur. Nous proposons à l'utilisateur de pouvoir sélectionner le critère à utiliser pour définir le LoC des représentations. Ces représentations sont ensuite affichées de la moins complexe à la plus complexe en fonction de la distance entre la caméra et l'entité. La visualisation simultanée de plusieurs représentations spatiales présente des intérêts, par exemple pour pouvoir comparer différentes versions d'une entité dans le temps. Il est nécessaire de proposer des méthodes pour visualiser et comparer des représentations spatiales d'une même entité. Des critères et métriques pourraient être trouvés pour qualifier les représentations thématiques et proposer

des vues de graphes ou multimédias pour compléter la vue spatiale afin d'apporter de nouvelles informations complémentaires.

Nous nous sommes concentrés sur le contexte 3D de l'utilisateur pour l'aider à créer sa vue contextuelle d'une entité. En ajoutant d'autres informations sur les entités, comme leur nature, ou d'autres informations sur les variants, comme leur temporalité, il serait possible de proposer d'autres filtres à l'utilisateur, afin de prendre en compte son contexte métier ou ses problématiques temporelles. L'exploitation des graphes de connaissances pourrait aider à remonter ces informations.

Contributions techniques : applications, outils open source et reproductibilité

Ce chapitre présente les contributions techniques réalisées au sein de la plateforme *Urban Data Services and Visualization* (UD-SV) du Liris et de Carl Software, qui respectent des méthodologies de recherches similaires, prenant en compte les besoins de la communauté scientifique, des entreprises et des utilisateurs finaux. Plus particulièrement, plusieurs applications et documentations ont été réalisées afin d'illustrer, démontrer et reproduire les différents travaux présentés (sections 3.3, 4.2, 5.3, 6.3). Ces efforts pour assurer la reproductibilité et la réutilisabilité de ces travaux ont permis leur intégration dans la suite logicielle de Berger-Levrault, dans un nouvel outil industriel nommé BL-Viz.

Ce chapitre présente les contributions techniques suivantes, qui ont permis de démontrer et illustrer les contributions scientifiques :

- C6 : Applications de gestion et de navigation multiscalaire d'entités et de leurs représentations
- C7 : Reproductibilité et réutilisabilité
- C8 : Création d'un outil (BL-Viz) dans la suite logicielle de Berger-Levrault basé sur les contributions C1 et C5

Les travaux présentés dans ce chapitre ont été présentés au concours "Les challenges GeoDataDays" en 2022¹ et ont reçu le premier prix, ce qui a permis une publication dans la revue géographique nationale **Mappemonde** [Col+23]. Ces travaux ont été également publiés dans le journal international **Transaction In GIS** [Sam+23].

7.1 Méthodologie

Les travaux techniques réalisés dans le cadre de cette thèse font partie de la plateforme Urban Data Services and Visualization (UD-SV) du Liris, visant à promouvoir

1. <https://www.geodatadays.fr/page/GeoDataDays-2022-Les-Challenges-Geodata/113>

le partage de la recherche multidisciplinaire. Le développement de cette plateforme se concentre autour de besoins clés d'une plateforme pour accéder, gérer, naviguer et visualiser un environnement urbain nD dans le cadre d'une approche ouverte, interopérable et reproductible. Cette plateforme est fondée sur des solutions interopérables pour construire, visualiser et interagir avec des jumeaux numériques urbains en 3D sur le web.

La création de l'innovation au sein de Berger-Levrault s'appuie sur une méthodologie² semblable à celle de la plateforme UD-SV. La recherche effectuée tout au long des travaux de cette thèse a été régulièrement confrontée à l'entreprise et aux utilisateurs finaux des applications de maintenance afin d'identifier leurs besoins et de s'assurer de la pertinence et faisabilité des solutions proposées. Ces échanges ont été rendus possibles grâce au développement d'applications permettant de démontrer les travaux réalisés, et de les confronter à des situations réelles. Les travaux scientifiques menés s'appuient donc sur ces deux méthodologies (figure 7.1) pour construire une solution scientifique robuste, documentée, réutilisable et reproductible pour répondre aux besoins des communautés ciblées, mais aussi des utilisateurs finaux.

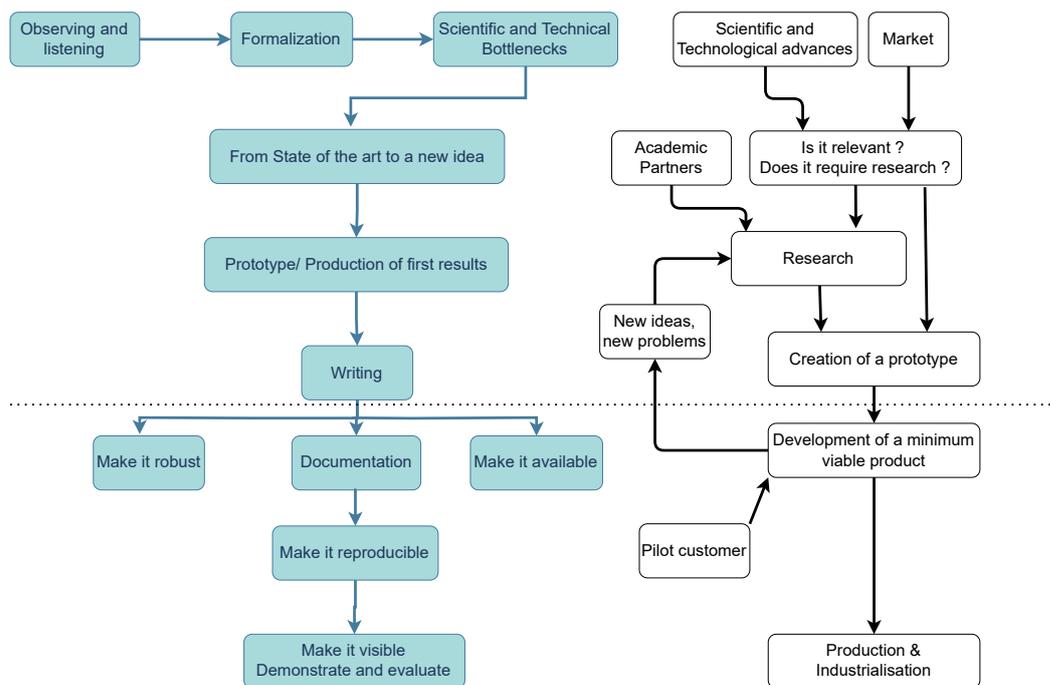


Figure 7.1 : Méthodologies appliquées dans la plateforme UD-SV (à gauche) et Berger-Levrault (à droite) pour construire une solution scientifique robuste, documentée, réutilisable et reproductible

2. <https://www.research-bl.com/about>

Cette méthodologie commence par l'observation et l'écoute des membres créateurs et utilisateurs de données géospatiales dans les domaines de la recherche et de l'industrie afin de mieux comprendre les technologies et les processus employés. En tenant compte de ces observations, nous concevons des spécifications fonctionnelles et techniques, des modèles de données et des traitements de données pour formaliser la manière dont un problème spécifique peut être décrit et résolu. Les problèmes ainsi formalisés doivent alors être analysés et étudiés afin de déterminer les défis scientifiques et techniques à relever. L'étude de l'état de l'art permet d'identifier les solutions existantes pour résoudre ces problèmes, celles qui doivent être modifiées ou améliorées, et celles qui n'existent pas. Dans le cadre de réalisation de travaux scientifiques au sein d'UD-SV, la recherche de solutions techniques est centrée autour de solutions open source, dans lesquelles il est possible de contribuer les éléments manquants si besoin.

Les premiers résultats obtenus sont présentés dans un prototype pour confronter nos formalismes à la réalité et s'assurer que nos solutions répondent aux besoins initiaux. La réalisation de ce prototype peut amener à créer de nouveaux outils techniques ou à contribuer à des outils existants. Cette étape permet de remettre en question nos travaux réalisés, nos formalismes et nos conceptions proposés.

Les résultats ainsi que le processus de création ou de développement de ces prototypes font partie intégrante des recherches effectuées. Cela signifie que le code ou l'application doit être robuste (évolutif et utilisable pour différents ensembles de données géospatiales), bien documenté et accessible au public. La documentation doit inclure les étapes pour construire l'application et reproduire les résultats publiés. La contribution envers la communauté scientifique s'effectue finalement à l'aide de la rédaction d'articles scientifiques, comprenant l'ensemble des solutions scientifiques et techniques proposées. Dans le cadre industriel de Berger-Levrault, une contribution se conclut idéalement par une industrialisation ou une mise en production de la solution scientifique proposée.

7.2 UD-SV : Plateforme de services et de visualisation de données urbaines pour le partage de la recherche multidisciplinaire

UD-SV est une plateforme basée sur des composants open source permettant d'intégrer, de visualiser, de naviguer, d'interagir et d'interroger des données géospatiales

n-dimension (2D, 3D, Temps, Thématique, ...) provenant de sources hétérogènes. La plateforme est construite à l'aide de normes et standards ouverts. Ces composants sont destinés à la recherche et à l'industrie dans le milieu des données géospatiales. Les utilisateurs peuvent interagir et naviguer dans un jumeau numérique urbain créé à l'aide des composants UD-SV et sont en mesure de visualiser la ville et ses objets de différentes manières. UD-SV intègre la dimension temporelle pour la gestion de l'évolution du paysage urbain [Vin23 ; PUG+23], et l'enrichissement des vues 3D à l'aide de documents multimédias [Jai20a ; GDG22b].

Les différents composants techniques utilisés dans la plateforme UD-SV sont présentés dans la figure 7.2 sous forme d'une architecture technique à 4 niveaux. Cette architecture à plusieurs niveaux permet de garantir la modularité et la création de composants dédiés au calcul, à l'analyse et à la gestion des données, au serveur et au client. UD-SV peut être utilisé, en tout ou en partie, sous la forme d'un ou plusieurs composants intégrés dans des plateformes internes préexistantes dans les organisations.

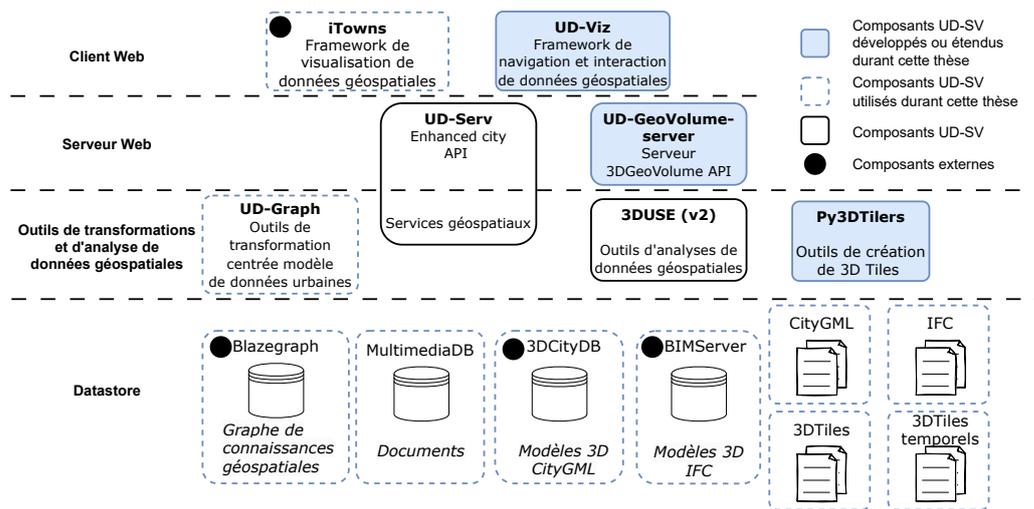


Figure 7.2 : Architecture technique à 4 niveaux de la plateforme UD-SV, adapté de [Sam+23]. Les composants développés ou étendus au cours durant les travaux de cette thèse sont surlignés en bleu. Les autres composants ou logiciels utilisés sont entourés en bleu.

Les quatre niveaux sont les niveaux : *client*, *serveur web*, *outils de traitements de données* et *données*. Le niveau *client*, composé d'UD-Viz³ et iTowns, est utilisé pour la visualisation des données urbaines sur le web à l'aide de navigateurs web. Le niveau

3. <https://github.com/VCityTeam/UD-Viz>

serveur web traite des demandes du client afin de sélectionner les informations pertinentes à visualiser. Nous avons développé un serveur de 3DContainer, servis à l'aide de l'API GeoVolume, pour pouvoir lister les entités et leurs représentations qu'il est possible de visualiser selon un point de vue donné. Le niveau *outils de traitements de données* effectue divers traitements sur les données géospatiales. Les travaux de cette thèse ont mené à la création de l'outil Py3DTilers pour créer des 3DTiles depuis différents formats de données géospatiales populaires. Le niveau *données* se compose de fichiers CityGML et IFC pouvant être hébergés dans des bases de données 3DCityDB et BimServer. On retrouve également des fichiers 3DTiles et 3DTiles temporels produits à l'aide du niveau *outils de traitements de données*.

