

N° d'ordre : 12/2015–M/INF

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
Ministère de l'enseignement supérieur et de la recherche Scientifique  
Université des sciences et de la Technologie Houari Boumediène

Faculté de l'Electronique et d'Informatique



MEMOIRE

Présenté pour l'obtention du **diplôme** de **MAGISTER**

**En** : INFORMATIQUE

**Spécialité** : Traitement d'images et Systèmes d'informations Géographiques

**Par** : Messaoud BETKA

**Sujet**

Etude et mise en œuvre des passerelles  
entre les modèles vectoriels topologiques  
2D et 3D

Soutenu publiquement, le 16/06/2015, devant le jury composé de :

Mr	Y. SMARA	Professeur	à l'USTHB	Président
Mr	M.BELHADJ-AISSA	Professeur	à l'USTHB	Directeur de mémoire
Mlle	S. AOUAT	Maître de conférences/A	à l'USTHB	Examineur
Mr	K. ATIF	Maître de conférences/A	à l'USTHB	Examineur
Mr	B. BESSAA	Maître Assistant/A	à l'USTHB	Invité

## **Remerciements**

*Au terme de ce travail, je tiens à exprimer ma profonde gratitude et mes sincères remerciements à Monsieur M. Belhadj-Aissa, Professeur à l'USTHB, pour son encadrement, sa patience et son soutien. Je remercie M. B. BESSAA pour les conseils et l'aide qu'il m'a prodigués.*

*Je remercie également Monsieur Y. SMARA, Professeur à l'USTHB, pour ses précieux conseils et m'avoir fait l'honneur de présider le jury, qu'il trouve ici, l'expression de ma grande gratitude.*

*Mes remerciements vont également à Madame S. AOUAT, Maitre de Conférences à l'USTHB, et monsieur K. ATIF, Maitre de Conférences à l'USTHB, qui me font l'honneur de participer au jury de ma soutenance.*

*Je tiens à adresser mes profonds remerciements à Madame Belhadj-Aissa, Professeur à l'USTHB, pour son aide, ses conseils, et ses corrections appréciables.*

*Je tiens à remercier particulièrement Madame Oukil, Maitre de Conférences A à l'USTHB, qui a pris de son temps pour la relecture de ce document et pour ses remarques précieuses.*

*Je remercie également l'ensemble des membres du Laboratoire du Traitement d'Images et Rayonnement (LTIR) de la Faculté d'Électronique et d'Informatique de l'USTHB pour m'avoir aidé chacun à sa façon.*

*Pour finir, je remercie tous ceux qui m'ont aidé de près ou de loin à la réalisation de ce travail.*

# *Dédicaces*

*Je Dédie ce modeste travail*

*A ma chère mère*

*A mon cher père*

*A mes sœurs*

*A mes Frères*

*A ma nièce*

*A mon oncle et ma tante*

*A mes cousins et cousines*

*A mes amis*

*A tous ceux que j'aime.*

## Table des matières

<b>TABLE DES MATIERES</b> .....	<b>I</b>
<b>LISTE DES FIGURES</b> .....	<b>IV</b>
<b>TABLEAUX</b> .....	<b>V</b>
<b>ALGORITHMES</b> .....	<b>V</b>
<b>ACRONYMES</b> .....	<b>VI</b>
<b>INTRODUCTION GENERALE</b> .....	<b>1</b>
<b>CHAPITRE 1 : GENERALITES SUR LES SIG</b> .....	<b>3</b>
<b>Introduction</b> .....	<b>3</b>
<b>1.1. Système d’information géographique</b> .....	<b>3</b>
<b>1.2. Fonctionnement d’un SIG</b> .....	<b>3</b>
<b>1.3. Modes de représentation de l’information géographique</b> .....	<b>5</b>
1.3.1. Mode Raster	5
1.3.2. Mode Vecteur	6
<b>1.4. Composants d’un modèles géo-spatiales</b> .....	<b>6</b>
<b>1.5. Analyse Spatiale</b> .....	<b>7</b>
1.5.1 Relations spatiales	7
1.5.2 Relation Topologique	8
1.5.3 Relation projective	9
<b>1.6. Modèles 2D</b> .....	<b>10</b>
1.6.1 Modèle Spaghetti	10
1.6.2 Modèle Réseau	11
1.6.3 Modèle Topologique	11
<b>Conclusion</b> .....	<b>13</b>
<b>CHAPITRE2 : MODELISATION SPATIALE 3D</b> .....	<b>14</b>
<b>Introduction</b> .....	<b>14</b>
<b>2.1. La 3<sup>ième</sup> Dimension</b> .....	<b>14</b>