Dans notre contexte d'étude, nous avons identifiés les outils open source existants iTowns⁴, Py3DTiles⁵, 3DCityDB⁶ et BimServer⁷ pour répondre à nos besoins de visualisation de données géospatiales dans un contexte web ; de transformation de données géospatiales vers un format pivot (ici, 3DTiles) ; et d'hébergement de données CityGML et IFC. Nos recherches ont permis d'étendre ces outils, notamment iTowns et Py3DTiles. iTowns est une librairie JS conçue pour aider les développeurs à intégrer des fonctionnalités de cartographie et de visualisation 3D dans leurs applications web. UD-Viz est une librairie JS basée sur iTowns qui permet d'expérimenter et d'explorer les possibilités offertes par iTowns. Nos travaux ont permis d'étendre UD-Viz pour créer différentes applications de visualisation de données géospatiales. Certaines fonctionnalités ont ensuite été reversées dans iTowns. Py3DTiles est originellement un outil permettant de transformer des nuages de points en 3DTiles. Nous avons développé l'outil Py3DTilers, basé sur Py3DTiles, pour créer des 3DTiles conformes au standard depuis d'autres formats de données. Ces améliorations sont en cours de reversement au sein de Py3DTiles.

Afin de respecter la méthodologie de la plateforme, les solutions techniques proposées dans celle-ci ont vocation à être reproductibles et réutilisables. Pour ce faire, chaque composant et application dispose d'une documentation présentant leur architecture et fonctionnement, leur installation et leur utilisation. Des technologies de virtualisation à l'aide de conteneur (tel que Docker⁸) ont été utilisées pour exécuter et déployer des programmes. Ainsi, chaque composant et application développés ont été dotés de conteneurs Docker pour faciliter leur installation et leur réutilisation. Ces conteneurs sont ensuite utilisés pour rendre nos démonstrations accessibles

4. <https://www.itowns-project.org/>

5. <https://gitlab.com/Py3DTiles/Py3DTiles>

6. <https://www.3dcitydb.org/3dcitydb/>

7. <https://bimserver.org/>

8. <https://www.docker.com>

par tous, en les déployant sur un serveur public⁹. Un système de lien permanent redirigeant sur les pages où sont déployées actuellement les démonstrations est d'ailleurs employé pour assurer la pérennité des liens partagés dans les articles scientifiques. Dans cet esprit, nous proposons des liens permanents (listés dans le tableau 7.1) vers les composants de la plateforme à l'aide de Software Heritage¹⁰, une organisation qui propose un service de conservation et de référencement de logiciel.

S.No.	Name	SWHID
1.	UD-SV	swh:1:dir:65963235ff00a37088b1b93283a896f4445c0cf4
2.	UD-Reproducibility	swh:1:dir:3aa1541cdf3f05bb1a1409f7afb314ebc19ee2cd
3.	3DUSE	swh:1:rev:0d643ee09460a7e3f594d5530eb0f0926286c6ab
4.	Py3DTiles	swh:1:rev:bf28341f32998ddd96387da90dcfb63fa6133ce6
5.	Py3DTilers	swh:1:dir:4f6dee808ce50aad4a448a8374f37c6e9c4407bf
6.	UD-Viz	swh:1:rev:b5a523825862a3f2dd440c2922485cdd30009834
7.	UD-Serv	swh:1:rev:c89553bf9098246f59ec68df2b12501f698aff77
8.	iTownS	swh:1:rev:e996369f1da742b8ca923f4c64d656f8d1a537db
9.	UD-Graph	swh:1:dir:7f43309db6dbee5bb56198445e69ddb698ecc682
10.	UD-GeoVolume-server	swh:1:dir:b47f1523dc6c5e3dffe98409ac96954c808d5e2d
11.	UD-Demo-GeoDataNavigation	swh:1:dir:6d19da6394e2d8900d8f89c0b5f80105054f35c3
12.	UD-Demo-GeoDataDays	swh:1:dir:8b08c8074217500435d833fddc8883bb4d610b82

Table 7.1 : Identifiants permanents Software Heritage (SWHIDs) des différents composants UD-SV et d'applications créées avec ces composants pendant les travaux de cette thèse (11 et 12).

Grâce aux différents composants développés dans cette plateforme, il est possible de composer le résultat des travaux scientifiques gravitant autour de cette plateforme dans une même application de jumeau numérique. Nous avons développé une démonstration de traitement, stockage et visualisation de données urbaines du quartier de la Doua à Lyon entre les années 2009 et 2021 : des données CityGML et SIG du quartier issues de l'Open Data du Grand Lyon, l'IFC d'un bâtiment, et des multimédias. Cette démonstration est basée sur les travaux scientifiques permettant de :

9. <https://projet.liris.cnrs.fr/vcity/demos>

10. <https://www.softwareheritage.org/>

- Extraire, transformer et stocker des géométries depuis des standards vers un format pivot spécialisé pour le web : 3DTiles (chapitre 3)
- Créer et stocker les données sémantiques et temporelles dans des graphes de connaissances et base de données non relationnelle [Vin23]
- Visualiser et organiser l'intégration de multimédias dans un espace 3D afin de guider l'utilisateur dans sa déambulation [GDG22a]

Cette démonstration a été développée dans le cadre de notre participation au challenge GeoDataDays 2022 que nous avons remporté, ce qui nous a permis de publier un article dans le journal Mappemonde [Col+23].

Un exemple du travail de reproductibilité requis peut être trouvé dans la démonstration créée pour illustrer les travaux présentés dans le chapitre 4. Dans cette démonstration, nous proposons de transformer des données CityGML et IFC en graphe et en 3DTiles, puis de les visualiser dans une application reposant sur UD-Viz. Afin de reproduire cette démonstration, nous mettons à disposition un répertoire GitHub¹¹ dans lequel on retrouve :

- Les données originales utilisées et les données produites après transformations
- Les instructions pour reproduire ces transformations
- Le code source de l'application de visualisation de données 3D et de graphes
- La documentation expliquant l'architecture et l'installation de l'application
- Des images Docker afin de lancer l'application localement

Pour effectuer cette démonstration, des fonctionnalités ont été ajoutées à UD-Viz pour avoir la capacité de requêter et traiter des GeoVolumes afin de naviguer entre les différentes représentations d'une même entité¹². Aussi, nous proposons un conteneur Docker public¹³ pour installer un store RDF Blazegraph sur Docker Hub, un registre public de conteneur Docker.

7.3 BL-Viz : Viewer 3D de jumeau numérique, du prototype à l'outil intégré aux solutions de GMAO

Dans le cadre de la recherche effectuée au sein de Berger-Levrault, nous avons proposé un prototype basé sur UD-Viz pour l'intégrer dans le logiciel CarlSource, le logiciel de GMAO développé par CarlSoftware | Berger-Levrault. Il a pour objectifs

-
11. <https://github.com/VCityTeam/UD-GeoDataNavigation>
 12. <https://github.com/VCityTeam/UD-GeoDataNavigation/tree/master/udviz-context/src/Extensions/GeoVolume>
 13. <https://hub.docker.com/r/vcity/blazegraph-cors>

d'illustrer les travaux réalisés, de démontrer la faisabilité technique du projet, mais aussi de confronter les résultats obtenus aux utilisateurs finaux. Pour réaliser ces objectifs, son intégration dans CarlSource est faible, le code est hébergé séparément du projet principal de CarlSource et l'instance du viewer est hébergée sur un serveur distant indépendant. Les contributions intégrées dans ce prototype sont celles du chapitre 3 permettant de transformer et homogénéiser des données 3D en utilisant un format pivot dédié aux données 3D, ici 3DTiles. Ce viewer est directement accessible depuis la page d'accueil du logiciel (figure 7.3). Il permet ainsi de visualiser le parc d'équipements à maintenir dans une carte.

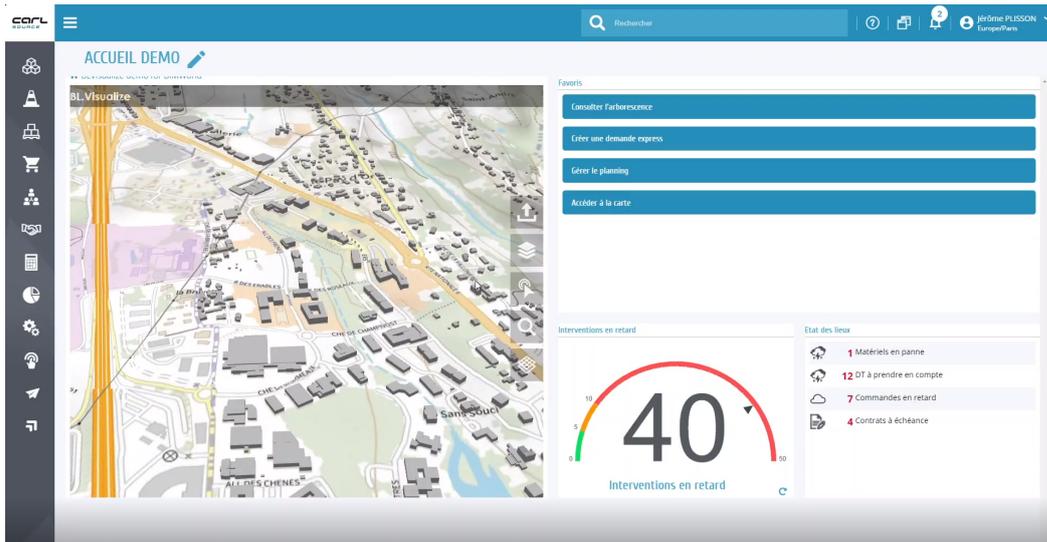


Figure 7.3 : Page d'accueil de CarlSource, avec le prototype intégré

Depuis cette vue (figure 7.4), il est possible de naviguer dans l'espace 3D, de filtrer les équipements à afficher, de changer de fond de plan, de rechercher un équipement à l'aide de son identifiant ou bien de sélectionner un équipement. Ces fonctionnalités ont d'ailleurs été contribuées dans UD-Viz, et réutilisées dans ce prototype. Une interface apparaît (en bas à gauche du viewer) pour afficher des informations sur un équipement sélectionné (identifiant, type, ...), mais aussi pour effectuer certaines opérations de maintenance : créer une intervention ou une demande d'intervention, déclarer un désordre, ou encore accéder à la fiche équipement GMAO de l'objet.

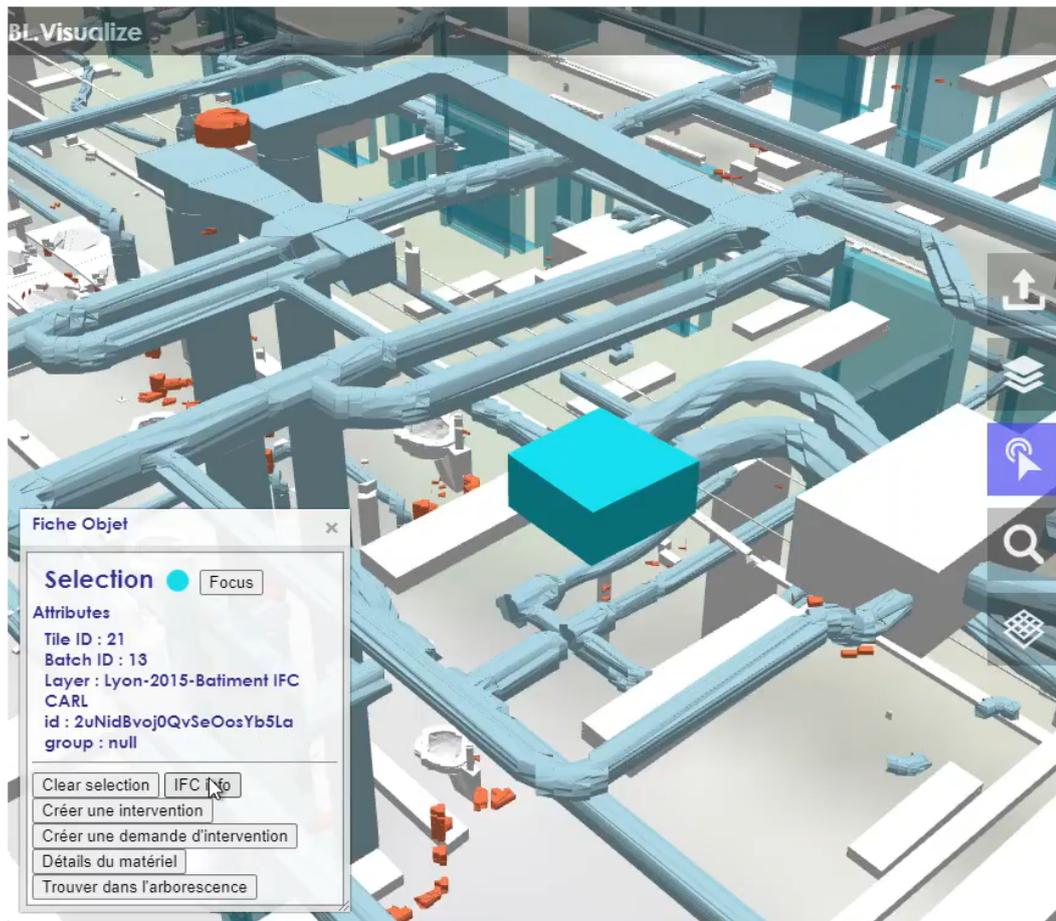


Figure 7.4 : Vue 3D d'équipements dans le prototype. Un équipement est sélectionné et affiché en bleu, ses informations sont affichés dans le cadre en bas à gauche, dans lequel il est possible d'effectuer des opérations de maintenance supplémentaires.