## Table des Matières

---

<b>2.2. Modélisation de la 3D et applications .....</b>	<b>15</b>
2.2.1 L'information géographique 3D .....	15
<b>2.3. Modèles spatiaux 3D.....</b>	<b>15</b>
<b>2.4. Modèle par balayage .....</b>	<b>16</b>
<b>2.5. Modèle Volumique .....</b>	<b>17</b>
2.5.1 Modèle « Constructive Solid Geometry » (CSG) .....	17
2.5.2 Primitive Instancing .....	18
<b>2.6. Cell Decomposition.....</b>	<b>19</b>
2.6.1 Modélisation par Voxel .....	19
2.6.2 Modélisation par Octree .....	20
<b>2.7. Modèles par frontières .....</b>	<b>21</b>
2.7.1 Modèle filaire (fil-de-fer) .....	21
2.7.2 Modèle Boundary représentation (B-Rep) .....	22
<b>Conclusion.....</b>	<b>22</b>
<b>CHAPITRE 3 : ETUDES DES MODELES GEOGRAPHIQUES EN 2.X-D ET 3D.....</b>	<b>24</b>
<b>Introduction .....</b>	<b>24</b>
<b>3.1 Modèles 2.5-D.....</b>	<b>25</b>
3.1.1. Modèle TIN .....	25
3.1.2. Modèle 2.75-D .....	26
3.1.3. Modèle 2.8-D .....	26
<b>3.2 Modèles 3D .....</b>	<b>27</b>
3.2.1. 3D FDS (Formal Data Structure) .....	27
3.2.2. Modèle multidimensionnel MD (2.5D, 2D, 3D) .....	28
3.2.3. Modèle 3DGT .....	28
3.2.4. Modèle TEN (Tetrahedral Network) ou 3D TIN .....	29
3.2.5. Modèle SSM (Simplified Spatial Model) .....	30
<b>3.3 Modèles 3D Urbain.....</b>	<b>31</b>
3.3.1. UDM (Urban Data Model) .....	32
3.3.2. Modèle Standard CityGML .....	32
3.3.3. Modèle 3D topologique pour le Réseau Routier .....	34
<b>Conclusion.....</b>	<b>35</b>
<b>CHAPITRE 4 : CONCEPTION ET IMPLEMENTATION DES PASSERELLES DU MODELE VECTORIEL TOPOLOGIQUE 2D VERS LE 3D.....</b>	<b>36</b>
<b>Introduction .....</b>	<b>36</b>
<b>4.1. Notion de Niveau.....</b>	<b>37</b>

## Table des Matières

---

<b>4.2. Passerelle 2D vers 3D pour les structures linéaires</b> .....	<b>37</b>
4.2.1. Vectorisation	37
4.2.2. Affectation des niveaux	38
4.2.3. Génération de la topologie 2D	38
4.2.4. Traitement 2D vers 3D	39
<b>4.3. Description du Modèle Topologique 3D</b> .....	<b>41</b>
4.3.1. Niveaux des primitives en 3D	42
<b>4.4. Passerelle topologique 2D vers la 3D (Structures Surfacades : Bâtiments)</b> .....	<b>43</b>
4.4.1. Objet Volumique en (2D et 3D)	43
4.4.2. Objet Surfaccique (2D et 3D)	44
4.4.3. Affectation des <i>niveaux</i> en 2D	45
4.4.4. Aspect Sémantique	45
<b>4.5. Processus de passage topologique du 2D vers le3D</b> .....	<b>45</b>
4.5.1. Importation et vectorisation	46
4.5.2. Génération de topologie 2D	46
4.5.3. Création des objets selon l'aspect sémantique	46
4.5.4. Traitement du 2D vers le 3D	47
<b>4.6. Modèle Topologique 3D (extension de la topologie 2D)</b> .....	<b>48</b>
4.6.1. Description Sémantique et précision Topologique	49
<b>4.7. Mise en œuvre des passerelles 2D&gt;3D</b> .....	<b>50</b>
4.7.1. <i>SGBD-S et 3D</i> :	50
4.7.2. Choix du SGBD-S :	51
<b>4.8. Passerelle pour le réseau routier (structure linéaire)</b> .....	<b>51</b>
<b>4.9. Mise en œuvre des passerelles 2D&gt;3D pour les bâtiments (structures surfacciques)</b> . 53	
4.9.1. Importation et vectorisation	53
4.9.2. Génération de topologie 2D	54
4.9.3. Création et paramétrage des objets (entités sémantiques)	55
4.9.4. Ajout de détails sémantiques :	56
4.9.5. Construction topologique 2D vers 3D	57
4.9.6. Création de la géométrie Z	61
4.9.7. Construction de la topologie des volumes	62
4.9.8. Visualisation et interrogation topologique en 3D	65
<b>Conclusion</b> .....	<b>67</b>
<b>CONCLUSION ET PERSPECTIVES</b> .....	<b>68</b>
<b>REFERENCES BIBLIOGRAPHIQUES</b> .....	<b>70</b>

### Liste des figures

<b>FIGURE 1.1.</b> LES TROIS VOILETS D'UN SIG [18].....	4
<b>FIGURE 1.2.</b> COUCHES THEMATIQUES .....	4
<b>FIGURE 1.3.</b> MODELISATION D'UN OBJET GEOGRAPHIQUE.....	4
<b>FIGURE 1.4.</b> HIERARCHIE DES RELATIONS SPATIALES .....	8
<b>FIGURE 1.5.</b> MODELE SPAGHETTI .....	10
<b>FIGURE 1.6.</b> DIAGRAMME DE CLASSE DU MODELE RESEAU .....	11
<b>FIGURE 1.7.</b> DIAGRAMME DE CLASSE DU MODELE TOPOLOGIQUE.....	11
<b>FIGURE 1.8.</b> MODELE CONCEPTUEL TOPOLOGIQUE D'EDIGEO .....	12
<b>FIGURE 2.1.</b> CLASSIFICATION DES MODELES 3D .....	16
<b>FIGURE 2.2.</b> REPRESENTATION D'UN CYLINDRE PAR LE MODELE DE BALAYAGE.....	16
<b>FIGURE 2.3.</b> ARBRE DE REPRESENTATION CSG D'UN OBJET .....	17
<b>FIGURE 2.4.</b> ENSEMBLE DE VOXELS.....	19
<b>FIGURE 2.5.</b> PRESENTATION D'UN OBJET EN OCTREE ET SON ARBRE HIERARCHIQUE.....	20
<b>FIGURE 2.6.</b> PRESENTATION AMBIGU PAR LE MODELE FIL-DE-FER .....	21
<b>FIGURE 2.7.</b> MODELISATION B-REP .....	22
<b>FIGURE 3.1.</b> EXEMPLE DE 2.8D AVEC SON EQUIVALENT TOPOLOGIQUE EN 2D [31].....	26
<b>FIGURE 3.2.</b> CONCEPTION DU MODELE 3D FDS, [46] CITE DANS [40] .....	28
<b>FIGURE 3.3.</b> PRIMITIVES TOPOLOGIQUE DU MODELE 3DGT [62] .....	29
<b>FIGURE 3.4.</b> SIMPLIFIED SPATIAL MODEL SSM, CITE DANS [40] .....	31
<b>FIGURE 3.5.</b> RELATION VOLUME-VOLUME [63] .....	31
<b>FIGURE 3.6.</b> URBAN DATA MODEL [16].....	32
<b>FIGURE 3.7.</b> MODELE GEOMETRIQUE DE CITYGML .....	33
<b>FIGURE 3.8.</b> LEVEL OF DETAIL (LOD) DEFINIS PAR CITYGML .....	33
<b>FIGURE 3.9.</b> DIAGRAMME DE CLASSE DU MODELE TOPOLOGIQUE 3D (STRUCTURES LINEAIRES).....	34
<b>FIGURE 4.1.</b> PASSERELLE TOPOLOGIQUE 2D VERS 3D POUR LES STRUCTURES LINEAIRES .....	37
<b>FIGURE 4.2.</b> AMBIGUÏTE D'AFFECTATION DE NIVEAU.....	38
<b>FIGURE 4.3.</b> CONFUSION ENTRE NOEUD-REEL ET NOEUD-VIRTUEL .....	39
<b>FIGURE 4.4.</b> CONFUSION ENTRE DES LIGNES AVEC DES NIVEAUX DIFFERENTS.....	40
<b>FIGURE 4.5.</b> MODELE TOPOLOGIQUE 3D .....	42
<b>FIGURE 4.6.</b> EXEMPLE D'AFFECTATION DES NIVEAUX AUX PRIMITIVES FACES .....	43
<b>FIGURE 4.7.</b> COMPOSITION DES OBJETS AVEC DES PRIMITIVES TOPOLOGIQUES .....	45
<b>FIGURE 4.8.</b> PASSERELLE TOPOLOGIQUE 2D VERS 3D POUR LES STRUCTURES SURFACIQUES.....	46
<b>FIGURE 4.9.</b> MODELE TOPOLOGIQUE 3D (EXTENSION 2D) .....	48
<b>FIGURE 4.10.</b> MODELE SEMANTIQUE (PRESENTATION 2D ET 3D) .....	49
<b>FIGURE 4.11.</b> AFFECTATION DES NIVEAUX/SOUS-NIVEAUX (COUPE VERTICALE D'ESCALIER) .....	49
<b>FIGURE 4.12.</b> COMPOSITION D'ESCALIER.....	50
<b>FIGURE 4.13.</b> EXEMPLES DES PLANS DE NIVEAUX.....	53
<b>FIGURE 4.14.</b> CREATION DE TOPOLOGIE 2D.....	54
<b>FIGURE 4.15.</b> CREATION DE TABLE SEMANTIQUE (EXEMPLE : MUR).....	56
<b>FIGURE 4.16.</b> AJOUT DE DETAILS SEMANTIQUES (VOLE D'ESCALIER) .....	56
<b>FIGURE 4.17.</b> CONSTRUCTION TOPOLOGIQUE 3D D'ESCALIER .....	61
<b>FIGURE 4.18.</b> COLONNE GEOMETRIE 3D DE FACE-IM ET EDGE-IM EN FORMAT WKB ET KT .....	62
<b>FIGURE 4.19.</b> TOPOLOGIE 3D VOLUME-FACE (FACE-N ET FACE-IM) .....	64
<b>FIGURE 4.20.</b> VISUALISATION 3D DES MURS DES NIVEAUX 1 & 3.....	65
<b>FIGURE 4.21.</b> VISUALISATION 3D DES MURS PAR NIVEAU.....	65
<b>FIGURE 4.22. (A) :</b> FACES D'INTERIEUR .....	66
<b>FIGURE 4.22. (B) :</b> FACES D'INTERIEUR.....	66