Le viewer peut également être affiché en même temps qu'une vue de l'arborescence du parc d'équipements, créée à partir des données IFC. Celle-ci permet de consulter les équipements existants par site, bâtiment, étage puis salle. Dans la figure 7.5, seules les cassettes de climatisation sont affichées dans la vue 3D à droite. L'une d'entre elles est colorée en bleu, car elle a été sélectionnée par l'utilisateur. Grâce au lien maintenu entre les représentations thématiques dans la base CarlSource et la représentation spatiale dans le viewer, la représentation thématique dans l'arborescence de cette entité peut alors être sélectionnée et mise en avant (en bleu à gauche). Il est également possible de sélectionner la représentation spatiale d'un équipement depuis cette arborescence.

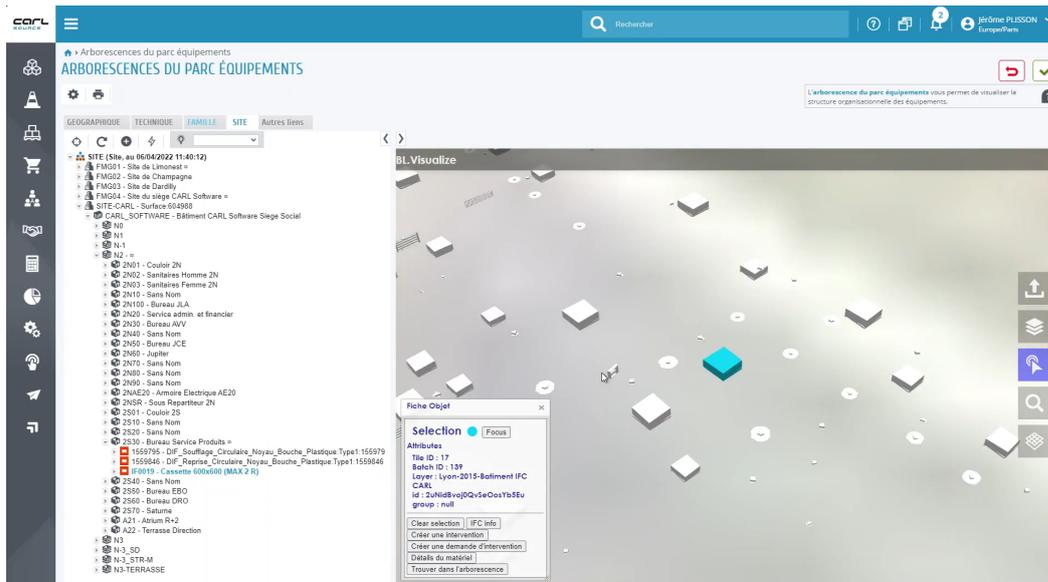


Figure 7.5 : Vue composée d'une vue 3D des équipements dans le prototype et d'une vue de l'arborescence des équipements. Un équipement est sélectionné et affiché en bleu, ses informations sont affichées dans le cadre en bas à gauche du prototype, sa représentation dans l'arborescence est colorée en bleu.

Grâce à la documentation et aux travaux de reproductibilité réalisés, ce prototype a pu être industrialisé dans un outil nommé BL-Viz au sein de la suite logicielle de Berger-Levrault, sous la forme d'un projet de développement constitué de trois équivalents temps pleins, un chef de projet et deux développeurs. L'architecture technique du viewer est présentée dans la figure 7.6. On retrouve l'outil Py3DTilers pour transformer les données géospatiales vers 3DTiles tout en conservant les identifiants des objets afin de garantir un lien avec des représentations thématiques issues des mêmes sources ou externes. L'outil IFCLoader permet d'intégrer les données sémantiques des maquettes IFC au sein de la base de données de maintenance. Enfin, le viewer, BL-Viz, est basé sur iTowns. Celui-ci est intégré fortement au sein de CarlSource : le code est hébergé et maintenu au sein de la même base de code, il peut se connecter directement aux bases de données clients et implémente la charte visuelle des logiciels de l'entreprise.

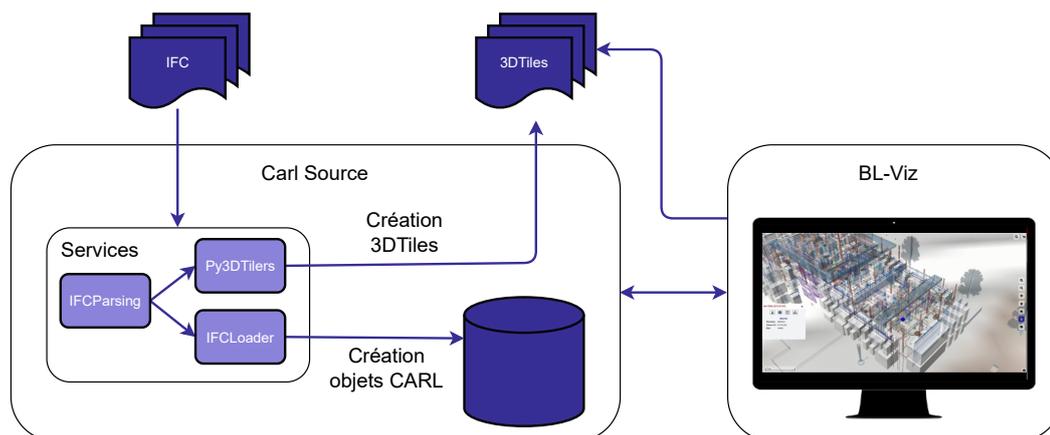


Figure 7.6 : Architecture technique de BL-Viz, outil intégré dans CarlSource

Cet outil permet d'intégrer des données géospatiales issues du BIM et du CIM afin d'obtenir de nouvelles représentations spatiales en plus des représentations thématiques des objets de la GMAO. Grâce aux liens créés entre ces représentations, il est possible d'effectuer des opérations de maintenance depuis les représentations spatiales (figure 7.7). Les responsables du patrimoine peuvent ainsi optimiser le pilotage de leur activité grâce à l'interaction des données de gestion et de maquettes 3D. Les techniciens bénéficient d'une aide aux diagnostics grâce à une meilleure compréhension de leur environnement (connaissance des chemins des réseaux, canalisations, accès virtuel aux espaces fermés et sécurisés...). Par rapport au prototype, on retrouve les fonctionnalités permettant de sélectionner un objet dans la scène 3D et des opérations de maintenance : déclarer un désordre, une demande d'intervention, visualiser la fiche équipement ou l'arborescence du site (cadre matériel à gauche). En plus des fonctionnalités présentes dans le prototype de sélection, de recherche d'un équipement par identifiant, de choix de fond de plan et de filtre des objets à afficher, il est possible via l'interface de zoomer ou dézoomer dans la vue, de créer des vues favorites, conservant la position et l'orientation de la caméra et de paramétrer le viewer.



Figure 7.7 : Visualisation d'un bâtiment IFC dans le viewer BL-Viz. Un objet (en bleu) est sélectionné et l'interface affichée permet d'effectuer des opérations de maintenance classique.

7.4 Discussion

Les travaux techniques menés durant cette thèse ont pris place au sein de la plateforme UD-SV, assurant une mise à disposition de la communauté scientifique un ensemble d'outils pour créer des applications d'analyse, de traitement et de visualisation de données géospatiales. Cette plateforme utilise des données et standards libres dans des solutions documentées et reproductibles, tout en essayant de réutiliser et contribuer à des composants logiciels open source. Plusieurs applications documentées et reproductibles de visualisation et navigation dans des données géospatiales ont donc été développées pour illustrer et démontrer les contributions scientifiques réalisées. Ces applications ont permis de développer de nouvelles fonctionnalités au sein d'UD-Viz, et de contribuer à la librairie iTowns. L'illustration de la méthodologie d'intégration de données géométriques à l'aide d'un format a nécessité le développement d'un outil, Py3DTilers, pour transformer des données géospatiales de sources hétérogènes vers 3DTiles, qui est en cours de reversement vers la librairie d'origine, Py3DTiles. La reproductibilité des applications est assurée par la mise à disposition de documentation du code et d'installation, des données sources utilisées et des données résultantes après transformation, ainsi que de conteneur Docker pour faciliter leur installation.

Le tableau suivant 7.2 présente l'état actuel du transfert de connaissance des contributions scientifiques et technologiques réalisées pendant les travaux cette thèse au sein des outils de l'entreprise. La contribution C1 d'intégration de données géométriques géospatiales vers un format pivot a mené à la réalisation d'un prototype faiblement intégré dans CarlSource, qui a permis de confronter les résultats obtenus aux utilisateurs finaux. Le transfert des connaissances scientifiques et technologiques a été entièrement réalisé, ce qui a permis d'industrialiser ces travaux, sous la forme d'un viewer nommé BL-Viz, basé sur iTowns, et de l'appropriation de l'outil Py3DTilers pour gérer la transformation des données géométriques en 3DTiles. La contribution C2 d'intégration de données géospatiales vers un format graphe est encore en cours d'étude auprès des utilisateurs finaux, notamment autour de l'exploitation des graphes à l'aide de requêtes prédéfinies, bien que le transfert des connaissances ait débuté. Les prototypes des contributions C3 et C4 permettant une navigation multiscalaire dans les variants d'une entité doivent encore être améliorés.

Contributions scientifiques	Prototype	Échanges : pertinence, problèmes et idées	Transfert des connaissances scientifiques et technologiques	Intégration dans les outils industriels
C1. Intégration de données géométriques vers un format pivot	✓	✓	✓	✓
C2. Intégration de données géospatiales vers un format graphe	✓	/	/	×
C3. Navigation dans les variants d'une entité	✓	/	×	×
C4. Navigation multiscalaire dans les variants d'une entité	/	×	×	×

Table 7.2 : Etapes des contributions scientifiques et technologiques proposées vers leurs industrialisations. ✓ : réalisé, / : en cours, × en attente

Conclusion : Contributions et perspectives

8.1 Contributions

Dans cette thèse, nous nous sommes intéressés à **l'intégration de données géospatiales pour permettre une navigation et une visualisation multiscalaire d'entités et de leur représentation spatiale et thématique dans un même contexte web**. L'objectif principal étant de pouvoir apporter un plus grand nombre d'informations à un utilisateur d'applications de jumeaux numériques urbains sur l'ensemble des entités décrites dans des sources de données hétérogènes. Pour nous permettre d'étudier ce problème, nous avons identifié puis répondu à trois questions de recherche à l'aide de contributions scientifiques et techniques.

Dans un premier temps, nous nous sommes intéressés à la **visualisation des données géométriques et sémantiques qui composent les données géospatiales dans un même contexte web**. Nous avons proposé une première méthodologie pour visualiser les données géométriques en les transformant vers un format pivot adapté pour ce type de données, suite à l'observation suivante : les modèles de données géospatiales ont été créés afin d'échanger et stocker des données sémantiques. Faisant face à un besoin technique permettant d'appliquer cette méthodologie, nous avons proposé un outil, Py3DTilers, ouvert à la communauté scientifique, pour transformer des données géospatiales de différents modèles (IFC, CityGML, .Obj, ...) vers le format 3DTiles. Dans cette méthodologie, nous permettons de conserver un lien vers les données sources afin d'accéder aux représentations thématiques associées aux représentations spatiales transformées, ainsi qu'une solution permettant d'accéder à des représentations thématiques qui ne disposant pas de représentations spatiales. Ces premiers travaux ont permis de visualiser dans un même contexte web les différentes représentations spatiales et thématiques apportées par des sources et modèles de données hétérogènes. Nous avons ensuite proposé une seconde méthodologie, qui permet une meilleure intégration des données sémantiques à l'aide d'une méthode d'intégration basée modèles. Celle-ci repose également sur le principe d'utilisation d'un format pivot : les modèles de données et les instances de données sémantiques sont transformés vers un format graphe homogène, une technologie issue du

domaine du web sémantique pensée pour la gestion et l'utilisation de ce type de donnée. Ces travaux ont été publiés dans l'annale du journal ISPRS Photogrammetry, Remote Sensing and Spatial Information Sciences [Col+22 ; Mar+22] et présentés lors de la 27^{ème} conférence internationale AGILE [Vin+24].

Dans un second temps, nous avons répondu au problème de **navigation entre différentes représentations d'une même entité issues de sources hétérogènes**. Nous avons proposé une définition plus précise et une formalisation plus complète du concept de *Variant* et proposé le concept de *Variant Identifier*. Le concept de *Variant* permet de décrire conceptuellement les représentations d'une même entité, et permet de les étudier. Le concept de *Variant Identifier* permet de naviguer dans les variants d'une entité. Il permet de référencer des variants existants dans tous types de sources et de les laisser perdurer si nécessaire dans leur contexte.

Dans un troisième et dernier temps, nous avons effectué des recherches pour permettre une **navigation multiscalaire dans des entités et leurs représentations spatiales**. Nous avons proposé une étude des formalisations et descriptions existantes d'une entité, basée sur les modèles CityGML et IFC, puis un alignement des différents concepts d'espaces proposés par ces modèles. Grâce à cette étude, nous avons émis des propositions pour naviguer dans un contexte 3D géospatial entre différentes entités en leur associant un niveau d'échelle (Level Of Scale), basé sur des travaux existants dans les SIG, et pour naviguer dans les représentations spatiales des entités visualisées, en leur associant un niveau de complexité (Level Of Complexity). Nous avons mis en place une expérimentation pour illustrer ces propositions et permettre l'étude des critères à prendre en compte pour calculer le LoC des représentations. Cette expérimentation nécessite des développements techniques supplémentaires afin d'obtenir des résultats intéressants sur l'utilisation de différents critères.

La réalisation de ces travaux scientifiques a pris place au sein de la plateforme UD-SV afin de construire des solutions scientifiques robustes, documentées, réutilisables et reproductibles pour répondre aux besoins des communautés ciblées, mais aussi des utilisateurs finaux. Cette plateforme regroupe différents outils libres développés dans le cadre de cette plateforme ou externe. Elle dispose de documentations et technologie de dockerisation associées à ces outils, mais aussi aux travaux scientifiques, afin de reproduire les résultats obtenus. Ainsi, chacune de ces contributions scientifiques a été illustrée par des applications reproductibles et réutilisables, qui ont mené à des contributions techniques au sein d'outils de la plateforme UD-SV. Ces efforts de reproductibilité et réutilisabilité ont permis la création d'un viewer basé

sur les premières contributions de cette thèse au sein de Berger-Levrault, nommé BL-Viz.

8.2 Perspectives

Dans notre contexte d'étude, les jumeaux numériques, il est possible d'identifier de nombreuses perspectives de recherches s'inscrivant dans la lignée des travaux réalisés pendant cette thèse. Les challenges liés aux jumeaux numériques reposaient jusqu'à aujourd'hui sur la capacité d'intégrer et de visualiser des représentations sémantiques et spatiales 2D et 3D dans un même contexte. Il est désormais nécessaire de trouver des solutions pour utiliser ces données afin de répondre à un besoin spécifique d'un utilisateur, et d'aller plus loin que la simple visualisation des données. L'apport d'un contexte 3D permet d'apporter de nouvelles informations à l'utilisateur, mais n'est pas forcément nécessaire. Grâce à nos travaux, nous disposons d'un continuum de données SIG-CIM-BIM, grâce auquel on peut visualiser différents variants d'une même entité, et donc différentes représentations spatiales et thématiques. Il serait alors possible de mobiliser ce continuum pour proposer des représentations pertinentes à l'utilisateur, en fonction de son besoin ou de l'échelle de la vue actuelle. Par exemple, l'utilisation de plan 2D pour représenter un itinéraire semble être la méthode la plus efficace à l'échelle de la ville. Cependant, il serait désormais possible d'aller plus loin en proposant des itinéraires allant jusqu'à l'intérieur des bâtiments grâce aux informations apportées par des maquettes BIM. De plus, il serait alors intéressant d'étudier la représentation d'un itinéraire en 3D dans les bâtiments, ceux-ci étant de nature verticale. Il faudrait alors avoir la capacité d'enchaîner des représentations 2D puis 3D afin de permettre à l'utilisateur de mieux comprendre le trajet à effectuer. De telles solutions permettraient d'intégrer les résultats de planification prédictive réalisés grâce à l'outil BL-Optim¹ dans BL-Vie et notre contexte d'étude. Le choix de la représentation d'une entité à l'aide d'une symbologie générique 3D peut également aider l'utilisateur à comprendre ce qu'il voit. Cela nécessiterait de s'intéresser à la création d'un standard pour créer de la symbologie 3D, en adaptant ceux existants dans les cartes 2D.

Les applications de jumeaux numériques ont pour objectifs d'agir comme plateforme de collaboration et de médiation entre les différentes parties prenantes des entités étudiées : par exemple, les citoyens, les municipalités, et commerçants à l'échelle urbaine, ou les plombiers, les constructeurs et les électriciens dans le contexte de construction d'un bâtiment. L'utilisation de maquette tangible, ou de réalité virtuelle

1. <https://www.research-bl.com/bl-optim/>

ou augmentée couplées aux données CIM-BIM peut permettre de nouvelles interactions et représentations avec cet environnement 3D complexe. Cette perspective est d'ailleurs en cours d'étude dans le cadre d'une thèse au sein du projet VCity du Liris² [And+22].

L'un des challenges émergeant aujourd'hui dans les applications de jumeaux numériques est la capacité de maintenir à jour les données afin de garantir que les données représentées correspondent au monde réel. L'une des solutions populaires est l'utilisation de capteurs et de données IoT. Cependant, l'intégration de données issues de capteurs avec des données géospatiales est un véritable challenge, notamment dû à la nature de ces données. Des solutions ressemblantes au dynamizer décrit par [CK16] et maintenant présent dans CityGML 3.0, qui permet de d'associer un champ de valeur à des données capteurs vivant à l'extérieur des données, semblent prometteuses. Les représentations de ces données dans un espace 3D à l'aide de tableau de bord, couleurs et / ou représentations 3D restent à étudier et définir. Ces recherches peuvent permettre de créer un lien fort entre BL-Predict³, une plateforme d'interopérabilité innovante dédiée à la gestion intelligente des équipements qui collecte, stocke, analyse et transforme des données IoT pour optimiser la performance des équipements, et BL-Viz.

La création régulière de données BIM et CIM par les acteurs du bâtiments et les villes, ainsi que l'intégration de données IoT avec ces données, posent la question de la gestion des versions ainsi créées des entités. Pour y répondre, [PUG+23] étudie au sein du projet VCity la modélisation du versionnement de données géospatiales transformées en graphe. Au delà de la modélisation de ces données, l'aspect temporel de ces données nécessitent des modes de représentations 3D innovants pour pouvoir comparer les différentes versions d'un même territoire et comprendre son évolution [GDG23]. Quelques perspectives possibles associées aux graphes de connaissances ont été présentées dans la section 4.1 à l'aide de la figure 4.2. Au-delà de l'exploitation d'un graphe de connaissance à l'aide de raisonneur, la représentation visuelle des données sous forme de graphe apporte elle aussi un grand intérêt. Par exemple, elle est particulièrement adaptée pour comprendre des réseaux d'eaux ou électriques. Grâce aux liens entre les entités géolocalisées et leurs données sous forme de graphes, visualiser ces graphes dans l'environnement 3D peut être une piste intéressante à étudier.

Au sein de Berger-Levrault, de nouvelles problématiques ont émergé à la suite d'échanges avec les utilisateurs finaux des travaux intégrés dans BL-Viz. Première-

2. <https://imu.universite-lyon.fr/atelier-maquette-semaine-de-l-anthropocene-2023--302584.kjsp>

3. <https://www.berger-levrault.com/fr/produit/bl-predict/>

ment, l'interface de navigation d'un utilisateur expert ou non dans un environnement 3D géospatial reste complexe, d'autant plus à l'échelle réduite du bâtiment. Il est nécessaire de trouver des solutions pour guider l'utilisateur dans cet espace 3D, comme lui proposer automatiquement des points de vue pour observer son objet d'intérêt, et de filtrer les entités à afficher selon son profil. La gamification des viewers géospatiaux 3D peut aider dans ce sens. Deuxièmement, il semble complexe d'obtenir des données qualitatives et complètement renseignées de la part des utilisateurs finaux. Par exemple, le modèle de données IFC permet de renseigner des relations entre des entités, comme la connection entre plusieurs tuyaux, qui forment alors un réseau d'eau distinct. En pratique, ces informations, bien que précieuses, sont rarement renseignées. Il serait donc intéressant de trouver des méthodes pour compléter les informations manquantes dans les données reçues. Dans le cas des réseaux, une solution consisterait à calculer des relations topologiques entre chaque tuyau à partir de leur géométrie 3D afin de retrouver les informations manquantes.

Bibliographie

- [Abo+99] Gregory D. ABOWD, Anind K. DEY, Peter J. BROWN et al. “Towards a Better Understanding of Context and Context-Awareness”. In : *Handheld and Ubiquitous Computing*. Sous la dir. d’Hans-W. GELLERSEN. Réd. par Gerhard GOOS, Juris HARTMANIS et Jan van LEEUWEN. T. 1707. Series Title : Lecture Notes in Computer Science. Berlin, Heidelberg : Springer Berlin Heidelberg, 1999, p. 304-307 (cf. p. 8, 36).
- [AB22] Jimmy ABUALDENIEN et André BORRMANN. “Levels of detail, development, definition, and information need : a critical literature review”. In : *Journal of Information Technology in Construction* 27 (19 avr. 2022), p. 363-392 (cf. p. 30, 31).
- [Alv+21] Diego Vinasco ALVAREZ, John SAMUEL, Sylvie SERVIGNE et Gilles GESQUIÈRE. “Towards Limiting Semantic Data Loss In 4D Urban Data Semantic Graph Generation”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* VIII-4/W2-2021 (2021), pp.37-44 (cf. p. 7).
- [And+22] Tiffany ANDRY, Julia BONACCORSI, Gilles GESQUIÈRE, Arnaud GRIGNARD et Thierry JOLIVEAU. “À quoi rêvent les maquettes? Maquette augmentée et médiation urbaine, un défi pluridisciplinaire”. In : *SHS Web of Conferences* 147 (2022). Sous la dir. de K. JACQUOT et H. LEQUAY, p. 02004 (cf. p. 124).
- [Avi91] D. E. AVISON. “MERISE : A European methodology for developing information systems”. In : *European Journal of Information Systems* 1.3 (août 1991), p. 183-191 (cf. p. 14, 15).
- [Bat18a] Michael BATTY. “Digital twins”. In : *Environment and Planning B : Urban Analytics and City Science* 45.5 (2018), pp.817-820 (cf. p. 3, 33).
- [Bat18b] Michael BATTY. “Digital twins”. en. In : *Environment and Planning B : Urban Analytics and City Science* 45.5 (sept. 2018). Publisher : SAGE Publications Ltd STM, p. 817-820 (cf. p. 13).
- [Bat21] Michael BATTY. “Multiple models”. In : *Environment and Planning B : Urban Analytics and City Science* 48.8 (2021), pp.2129-2132 (cf. p. 3, 33).
- [BBK20] F. BECK, A. BORRMANN et T. H. KOLBE. “The need for a differentiation between heterogeneous information integration approaches in the field of “bim-gis integration” : A literature review”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* VI-4/W1-2020 (2020), p. 21-28 (cf. p. 17, 18).