## Liste des figures & Tableaux & Algorithmes

---

<b>FIGURE 4.24.</b> FACES LIEES A UN NIVEAU PAIR DONNE .....	66
<b>FIGURE 4.23.</b> FACES D'EXTERIEUR.....	66

### Tableaux

<b>TABLEAU 1.1.</b> MATRICE DISJOINT (A, B) .....	9
---	---

### Algorithmes

<b>ALGORITHME 4.1.</b> TRAITEMENT TOPOLOGIQUE 2D>3D DE LA TOPOLOGIE RESEAU ....	52
<b>ALGORITHME 4.2.</b> RECONSTRUCTION TOPOLOGIQUE 3D D'UN OBJET SURFACIQUE DE NIVEAU PAIR .....	58
<b>ALGORITHME 4.3.</b> RECONSTRUCTION TOPOLOGIQUE 3D D'UN OBJET SURFACIQUE DE NIVEAU IMPAIR ORTHOGONALE .....	59
<b>ALGORITHME 4.4.</b> RECONSTRUCTION TOPOLOGIQUE 3D D'UN OBJET SURFACIQUE DE NIVEAU IMPAIR INCLINE .....	59
<b>ALGORITHMES 4.5.</b> RECONSTRUCTION TOPOLOGIQUE EN 3D D'ESCALIER EN DETAILS .....	60
<b>ALGORITHME 4.6.</b> CONSTRUCTION TOPOLOGIQUE DES VOLUMES .....	63