- [Bec+21] Stefan F. BECK, Jimmy ABUALDENIEN, Ihab H. HIJAZI, André BORRMANN et Thomas H. KOLBE. “Analyzing Contextual Linking of Heterogeneous Information Models from the Domains BIM and UIM”. en. In : *ISPRS International Journal of Geo-Information* 10.12 (déc. 2021), p. 807 (cf. p. 8, 63-65).
- [BBL10] Jakob BEETZ, Léon van BERLO et Ruben de LAAT. “Bimserver.org - an Open Source IFC model server”. en. In : (2010), p. 9 (cf. p. 18, 22).
- [Bei+21] C. BEIL, T. KUTZNER, B. SCHWAB et al. “Integration of 3d point clouds with semantic 3d city models – providing semantic information beyond classification”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* VIII-4/W2-2021 (7 oct. 2021), pp.105-112 (cf. p. 17).
- [BLS16] Filip BILJECKI, Hugo LEDOUX et Jantien STOTER. “An improved LOD specification for 3D building models”. In : *Computers, Environment and Urban Systems* 59 (sept. 2016), p. 25-37 (cf. p. 31).
- [Bil+14] Filip BILJECKI, Hugo LEDOUX, Jantien STOTER et Junqiao ZHAO. “Formalisation of the level of detail in 3D city modelling”. In : *Computers, Environment and Urban Systems* 48 (nov. 2014), p. 1-15 (cf. p. 29, 31).
- [Bil+15] Filip BILJECKI, Jantien STOTER, Hugo LEDOUX, Sisi ZLATANOVA et Arzu ÇÖLTEKIN. “Applications of 3D City Models : State of the Art Review”. en. In : *ISPRS International Journal of Geo-Information* 4.4 (déc. 2015). Number : 4 Publisher : Multidisciplinary Digital Publishing Institute, p. 2842-2889 (cf. p. 2, 13).
- [Bre+20] Martin BREUNIG, Patrick Erik BRADLEY, Markus JAHN et al. “Geospatial Data Management Research : Progress and Future Directions”. In : *ISPRS International Journal of Geo-Information* 9.2 (4 fév. 2020), pp.95 (cf. p. 33).
- [CH22] Nicholas J. CAR et Timo HOMBURG. “GeoSPARQL 1.1 : Motivations, Details and Applications of the Decadal Update to the Most Important Geospatial LOD Standard”. In : *ISPRS International Journal of Geo-Information* 11.2 (7 fév. 2022), p. 117 (cf. p. 23).
- [CK16] K. CHATURVEDI et T. H. KOLBE. “INTEGRATING DYNAMIC DATA AND SENSORS WITH SEMANTIC 3D CITY MODELS IN THE CONTEXT OF SMART CITIES”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-2/W1 (5 oct. 2016), p. 31-38 (cf. p. 26, 62, 124).
- [CK19] Kanishk CHATURVEDI et Thomas KOLBE. “A Requirement Analysis on Extending Semantic 3D City Models for Supporting Time-Dependent Properties”. In : *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences* IV-4/W9 (sept. 2019), p. 19-26 (cf. p. 13).
- [CLL16] Jing CHEN, Jiawei LI et Mo LI. “Progressive Visualization of Complex 3D Models Over the Internet : Progressive Visualization of Complex 3D Models”. In : *Transactions in GIS* 20.6 (déc. 2016), p. 887-902 (cf. p. 27).

- [Che+18] Yiqun CHEN, Erfan SHOORAJ, Abbas RAJABIFARD et Soheil SABRI. “From IFC to 3D Tiles : An Integrated Open-Source Solution for Visualising BIMs on Cesium”. en. In : *ISPRS International Journal of Geo-Information* 7.10 (sept. 2018), p. 393 (cf. p. 7, 21, 39).
- [CDD13] Jack CHENG, Yichuan DENG et Qianru DU. “MAPPING BETWEEN BIM MODELS AND 3D GIS CITY MODELS OF DIFFERENT LEVELS OF DETAIL”. In : (2013) (cf. p. 31).
- [Col+22] C. COLIN, J. SAMUEL, S. SERVIGNE, C. BORTOLASO et G. GESQUIÈRE. “Creating Contextual View of Cmm Assets Using Geospatial 2D/3D Data”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences X-4/W2-2022* (2022), p. 37-44 (cf. p. xix, 35, 37, 44, 47, 48, 122).
- [Col+21] Clément COLIN, John Samuel SAMUEL, Sylvie SERVIGNE, Y MILOUDI et Gilles GESQUIÈRE. “Visualisation d’équipements de la maintenance à l’aide de données géospatiales”. In : *Les journées Françaises de l’Informatique Graphique (JFIG) 2021*. Sophia-antipolis, France, nov. 2021 (cf. p. xix, 35).
- [Col+23] Clément COLIN, Corentin GAUTIER, Diego VINASCO-ALVAREZ et al. “UD-SV : Plateforme d’exploration de données urbaines à n-dimensions — Espace, Temps, Thématiques”. fr. In : *Mappemonde. Revue trimestrielle sur l’image géographique et les formes du territoire* 135 (mars 2023). Number : 135 Publisher : UMR Espace (cf. p. xix, 5, 11, 107, 113).
- [Col+20] Elisabetta COLUCCI, Valeria DE RUVO, Andrea LINGUA, Francesca MATRONE et Gloria RIZZO. “HBIM-GIS Integration : From IFC to CityGML Standard for Damaged Cultural Heritage in a Multiscale 3D GIS”. en. In : *Applied Sciences* 10.4 (jan. 2020). Number : 4 Publisher : Multidisciplinary Digital Publishing Institute, p. 1356 (cf. p. 18).
- [DCA16] Yichuan DENG, Jack C.P. CHENG et Chimay ANUMBA. “Mapping between BIM and 3D GIS in different levels of detail using schema mediation and instance comparison”. en. In : *Automation in Construction* 67 (juill. 2016), p. 1-21 (cf. p. 18).
- [DMJ23] Adeline DEPRÊTRE, Alexandre MIELNICZEK et Florence JACQUINOD. “Le City Information Modelling (CIM) au service d’un projet urbain : retour d’expérience sur la première phase de mise en œuvre du CIM d’un quartier :” in : *Flux* N° 133.3 (2023), p. 57-75 (cf. p. 1).
- [Don13] Sjors DONKERS. “Automatic generation of CityGML LoD3 building models from IFC models”. en. In : (2013) (cf. p. 18).
- [Don+16] Sjors DONKERS, Hugo LEDOUX, Junqiao ZHAO et Jantien STOTER. “Automatic conversion of IFC datasets to geometrically and semantically correct CityGML LOD3 buildings”. en. In : *Transactions in GIS* 20.4 (2016). _eprint : <https://onlinelibrary.wiley.com/doi/pdf/10.1111/tgis.12162>, p. 547-569 (cf. p. 16, 31).

- [FR14] S. FAI et J. RAFEIRO. “Establishing an Appropriate Level of Detail (LoD) for a Building Information Model (BIM) – West Block, Parliament Hill, Ottawa, Canada”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences II-5* (28 mai 2014), p. 123-130 (cf. p. 31).
- [FM09] Hongchao FAN et Liqiu MENG. “Automatic Derivation of Different Levels of Detail for 3D Buildings Modeled by CityGML”. In : *Proceedings of 24th International Cartography Conference (ICC2009)* (1^{er} jan. 2009) (cf. p. 31).
- [Fos+15a] Richelle FOSU, Kamal SUPRABHAS, Zenith RATHORE et Clark CORY. “Integration of Building Information Modeling (BIM) and Geographic Information Systems (GIS)—a literature review and future needs”. In : (2015), pp.27-29 (cf. p. 18).
- [Fos+15b] Richelle FOSU, Kamal SUPRABHAS, Zenith RATHORE et Clark A. CORY. “Integration of Building Information Modeling (BIM) and Geographic Information Systems (GIS) – a literature review and future needs”. In : 2015 (cf. p. 17).
- [GPG18] Jérémy GAILLARD, Adrien PEYTAVIE et Gilles GESQUIÈRE. “Visualisation and personalisation of multi-representations city models”. In : *International Journal of Digital Earth* (déc. 2018), p. 1-18 (cf. p. 22, 31, 40).
- [Gai+15] Jérémy GAILLARD, Alexandre VIENNE, Rémi BAUME et al. “Urban data visualisation in a web browser”. In : *Proceedings of the 20th International Conference on 3D Web Technology. Web3D '15*. New York, NY, USA : Association for Computing Machinery, juin 2015, pp.81-88 (cf. p. 7).
- [GDG22a] C. GAUTIER, J. DELANOY et G. GESQUIÈRE. “Integrating Multimedia Documents in 3D City Models for a Better Understanding of Territories”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences X-4/W2-2022* (2022), p. 69-76 (cf. p. 113).
- [GDG22b] C. GAUTIER, J. DELANOY et Gilles GESQUIÈRE. “INTEGRATING MULTIMEDIA DOCUMENTS IN 3D CITY MODELS FOR A BETTER UNDERSTANDING OF TERRITORIES”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences X-4/W2-2022* (2022), p. 69-76 (cf. p. 110).
- [GDG23] C. GAUTIER, J. DELANOY et Gilles GESQUIÈRE. “Representation of urban geometry evolution through space-time cube”. In : *27 International Conference Information Visualisation. GraphicsLink*. Tampere (Finlande), France : IEEE, juill. 2023, p. 414-419 (cf. p. 124).
- [Gri15] Michael GRIEVES. “Digital Twin : Manufacturing Excellence through Virtual Factory Replication”. In : (2015) (cf. p. 1).
- [GP12] Gerhard GRÖGER et Lutz PLÜMER. “CityGML - Interoperable semantic 3D city models”. en. In : *ISPRS Journal of Photogrammetry and Remote Sensing* 71 (juill. 2012), p. 12-33 (cf. p. 15).
- [GOS09] Nicola GUARINO, Daniel OBERLE et Steffen STAAB. “What Is an Ontology?” en. In : *Handbook on Ontologies*. Sous la dir. de Steffen STAAB et Rudi STUDER. International Handbooks on Information Systems. Berlin, Heidelberg : Springer, 2009, p. 1-17 (cf. p. 14).

- [HRB20] Elio HBEICH, Ana ROXIN et Nicolas BUS. “Previous BIM-GIS Integration Approaches : Analytic Review and Discussion”. In : oct. 2020 (cf. p. 16, 17).
- [Hij+20] I. H. HIJAZI, T. KRAUTH, A. DONAUBAUER et T. KOLBE. “3DCITYDB4BIM : a System Architecture for Linking Bim Server and 3d Citydb for Bim-Gis”. en. In : *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences V-4-2020* (août 2020), p. 195-202 (cf. p. 7, 22, 26).
- [Hij+09] Ihab HIJAZI, Manfred EHLERS, Sisi ZLATANOVA et Umit ISIKDAG. “IFC to CityGML Transformation Framework for Geo- Analysis : A Water Utility Network Case”. In : (2009) (cf. p. 18).
- [HJS16] A-H. HOR, A. JADIDI et G. SOHN. “BIM-GIS Integrated geospatial information model using semantic web and RDF graphs”. en. In : *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences III-4* (juin 2016), pp.73-79 (cf. p. 7, 22).
- [HS21] A.-H. HOR et G. SOHN. “Design and Evaluation of a BIM-GIS Integrated information Model Using RDF Graph Database”. English. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences. T. VIII-4-W2-2021*. Copernicus GmbH, oct. 2021, p. 175-182 (cf. p. 7).
- [Hua+20] W. HUANG, P.-O. OLSSON, J. KANTERS et L. HARRIE. “Reconciling city models with BIM in knowledge graphs : A feasibility study of data integration for solar energy simulation”. en. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences VI-4/W1-2020* (sept. 2020), pp.93-99 (cf. p. 22).
- [ISO04] ISO. *ISO 10303-11 Industrial automation systems and integration : The EXPRESS language reference manual*. <https://www.iso.org/standard/38047.html>. [Online ; accessed 23-May-2023]. 2004 (cf. p. 15).
- [ISO14a] ISO. *ISO19101 Geographic information — Reference Model - Part 1 : Fundamentals*. <https://www.iso.org/standard/59164.html>. [Online ; accessed 23-May-2023]. 2014 (cf. p. 14).
- [ISO15a] ISO. *ISO 19150-2 Geographic information : Ontology Part 2 : Rules for developing ontologies in the Web Ontology Language (OWL)*. <https://www.iso.org/standard/59193.html>. [Online ; accessed 23-May-2023]. 2015 (cf. p. 23).
- [ISO15b] ISO. *ISO19109 Geographic information : Rules for application schema*. <https://www.iso.org/standard/59193.html>. [Online ; accessed 23-May-2023]. 2015 (cf. p. 14, 68).
- [ISO16] ISO. *ISO19119 Geographic information — Services*. <https://www.iso.org/standard/59221.html>. [Online ; accessed 23-May-2023]. 2016 (cf. p. 1, 82).
- [ISO18a] ISO. *ISO19650 Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM)*. <https://www.iso.org/standard/68078.html>. [Online ; accessed 23-May-2023]. 2018 (cf. p. 2).

- [ISO18b] ISO. *ISO70303 Industry Foundation Classes (IFC) for data sharing in the construction and facility management industries*. <https://www.iso.org/standard/70303.html>. [Online; accessed 23-May-2023]. 2018 (cf. p. 16).
- [ISO19a] ISO. *ISO19107 Geographic information : Spatial schema*. <https://www.iso.org/standard/66175.html>. [Online; accessed 23-May-2023]. 2019 (cf. p. 14, 82).
- [ISO19b] ISO. *ISO19108 Geographic information : Temporal schema*. <https://www.iso.org/standard/26013.html>. [Online; accessed 23-May-2023]. 2019 (cf. p. 14).
- [ISO14b] ISO/TC 211. *ISO 19101-1 :2014 Geographic information — Reference model — Part 1 : Fundamentals*. en. 2014 (cf. p. 14).
- [Jai20a] Vincent JAILLOT. “3D, temporal and documented cities : formalization, visualization and navigation”. Theses. Université de Lyon, sept. 2020 (cf. p. 7, 110).
- [Jai20b] Vincent JAILLOT. “3D, temporal and documented cities : formalization, visualization and navigation”. Theses. Université de Lyon, sept. 2020 (cf. p. 41).
- [Jai+21] Vincent JAILLOT, Valentin RIGOLLE, Sylvie SERVIGNE, John Samuel SAMUEL et Gilles GESQUIÈRE. “Integrating multimedia documents and time-evolving 3D city models for web visualization and navigation”. In : *Transactions in GIS* 25.3 (mars 2021), p. 1419-1438 (cf. p. 4, 22).
- [JSG20] Vincent JAILLOT, Sylvie SERVIGNE et Gilles GESQUIÈRE. “Delivering time-evolving 3D city models for web visualization”. en. In : *International Journal of Geographical Information Science* 34.10 (oct. 2020), p. 2030-2052 (cf. p. 13, 19, 21).
- [JR18] P. JAYARAJ et A. M. RAMIYA. “3D CITYGML BUILDING MODELLING FROM LIDAR POINT CLOUD DATA”. In : *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-5 (2018), p. 175-180 (cf. p. 17).
- [Jon+20] David JONES, Chris SNIDER, Aydin NASSEHI, Jason YON et Ben HICKS. “Characterising the Digital Twin : A systematic literature review”. en. In : *CIRP Journal of Manufacturing Science and Technology* 29 (mai 2020), p. 36-52 (cf. p. 1).
- [Jul+18] Arttu JULIN, Kaisa JAALAMA, Juho-Pekka VIRTANEN et al. “Characterizing 3D City Modeling Projects : Towards a Harmonized Interoperable System”. en. In : *ISPRS International Journal of Geo-Information* 7.2 (fév. 2018). Number : 2 Publisher : Multidisciplinary Digital Publishing Institute, p. 55 (cf. p. 13).
- [KIH16] Ebrahim P. KARAN, Javier IRIZARRY et John HAYMAKER. “BIM and GIS Integration and Interoperability Based on Semantic Web Technology”. en. In : *Journal of Computing in Civil Engineering* 30.3 (mai 2016), p. 04015043 (cf. p. 23).

- [KDK15] Aftab Ahmed KHAN, Andreas DONAUBAUER et Thomas H KOLBE. “A multi-step transformation process for automatically generating indoor routing graphs from existing semantic 3D building models”. en. In : (2015), pp.20 (cf. p. 18).
- [Kne+22] M. KNEZEVIC, A. DONAUBAUER, M. MOSHREFZADEH et T. H. KOLBE. “Managing urban digital twins with an extended catalog service”. en. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences X-4/W3-2022* (oct. 2022), pp.119-126 (cf. p. 8, 26).
- [Kol09] Thomas KOLBE. “Representing and Exchanging 3D City Models with CityGML”. In : *3D Geo-Information Sciences*. Journal Abbreviation : 3D Geo-Information Sciences. Jan. 2009, p. 15-31 (cf. p. 19, 29).
- [KGP05] Thomas KOLBE, Gerhard GRÖGER et Lutz PLÜMER. “CityGML - Interoperable access to 3D city models”. In : *Geo-information for Disaster Management* (jan. 2005) (cf. p. 15, 82).
- [Kol+20] Thomas H KOLBE, Mandana MOSHREFZADEH, Andreas DONAUBAUER et Kanishk CHATURVEDI. “The Data Integration Challenge in Smart City Projects”. en. In : (2020). Publisher : Technische Universität München (cf. p. 17).
- [KCG18] A. KOUKOFIKIS, V. COORS et R. GUTBELL. “INTEROPERABLE VISUALIZATION OF 3D CITY MODELS USING OGC’S STANDARD 3D PORTRAYAL SERVICE”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences IV-4* (19 sept. 2018), p. 113-118 (cf. p. 26).
- [Kut16] Tatjana KUTZNER. “Geospatial data modelling and model-driven transformation of geospatial data based on uml profiles”. PhD Thesis. Technische Universität München, 2016 (cf. p. 7, 14).
- [KCK20] Tatjana KUTZNER, Kanishk CHATURVEDI et Thomas H. KOLBE. “CityGML 3.0 : New Functions Open Up New Applications”. en. In : *PFG - Journal of Photogrammetry, Remote Sensing and Geoinformation Science* 88.1 (fév. 2020), p. 43-61 (cf. p. 15, 62, 82).
- [Kyz+18] Kostis KYZIRAKOS, Dimitrianos SAVVA, Ioannis VLACHOPOULOS et al. “Geo-Triples : Transforming geospatial data into RDF graphs using R2RML and RML mappings”. en. In : *Journal of Web Semantics* 52-53 (oct. 2018), p. 16-32 (cf. p. 7).
- [La +22] Marcello LA GUARDIA, Mila KOEVA, Filippo D’IPPOLITO et Samer KARAM. “3d data integration for web based open source webgl interactive visualisation”. In : *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLVIII-4/W4-2022* (14 oct. 2022), pp.89-94 (cf. p. 7, 32).
- [LB11] Ruben de LAAT et Léon van BERLO. “Integration of BIM and GIS : The Development of the CityGML GeoBIM Extension”. en. In : *Advances in 3D Geo-Information Sciences*. Sous la dir. de Thomas H. KOLBE, Gerhard KÖNIG et Claus NAGEL. Lecture Notes in Geoinformation and Cartography. Berlin, Heidelberg : Springer, 2011, p. 211-225 (cf. p. 18, 24, 25).

- [Lam+22] D. LAMAS, A. JUSTO, M. SOILÁN et B. RIVEIRO. “3D POINT CLOUD TO BIM : AUTOMATED APPLICATION TO DEFINE IFC ALIGNMENT AND ROADWAY WIDTH ENTITIES FROM MLS-ACQUIRED LiDAR DATA OF MOUNTAIN ROADS”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences X-4/W2-2022* (2022), p. 169-176 (cf. p. 17).
- [LSB23] Binyu LEI, Rudi STOUFFS et Filip BILJECKI. “Assessing and benchmarking 3D city models”. In : *International Journal of Geographical Information Science* (2023), pp.788-809 (cf. p. 3, 15).
- [Li+16] Hang LI, Hua LIU, Yong LIU et Yuan WANG. “AN OBJECT-RELATIONAL IFC STORAGE MODEL BASED ON ORACLE DATABASE”. In : *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLI-B2* (8 juin 2016), p. 625-631 (cf. p. 22).
- [Liu+17] Xin LIU, Xiangyu WANG, Graeme WRIGHT et al. “A State-of-the-Art Review on the Integration of Building Information Modeling (BIM) and Geographic Information System (GIS)”. en. In : *ISPRS International Journal of Geo-Information* 6.2 (fév. 2017), p. 53 (cf. p. 17, 18, 22, 23).
- [Lu+19] Hong-Lei LU, Jia-Xing WU, Yu-Shen LIU et Wan-Qi WANG. “Dynamically loading IFC models on a web browser based on spatial semantic partitioning”. In : *Visual Computing for Industry, Biomedicine, and Art* 2.1 (déc. 2019), p. 4 (cf. p. 22).
- [Mal+20] Eva Savina MALINVERNI, Berardo NATICCHIA, Jose Luis LERMA GARCIA et al. “A semantic graph database for the interoperability of 3D GIS data”. en. In : *Applied Geomatics* (août 2020) (cf. p. 62).
- [MBL20] Bo MAO, Yifang BAN et Björn LAUMERT. “Dynamic Online 3D Visualization Framework for Real-Time Energy Simulation Based on 3D Tiles”. en. In : *ISPRS International Journal of Geo-Information* 9.3 (mars 2020), p. 166 (cf. p. 22).
- [Mar+22] L. MARNAT, C. GAUTIER, C. COLIN et G. GESQUIÈRE. “Py3Dtilers : An Open Source Toolkit for Creating and Managing 2D/3D Geospatial Data”. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences X-4/W3-2022* (2022), p. 165-172 (cf. p. xix, 35, 42, 45, 46, 122).
- [EÖH12] Mohamed EL-MEKAWY, Anders ÖSTMAN et Ihab HIJAZI. “A Unified Building Model for 3D Urban GIS”. In : *ISPRS International Journal of Geo-Information* 1.2 (2012), pp.120-145 (cf. p. 19).
- [MN14] Clement MIGNARD et Christophe NICOLLE. “Merging BIM and GIS using ontologies application to Urban facility management in ACTIVE3D”. In : *Computers in Industry* 65 (1^{er} déc. 2014), p. 1276-1290 (cf. p. 19).
- [Mil12] Timothy MILES-BOARD. “The IFC Standard - A Review Of History, Development, And Standardization”. In : (2012), p. 29 (cf. p. 16).
- [MTL20] Timothy MILLER, Gil TRENUM et Josh LIEBERMAN. *3D Data Container Engineering Report*. <https://docs.ogc.org/per/20-029.html>. [Online ; accessed 23-May-2023]. 2020 (cf. p. 26, 73).

- [MTS20] Timothy MILLER, Gil TRENUM et Ingo SIMONIS. *3D Data Container and Tiles API Pilot Summary Engineering Report*. <https://docs.ogc.org/per/20-031.html>. [Online; accessed 23-May-2023]. 2020 (cf. p. 26, 27).
- [Mor+21] Nicola MORETTI, Claire ELLUL, Fulvio Re CECCONI, Nikolaos PAPAPESIOS et Mario Claudio DEJACO. "GeoBIM for built environment condition assessment supporting asset management decision making". In : *Automation in Construction* 130 (2021) (cf. p. 19).
- [MMG20] Mandana MOSHREFZADEH, Thomas MACHL et David GACKSTETTER. *Towards a Distributed Digital Twin of the Agricultural Landscape*. en. DE : Wichmann Verlag, juin 2020 (cf. p. 8, 26).
- [NK21] S. H. NGUYEN et T. H. KOLBE. "Modelling Changes, Stakeholders and Their Relations in Semantic 3D City Models". English. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. T. VIII-4-W2-2021. Copernicus GmbH, oct. 2021, p. 137-144 (cf. p. 17).
- [OGC06] OGC. *WMS Specification*. <https://www.ogc.org/standards/wms>. [Online; accessed 23-May-2023]. 2006 (cf. p. 44).
- [OGC14] OGC. *WFS Specification*. <https://www.ogc.org/standards/wfs>. [Online; accessed 23-May-2023]. 2014 (cf. p. 44).
- [OGC16a] OGC. *3DPortrayal Specification*. <https://www.ogc.org/standard/3dp/>. [Online; accessed 23-May-2023]. 2016 (cf. p. 26).
- [OGC16b] OGC. *OGC® Catalogue Services 3.0 - General Model*. Catalog SWG. <https://docs.ogc.org/is/12-168r6/12-168r6.html>. Implementation Standard. [Online; accessed 23-May-2023]. 2016 (cf. p. 8, 26).
- [OGC17] OGC. *I3S Specification*. <https://www.ogc.org/standards/i3s>. [Online; accessed 23-May-2023]. 2017 (cf. p. 19).
- [OGC18a] OGC. *CityGML Specification*. <https://www.ogc.org/standard/citygml/>. [Online; accessed 23-May-2023]. 2018 (cf. p. 15, 29, 82, 83, 86).
- [OGC18b] OGC. *CityJSON Specification*. <https://docs.ogc.org/cs/20-072r2/20-072r2.html>. [Online; accessed 23-May-2023]. 2018 (cf. p. 15).
- [OGC18c] OGC. *IndoorGML Specification*. <https://www.ogc.org/standard/indoorgml/>. [Online; accessed 23-May-2023]. 2018 (cf. p. 16).
- [OGC19] OGC. *3DTiles Specification*. <https://www.ogc.org/standards/3DTiles>. [Online; accessed 23-May-2023]. 2019 (cf. p. 19-21, 40).
- [OGC24] OGC. *OGC Glossary*. <https://www.ogc.org/resources/ogc-glossary/>. [Online; accessed 23-May-2023]. 2024 (cf. p. 14).
- [PZL17] Pieter PAUWELS, Sijie ZHANG et Yong-Cheol LEE. "Semantic web technologies in AEC industry : A literature overview". In : *Automation in Construction* 73 (jan. 2017), pp.145-165 (cf. p. 33).