### Acronymes

<b>B-REP</b>	<b>B</b> oundary <b>R</b> eprésentation
<b>CAD</b>	<b>C</b> omputer <b>A</b> ided <b>D</b> rafting / <b>C</b> omputer <b>A</b> ssisted <b>D</b> esign
<b>CAO</b>	<b>C</b> onception <b>A</b> ssistée par <b>O</b> rdinateur
<b>COLLADA</b>	<i><b>COLL</b>aborative <b>D</b>esign <b>A</b>ctivity</i>
<b>CSG</b>	<b>C</b> onstructive <b>S</b> olid <b>G</b> eometry
<b>DAO</b>	<b>D</b> essin <b>A</b> ssisté par <b>O</b> rdinateur
<b>DEM</b>	<b>D</b> igital <b>E</b> levation <b>M</b> odel
<b>DXF</b>	<b>D</b> rawing <b>eX</b> change <b>F</b> ile
<b>ESRI</b>	<b>E</b> nvironmental <b>S</b> ystems <b>R</b> esearch <b>I</b> nstitute
<b>FDS</b>	<b>F</b> ormal <b>D</b> ata <b>S</b> tructure
<b>GML</b>	<b>G</b> éographic <b>M</b> arkup <b>L</b> anguage
<b>ISO</b>	<b>I</b> nternational <b>O</b> rganization for <b>S</b> tandardization
<b>KML</b>	<b>K</b> eyhole <b>M</b> arkup <b>L</b> anguage
<b>LIDAR</b>	<b>L</b> ight <b>D</b> etection <b>A</b> nd <b>R</b> anging
<b>LOD</b>	<b>L</b> evel <b>O</b> f <b>D</b> etail
<b>MIF</b>	<b>M</b> ap <b>I</b> nfo <b>I</b> nterchange
<b>MNT</b>	<b>M</b> odèle <b>N</b> umérique de <b>T</b> errain
<b>OGC</b>	<b>O</b> pen <b>G</b> eospatial <b>C</b> onsortium
<b>SGBD</b>	<b>S</b> ystème de <b>G</b> estion de <b>B</b> ase de <b>D</b> onnées
<b>SGBD-S</b>	<b>S</b> ystème de <b>G</b> estion de <b>B</b> ase de <b>D</b> onnées <b>S</b> patiales
<b>SIG</b>	<b>S</b> ystème d'Information <b>G</b> éographique
<b>SQL</b>	<b>S</b> tructured <b>Q</b> uery <b>L</b> anguage
<b>SSM</b>	<b>S</b> implified <b>S</b> patial <b>M</b> odel
<b>TEN</b>	<b>T</b> etrahedral <b>N</b> etworks
<b>TIN</b>	<b>T</b> riangular <b>I</b> rregular <b>N</b> etwork
<b>UDM</b>	<b>U</b> rban <b>D</b> ata <b>M</b> odel
<b>UML</b>	<b>U</b> nified <b>M</b> odeling <b>L</b> anguage.
<b>UTM</b>	<b>U</b> niversal <b>T</b> ransverse <b>M</b> ercator
<b>VRML</b>	<b>V</b> irtual <b>R</b> eality <b>M</b> odeling <b>L</b> anguage
<b>WKT</b>	<b>W</b> ell <b>K</b> nown <b>T</b> ext
<b>XML</b>	<b>eX</b> tensible <b>M</b> arkup <b>L</b> anguage
<b>X3D</b>	<b>eX</b> tensible <b>3</b> <b>D</b> imension

### **Introduction Générale**

Les systèmes d'informations géographiques (SIG) ont connu une grande évolution et sont utilisés dans plusieurs domaines ces dernières années. Les modèles spatiaux ne prennent que certains aspects de la réalité qui ont un rôle décisif dans les applications considérés. En effet les premiers modèles géographiques sont limités à la représentation des données dans l'espace 2D tandis que la réalité est en 3D. De nos jours ces modèles limités ne sont pas suffisants pour répondre aux besoins actuels en informations géographiques. La visualisation en 3D, actuellement, est effectuée sur la base du modèle 2D auquel on a ajouté l'information d'altitude (z) comme attribut. L'ajout de la 3D aux bases de données 2D permet d'afficher l'aspect 3D des données mais sans aucune information topologique selon l'axe Z. Ces modèles ne sont pas du vrai 3D mais sont restreints au 2.5D.