- [PID12] I. PRIETO, J. L. IZKARA et F. J. DELGADO. “From point cloud to web 3D through CityGML”. In : *2012 18th International Conference on Virtual Systems and Multimedia*. 2012 18th International Conference on Virtual Systems and Multimedia. Sept. 2012, p. 405-412 (cf. p. 17).
- [PUG+23] Jey PUGET GIL, Emmanuel COQUERY, John SAMUEL et Gilles GESQUIÈRE. “Versionnement de graphe pour des données urbaines évolutives”. In : *BDA 2023*. Montpellier, France, oct. 2023 (cf. p. 110, 124).
- [Rad+15] Filip RADULOVIC, María POVEDA-VILLALÓN, Daniel VILA-SUERO et al. “Guidelines for Linked Data generation and publication : An example in building energy consumption”. In : *Automation in Construction* 57 (sept. 2015), pp.178-187 (cf. p. 22).
- [Rua02] Anne RUAS. “Généralisation et représentation multiple”. In : *Hermès science* (2002) (cf. p. 28).
- [Sam+23] John SAMUEL, Vincent JAILLOT, Clément COLIN et al. “UD-SV : Urban data services and visualization framework for sharing multidisciplinary research”. en. In : *Transactions in GIS* 27.3 (avr. 2023), p. 841-858 (cf. p. xix, 5, 11, 107, 110).
- [SSG20] John SAMUEL, Sylvie SERVIGNE et Gilles GESQUIÈRE. “Representation of concurrent points of view of urban changes for city models”. en. In : *Journal of Geographical Systems* 22.3 (juill. 2020), p. 335-359 (cf. p. 13).
- [San+22] Thunyathep SANTHANAVANICH, Patrick WUERSTLE, Rushikesh PADSALA et Volker COORS. “Digital 3D City Models Towards Urban Data Platform using OGC 3D GeoVolumes APP”. In : (2022) (cf. p. 27).
- [SBN16] Arne SCHILLING, Jannes BOLLING et Claus NAGEL. “Using glTF for streaming CityGML 3D city models”. en. In : *Proceedings of the 21st International Conference on Web3D Technology*. Anaheim California : ACM, juill. 2016, p. 109-116 (cf. p. 22).
- [SH20] Gerhard SCHROTTER et Christian HÜRZELER. “The Digital Twin of the City of Zurich for Urban Planning”. en. In : *PFG - Journal of Photogrammetry, Remote Sensing and Geoinformation Science* 88.1 (fév. 2020), p. 99-112 (cf. p. 13).
- [Soi+20] Mario SOILÁN, Andrés JUSTO, Ana SÁNCHEZ-RODRÍGUEZ et Belén RIVEIRO. “3D Point Cloud to BIM : Semi-Automated Framework to Define IFC Alignment Entities from MLS-Acquired LiDAR Data of Highway Roads”. In : *Remote Sensing* 12.14 (2020) (cf. p. 17).
- [SK07] Alexandra STADLER et Thomas H. KOLBE. “Spatio-semantic coherence in the integration of 3D city models”. In : *Proceedings of the 5th International ISPRS Symposium on Spatial Data Quality ISSDQ 2007 in Enschede, The Netherlands, 13-15 June 2007*. Sous la dir. d’Alfred STEIN. ISPRS Archives. ISPRS, 2007 (cf. p. 29, 30).
- [STB18] Rudi STOUFFS, Helga TAUSCHER et Filip BILJECKI. “Achieving Complete and Near-Lossless Conversion from IFC to CityGML”. en. In : *ISPRS International Journal of Geo-Information* 7.9 (août 2018), pp.355 (cf. p. 18).

- [Tan+18] Lei TANG, Lin LI, Shen YING et Yuan LEI. “A Full Level-of-Detail Specification for 3D Building Models Combining Indoor and Outdoor Scenes”. In : *ISPRS International Journal of Geo-Information* 7.11 (30 oct. 2018), p. 419 (cf. p. 31).
- [TB14] Guillaume TOUYA et Matthieu BALEY. “Harmonizing Level of Detail in OpenStreetMap Based Maps”. In : (2014) (cf. p. 29).
- [TB13] Guillaume TOUYA et Carmen BRANDO-ESCOBAR. “Detecting Level-of-Detail Inconsistencies in Volunteered Geographic Information Data Sets”. In : *Cartographica : The International Journal for Geographic Information and Geovisualization* 48.2 (juin 2013), p. 134-143 (cf. p. 29).
- [Tra+20] Ba-Huy TRAN, Nathalie AUSSENAC-GILLES, Catherine COMPAROT et Cassia TROJAHN. “Semantic Integration of Raster Data for Earth Observation : An RDF Dataset of Territorial Unit Versions with their Land Cover”. en. In : *ISPRS International Journal of Geo-Information* 9.9 (sept. 2020). Number : 9 Publisher : Multidisciplinary Digital Publishing Institute, p. 503 (cf. p. 4).
- [Tra+16] Ba-Huy TRAN, Christine PLUMEJEAUD, Alain BOUJU et Vincent BRETAGNOLLE. “Towards a semantic framework for exploiting heterogeneous environmental data”. In : *International Journal of Metadata, Semantics and Ontologies* 11 (déc. 2016) (cf. p. 17).
- [UJS21] A. U. USMANI, M. JADIDI et G. SOHN. “Towards the Automatic Ontology Generation and Alignment of BIM and GIS Data Formats”. English. In : *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. T. VIII-4-W2-2021. Copernicus GmbH, oct. 2021, p. 183-188 (cf. p. 15).
- [VLA15] Yannick VERDIE, Florent LAFARGE et Pierre ALLIEZ. “LOD Generation for Urban Scenes”. In : *ACM Transactions on Graphics* 34.3 (8 mai 2015), p. 1-14 (cf. p. 31).
- [Vin+24] C. VINACSO-ALVAREZ D. Colin, J. SAMUEL, S. SERVIGNE, C. BORTOLASO et G. GESQUIÈRE. “A model-driven methodology for integrating heterogeneous geospatial semantic and geometric data”. In : *AGILE : GIScience Series* 1 (2024), p. 2 (cf. p. xix, 51, 122).
- [Vin22] Diego VINASCO-ALVAREZ. “Leveraging Standards in Model-Centric Geospatial Knowledge Graph Creation”. en. In : *The Semantic Web : ESWC 2022 Satellite Events*. Sous la dir. de Paul GROTH, Anisa RULA, Jodi SCHNEIDER et al. Lecture Notes in Computer Science. Cham : Springer International Publishing, 2022, p. 224-233 (cf. p. 56, 57).
- [Vin23] Diego VINASCO-ALVAREZ. “Model-driven Integration of Heterogeneous n-Dimensional Urban Data”. Thèse de doct. Université Lumière - Lyon II, 11 déc. 2023 (cf. p. 15, 23, 56, 110, 113).

- [Wag+20] Anna WAGNER, Mathias BONDUEL, Pieter PAUWELS et Uwe RÜPPEL. “Representing construction-related geometry in a semantic web context : A review of approaches”. In : *Automation in Construction* 115 (juill. 2020), p. 103130 (cf. p. 23, 24).
- [WSW22] Olaf WYSOCKI, Benedikt SCHWAB et Bruno WILLENBORG. *OloOcki/awesome-citygml : Release*. Version ver1.0. Jan. 2022 (cf. p. 15).
- [Xu+20] Zhao XU, Lu ZHANG, Heng LI, Yi-Hsin LIN et Shi YIN. “Combining IFC and 3D tiles to create 3D visualization for building information modeling”. en. In : *Automation in Construction* 109 (jan. 2020), p. 102995 (cf. p. 21, 39).
- [XZX16] Zhao XU, Yang ZHANG et Xiayan XU. “3D visualization for building information models based upon IFC and WebGL integration”. In : *Multimedia Tools and Applications* 75.24 (déc. 2016), p. 17421-17441 (cf. p. 21).
- [YZD21] Jinjin YAN, Sisi ZLATANOVA et Abdoulaye DIAKITÉ. “A unified 3D space-based navigation model for seamless navigation in indoor and outdoor”. In : *International Journal of Digital Earth* (avr. 2021) (cf. p. 19).
- [Yao+18] Zhihang YAO, Claus NAGEL, Felix KUNDE et al. “3DCityDB - a 3D geodatabase solution for the management, analysis, and visualization of semantic 3D city models based on CityGML”. en. In : *Open Geospatial Data, Software and Standards* 3.1 (déc. 2018), p. 5 (cf. p. 22).
- [Zeg21] younes ZEGAOUI. “Détection/reconnaissance d’objets urbains à partir de données 3D multicapteurs prises au niveau du sol, en continu”. Issue : 2021MONT070. Theses. Université Montpellier, oct. 2021 (cf. p. 17).
- [ZCC21] Wenxiao ZHAN, Yuxuan CHEN et Jing CHEN. “3D Tiles-Based High-Efficiency Visualization Method for Complex BIM Models on the Web”. en. In : *ISPRS International Journal of Geo-Information* 10.7 (juill. 2021), p. 476 (cf. p. 22).
- [ZBV18] Chi ZHANG, Jakob BEETZ et de VRIES. “BimSPARQL : Domain-specific functional SPARQL extensions for querying RDF building data”. In : *Semantic Web* (août 2018), p. 1-27 (cf. p. 23).
- [ZCM19] Huangchuang ZHANG, Chengqi CHENG et Shuangxi MIAO. “A Precise Urban Component Management Method Based on the GeoSOT Grid Code and BIM”. In : *ISPRS International Journal of Geo-Information* 8.3 (26 mars 2019), p. 159 (cf. p. 7).
- [ZJ21] Sherong ZHANG et Peiqi JIANG. “Implementation of BIM + WebGIS Based on Extended IFC and Batched 3D Tiles Data : An Application in RCC Gravity Dam for Republication of Design Change Model”. In : *KSCE Journal of Civil Engineering* (19 juill. 2021) (cf. p. 21).
- [Zhu+19a] Junxiang ZHU, Xiangyu WANG, Mengcheng CHEN, Peng WU et Mi Jeong KIM. “Integration of BIM and GIS : IFC geometry transformation to shapefile using enhanced open-source approach”. In : *Automation in Construction* 106 (oct. 2019), p. 102859 (cf. p. 22).

- [Zhu+19b] Junxiang ZHU, Xiangyu WANG, Peng WANG, Zhiyou WU et Mi Jeong KIM. “Integration of BIM and GIS : Geometry from IFC to shapefile using open-source technology”. In : *Automation in Construction* 102 (1^{er} juin 2019), p. 105-119 (cf. p. 22).
- [ZW21] Junxiang ZHU et Peng WU. “Towards Effective BIM/GIS Data Integration for Smart City by Integrating Computer Graphics Technique”. In : *Remote Sensing* 13.10 (12 mai 2021), p. 1889 (cf. p. 22).

Sujet : Gestion et visualisation multiscalaire du territoire au bâtiment. Application à la Gestion et Maintenance Assistée par Ordinateur

Encadrant : Gilles Gesquière

Co-encadrants : Sylvie Servigne, John Samuel

Mots-Clefs : Données 3D, Données urbaines, Données bâtiments, BIM, CIM, Intégration de données, Standards de données, Graphes de connaissances, Multi-échelle, GMAO

Résumé :

La ville et les objets qui la composent, comme les bâtiments, les réseaux d'eau, électriques ou routiers, agrémentent aujourd'hui de véritables jumeaux numériques toujours plus précis jouant un rôle important pour la compréhension des territoires. L'avènement de l'utilisation de Systèmes d'Informations Géographiques (SIG), de Modèles des Données du bâtiments (BIM - Building Information Model) et de la ville (CIM - City Information Model) a entraîné la création d'un nombre important de représentations géospatiales de ces objets urbains, constituées de données géométriques et sémantiques, structurées par de nombreux standards. Ces représentations apportent diverses informations thématiques et spatiales pour décrire ce que sont ces objets physiquement, fonctionnellement, et opérationnellement. Une meilleure compréhension de ces objets urbains peut être apportée par des applications permettant aux utilisateurs d'accéder, de visualiser et d'analyser ces objets urbains en utilisant ces différentes représentations.