D'un autre côté la modélisation 3D pour des fins de visualisation a connu une grande évolution dans les autres domaines non géographiques comme les jeux vidéo et le cinéma. Cet éclatement en 3D a envahi plusieurs domaines grâce à la puissance des ordinateurs actuels et les dispositifs d'affichage. Parmi ces disciplines, on trouve les systèmes CAO et DAO qui ont quelques fonctionnalités communes avec les SIG. L'adaptation des modèles 3D, qui sont issues des systèmes CAO et de l'infographie, aux besoins des SIG représentent une grande partie des travaux de recherche dans ce domaine. Ces modèles ne sont pas satisfaisants car ils sont orientés beaucoup plus vers la visualisation tridimensionnelle. Ils sont limités en termes d'analyse spatiale qui est, entre autres, la raison d'être de tous les SIG. L'ajout de la topologie 3D à ces modèles reste un cadre de recherche souffrant de l'absence d'une véritable implémentation dans les logiciels professionnels. Ces modèles ne sont pas compatibles avec les outils de stockage et les modèles existants en 2D. Leur implémentation nécessite la création de nouveaux standards, structures des données et moyen de stockage appropriés, ainsi que l'acquisition en 3D. Bien qu'il existe quelques moyens d'acquisition en 3D mais ils ne sont pas encore en large disponibilité ou ont un cout élevé. Donc les modèles et les données ainsi que les formats, les moyens de stockage en 2D doivent être exploités dans les modèles 3D pour garantir un passage avec le minimum d'investissement. C'est dans ce contexte que s'inscrit notre travail d'initiation à la recherche qui consiste en la conception et l'implémentation des passerelles des modèles topologiques 2D vers le 3D.

Le processus de passage à la 3D utilise les bases de données topologiques 2D pour construire une topologie en 3D. La topologie 2D doit être préservée en ajoutant l'information topologique selon la verticale. Dans ce travail, nous avons étudié les différents modèles 3D existants afin de construire un modèles topologique 3D comme extension des modèles topologiques 2D. La particularité de ce modèle est le passage topologique à la 3D à partir de

## **Introduction Générale :**

---

données vectorielles topologiques en 2D. Nous nous intéressons à l'application des passerelles du 2D vers le 3D dans un milieu urbain pour les structures linéaires comme le réseau routier, les structures surfaciques et volumiques telles que les bâtiments.

Le présent mémoire est organisé en quatre chapitres. Le premier chapitre est une introduction au domaine des SIG, et à la modélisation géographique ainsi qu'aux principaux modèles qui existent en 2D. Nous commençons ce chapitre par un bref rappel sur les généralités du SIG. Ensuite, nous introduisons une présentation des 3 modèles principaux en 2D. le second chapitre présente la 3D et les modèles géométriques 3D de base qui sont issus des différents domaines spatiaux tels que l'infographie et les systèmes CAD. Le troisième chapitre est consacré à l'étude des différents modèles géographiques qui sont de deux types : les modèles en 2.XD (2.5D, 2.75D, 2.8D) qui représentent la suite ou l'extension des modèles topologiques 2D, et les différents modèles 3D qui sont conçus à partir des modèles 3D des autres domaines spatiaux. Le quatrième chapitre présente notre travail qui consiste en la conception et la mise en œuvre des passerelles du passage du modèle vectoriel topologique 2D vers le modèle topologique 3D. Nous terminons par une conclusion générale résumant ce travail et une synthèse sur la suite de nos travaux.

## Chapitre 1 : Généralités sur les SIG

### Introduction

Le fort développement informatique et la disponibilité croissante de l'information géographique ont donné lieu à l'avènement de la géomatique. Ce boom de l'information géographique et de la géomatique a conduit dans de nombreuses applications à l'utilisation croissante des Systèmes d'Information Géographique (SIG). Le SIG est un système moderne permettant de rassembler, stocker, visualiser et traiter des données, de sources et d'échelles différentes, permettant la description du milieu naturel et les différentes activités (activités agricoles, industrielles, touristiques, urbaines, etc.). Il permet aussi de les combiner pour produire une cartographie à la fois globale, locale et synthétique. Le SIG est un système d'analyse spatiale qui met à la disposition du thématique voire du gestionnaire, des outils d'analyse rapide de l'information. Ce système est utilisé comme un outil de diagnostic et d'aide à la prise de décision.

D'après les divers domaines d'applications, le SIG englobe certaines fonctionnalités de plusieurs disciplines telles que la cartographie, le dessin assisté par ordinateur (DAO), les bases de données, la télédétection et les spatiaux-cartes.

### 1.1. Système d'information géographique

Nés il y a une trentaine d'années, les systèmes d'informations géographiques ont évolué très rapidement, surtout durant les 20 dernières années, grâce aux progrès technologiques sans cesse grandissants. Leur origine récente et leur progression rapide rendent difficile l'établissement d'une définition adéquate unique.