Dans cette thèse, nous nous intéressons à la navigation et visualisation interactive multiscalaire sur le web de multiples représentations d'un même objet. Nous prendrons en compte différents standards hétérogènes permettant la représentation de l'intérieur et de l'extérieur d'un bâtiment et d'une ville. Nos deux premières contributions permettent la création de vues navigables et contextuelles de ces représentations hétérogènes dans un même contexte web à l'aide d'approches reposant sur des méthodes d'intégration de données. Pour cela, nous proposons une méthodologie et un outil open-source, Py3DTilers, pour extraire, manipuler et visualiser la géométrie de données géospatiales, ainsi qu'une méthodologie d'intégration de données sémantiques basée modèle, afin de garantir que toutes les informations présentes dans ces données soient disponibles. Notre troisième contribution est la formalisation du concept de variant -instance ou ensemble d'instances représentant le même objet- et une proposition de son utilisation pour référencer et naviguer dans un ensemble de représentations d'un même objet. Enfin, notre dernière contribution s'intéresse au choix de la représentation géométrique à afficher d'un objet selon le contexte utilisateur. Nous proposons une étude des niveaux de détails décrits dans différents standards de données géospatiales ainsi qu'une métrique permettant de décrire la complexité d'une représentation géométrique pour permettre ce choix.

Enfin, cette thèse s'est déroulée en partenariat avec la société Carl software - Berger-Levrault, éditeur de logiciels et de solutions de gestion et maintenance assistée par ordinateur. Un intérêt tout particulier a donc été porté à l'interopérabilité (utilisation de standards), la réutilisabilité (création d'architecture logicielle partagée basée sur des outils open-source) et la reproductibilité des solutions proposées. L'objectif au travers de cette thèse est d'améliorer la compréhension des équipements pour en faciliter la maintenance et la gestion. Cela est permis grâce à la mise en place d'un lien naturel entre les représentations d'un équipement existantes dans un outils de gestion d'équipements et dans diverses sources de données géospatiales, la

visualisation 3D de ces objets et l'exploitation des connaissances que l'on peut retrouver dans ces représentations.

Title : Multi-scale management and visualisation of equipments in buildings and territories. An application in Computerized Maintenance Management System

Key-words : 3D Data, Urban data, Building data, BIM, CIM, data integration, data standards, knowledge graph, multiscale, CMMS

Resume :

Cities and the objects that make them up, such as buildings, water, electricity and road networks, have increasingly precise digital twins that play an important role in understanding territories. The growing use of Geographic Information Systems (GIS), Building Information Model (BIM) and City Information Model (CIM) has led to the creation of a large number of geospatial representations of these urban objects, made up of geometric and semantic data, structured by numerous standards. These representations provide a variety of thematic and spatial information to describe what these objects are physically, functionally and operationally. A better understanding of these urban objects can be provided by applications enabling users to access, visualize and analyze these urban objects using these different representations.

In this thesis, we focus on multiscalar interactive web navigation and visualization of multiple representations of the same object. We will consider various heterogeneous standards for representing the interior and exterior of a building and a city. Our first two contributions enable the creation of navigable and contextual views of these heterogeneous representations in a single web context, using approaches based on data integration methods. To this end, we propose a methodology and an open-source tool, Py3DTilers, for extracting, manipulating and visualizing the geometry of geospatial data, as well as a model-based semantic data integration methodology, to ensure that all the information present in these data can be brought and understood by the users. Our third contribution is the formalization of the concepts of Variant - instance or set of instances representing the same entity- and Variant Identifier to reference and navigate through a set of representations of the same object. Finally, our last contribution focuses on the choice of geometric representation of an object to be displayed, depending on the users' 3D context. We propose a study of the levels of detail described in different geospatial data standards, as well as a metric for describing the complexity of a geometric representation to enable this choice.

Finally, this thesis was carried out in partnership with Carl Software - Berger-Levrault, a publisher of computer-aided maintenance software and asset management solutions. Particular attention was paid to interoperability (use of standards), reusability (creation of shared software architecture based on open-source tools) and reproducibility of the proposed solutions. This thesis aims to improve the understanding of equipment to facilitate its maintenance and management, by allowing the 3D visualization of equipment and the exploitation of the knowledge that can be found in various representations. This is achieved by establishing a natural link between equipment representations existing in this domain and various geospatial data sources.



**CONTRAT DE DIFFUSION ELECTRONIQUE
DE LA THESE de DOCTORAT DE L'UNIVERSITE
LUMIERE LYON 2**

Vu l'arrêté du 25 mai 2016 modifié par l'arrêté du 26 août 2022 fixant le cadre national de la formation et les modalités conduisant à la délivrance du diplôme national de doctorat, en son titre IV, article 25. Vu le Code de l'Education en son article L612-7,

ENTRE

NOM : colin

NOM D'USAGE :

Prénom : clément

Date et Lieu de naissance : 16/02/1997 à Courcouronnes

Nationalité : Française

Demeurant : 418 rue du hameau de la grange, 42800 Chateauneuf

Courriel personnel: clement.colin69@gmail.com

Auteur-e de la thèse de doctorat intitulée :

Gestion et visualisation multiscalaire du territoire au bâtiment.

Application à la Gestion et Maintenance Assistées par Ordinateur

Date de soutenance de la thèse (JJ/MM/AAAA) : 18/06/2024

Directeur de thèse : Gilles Gesquière

Date de dépôt de la version électronique (JJ/MM/AAAA) : 29 / 04 / 2024

ci-après dénommé « l'Auteur-e »

ET

ETABLISSEMENT OPERATEUR : Université Lyon2

ADRESSE: 5, bis rue l'Université 69007 Lyon

Représenté par sa Présidente : Nathalie Dompnier

ci-après dénommé « l'Etablissement »

PREAMBULE

Le présent contrat vise à permettre à l'Etablissement de diffuser la thèse soutenue dans sa version définitive dans le respect des droits de propriété intellectuelle de son Auteur·e. Les autorisations qui y sont attachées ne sont pas exclusives et l'Auteur·e conserve toute liberté de publier ou de diffuser son travail sous quelque forme que ce soit et sous sa propre responsabilité. Le présent contrat ne concerne que les thèses dont le jury a autorisé la reproduction, toutes corrections effectuées.

L'Etablissement de soutenance procède au dépôt de la version validée de la thèse dans ses formats de diffusion et d'archivage, ainsi que du bordereau électronique, dans l'application nationale Star, gérée par l'Agence Bibliographique de l'Enseignement Supérieur.

ARTICLE 1 – Définition

Les parties conviennent, dans l'accord, des définitions respectives suivantes :

Les droits d'Auteur·e : d'après l'article 111-1 du Code de la propriété Intellectuelle : « L'Auteur·e d'une œuvre de l'esprit jouit sur cette œuvre, du seul fait de sa création, d'un droit de propriété incorporelle exclusif et opposable à tous ».

Intranet : s'entend du réseau informatique accessible gratuitement depuis des postes individualisés mis à disposition des enseignants, des chercheurs, des étudiants et du personnel dans l'enceinte d'un Etablissement et à distance après authentification sécurisée.

Internet : s'entend d'un réseau informatique mondial accessible au public sans identification préalable.

Le droit de reproduction : droit de reproduire l'œuvre, intégralement ou partiellement, en nombre illimité, quel que soit le mode d'enregistrement et le type de support.

ARTICLE 2 – AUTORISATIONS ET DROITS CÉDÉS

Le signalement au niveau national (via theses.fr) et la diffusion de la thèse au sein de l'Université Lyon 2 ainsi que les mesures techniques liées à sa conservation et à sa diffusion ne font pas l'objet d'une autorisation par l'Auteur·e mais sont autorisées de fait par la réglementation en vigueur. **En conséquence la thèse est de fait diffusée a minima via le réseau intranet.**

En outre, sous la responsabilité de l'Etablissement de soutenance, l'Auteur·e autorise la diffusion de sa thèse après soutenance et conformément à l'avis du jury **via le réseau internet** (texte intégral avec mention de son nom) par l'intermédiaire de la plate-forme HAL-Theses.

oui non

Si oui, A l'enregistrement de la thèse oui non

Si non, à une date ultérieure (JJ/MM/AAAA) : ___/___/____

La diffusion de la thèse sur Internet se fait selon le contrat Creative Commons suivant :

paternité-pas de modification (toute personne utilisant votre thèse devra mentionner le nom de son auteur et ne pourra pas modifier, ni transformer, ni adapter cette création)

paternité-pas de modification pas d'utilisation commerciale (toute personne utilisant votre thèse devra mentionner le nom de son auteur et ne pourra pas modifier, ni transformer, ni adapter, ni utiliser cette création à des fins commerciales)

Cette autorisation s'entend à titre non exclusif, pour le monde entier, pour toute la durée légale de protection de la propriété littéraire et artistique offerte par la loi française à l'Auteur·e, ses ayant-droits ou représentants, y compris les prolongations qui pourraient être apportées à cette durée. L'autorisation s'entend à titre gratuit.

ARTICLE 3 – CLAUSE DE CONFIDENTIALITE

Si la thèse présente un caractère de confidentialité avéré, sa diffusion est reportée à l'échéance de la période de la confidentialité. Le signalement d'une thèse confidentielle est obligatoire.

ARTICLE 4 – SUSPENSION DE LA DIFFUSION

L'Etablissement se réserve le droit de suspendre la diffusion ou d'effacer l'œuvre de ses serveurs notamment si elle prend connaissance du caractère manifestement illicite du contenu en cause (dans ce cas elle s'engage à prévenir les responsables des autres plateformes institutionnelles sur lesquelles elle aura également déposé la thèse).

L'Auteur·e pourra à tout moment annuler la présente autorisation de diffusion étendue à Internet. A charge pour lui d'en aviser l'Etablissement par courrier postal ou par mel. Dans ce cas, l'Etablissement aura l'obligation de retirer l'œuvre dans un délai maximum de trois mois à la réception de la demande.

ARTICLE 5 – RESPONSABILITE

L'Auteur·e demeure seul responsable, conformément à toutes les dispositions légales applicables en la matière, notamment quant aux citations ou autres usages d'œuvre de tiers.

En particulier :

L'Auteur·e certifie avoir obtenu toutes les autorisations écrites de reproduction et de diffusion, notamment sur internet, nécessaires de la part des titulaires des droits sur les œuvres reproduites, partiellement, ou globalement, (illustrations, extraits multimédias, etc...) et s'engage à relever immédiatement

l'Etablissement de toute action en responsabilité qu'elle pourrait encourir de ce chef.

Si, malgré ses efforts, l'Auteur·e n'a pas pu obtenir toutes les autorisations nécessaires des titulaires des droits sur les œuvres reproduites, partiellement ou globalement, (illustrations, extraits multimédias, etc...), il s'engage à fournir à l'Etablissement en plus de la version complète de sa thèse (en vue d'un archivage) une version expurgée de sa thèse, sans les œuvres non autorisées, afin qu'elle soit diffusée.

Par ailleurs, l'Etablissement ne pourra être tenu pour responsable des agissements illégaux de tiers, ni de la violation d'un éventuel contrat d'édition antérieur non signalé par l'Auteur·e. L'Auteur·e conserve tous ses droits d'ester en justice afin de protéger son droit d'Auteur·e sur l'œuvre.

ARTICLE 6 – LOI APPLICABLE ET DIFFERENDS

Le présent contrat est soumis aux lois et règlements français en vigueur au moment de la signature. Préalablement à toute action en justice, exception faite des actions engagées à titre conservatoire, les Parties conviennent de rechercher dans les délais raisonnables, une solution amiable au différend qui les oppose. En cas de désaccord persistant, tout contentieux lié au présent contrat sera soumis à la juridiction française compétente.

Fait à Villeurbanne _____ en deux exemplaires

Le 29 / 04 / 2024

Pour l'Etablissement, son représentant:

L'Auteur·e : (Prénom – NOM – Signature)

clement colin

A handwritten signature in black ink, appearing to read 'clement colin', written over a horizontal line.