Une des définitions largement utilisées pour décrire les SIG est celle fournie par le centre américain d'information géographique et d'analyse [29] :

*« Un SIG est un ensemble de matériels, de logiciels et de procédures conçus pour permettre la collecte, la gestion, la manipulation, l'analyse, la modélisation, l'affichage de données à référence spatiale afin de résoudre des problèmes complexes d'aménagement et de gestion ».*

### 1.2. Fonctionnement d'un SIG

D'après [32] [42], le SIG peut être défini comme un ensemble de procédures informatisées qui offrent aux professionnels des fonctions adaptées pour le stockage, l'extraction, le traitement et l'affichage des données à référence spatiale à l'aide de trois volets :

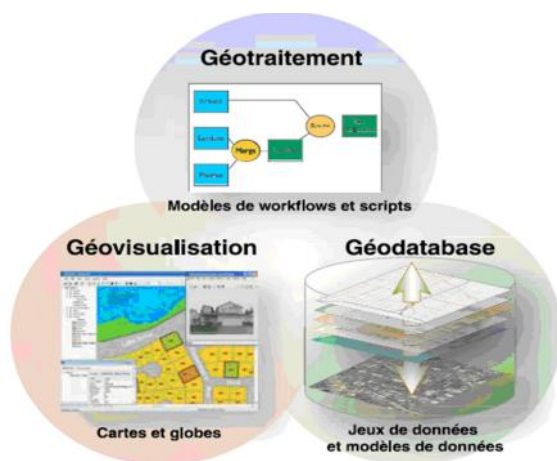


Figure 1.1. Les trois Volets d'un SIG [18]

- a. **Volet géodatabase** : Un SIG se connecte à une base de données spatiale. Cette dernière décrit les objets manipulés géographiquement et sémantiquement.
- b. **Volet géovisualisation** : Un SIG est un ensemble de cartes et de vues qui montrent les entités et leurs relations à la surface de la terre. Il est possible d'élaborer différentes vues cartographiques afin d'effectuer des requêtes, des analyses et de modifier les informations géographiques.
- c. **Volet géotraitement** : Un SIG comprend des outils de transformation des données qui produisent des informations à partir des données existantes en appliquant des fonctions analytiques.

Un SIG contient généralement plusieurs sortes d'objets géographiques appelés Informations Géographiques, organisés sous forme de couches superposées, où chacune englobe les objets de même nature (routes, bâtiments, cours d'eau, limites de communes, entreprises,...). Chaque objet est constitué d'une forme (géométrie de l'objet) et d'une description (appelé aussi sémantique). Le niveau géométrique est la description de la position et de la forme des objets. La position peut s'exprimer par la latitude et la longitude des objets (ou des points qui composent ces objets) ou par des coordonnées x, y dans un système de projection.

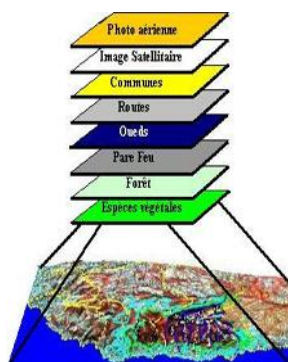


Figure 1.2. Couches thématiques

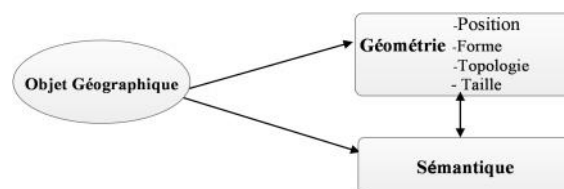


Figure 1.3. Modélisation d'un objet géographique

Ainsi la composante géométrique englobe les propriétés spatiales : la position, la forme, la taille et la topologie, tandis que la composante thématique associée décrit certaines propriétés non spatiales de l'objet qui sont en relation avec sa fonction, son état et l'utilité de l'objet dans la réalité. Pour décrire l'aspect dynamique des objets, certains modèles ajoutent une composante qui représente l'état des objets en fonction du temps.

Les objets peuvent être représentés sous forme de points (villes, entreprises,...), de lignes (routes, chemins de fer,...) et de polygones (communes, occupation du sol,...).

A chaque objet est attribuée une fiche contenant des informations de type alphanumérique. Ces informations décrivent l'objet (nom de la ville, code de la commune, type de l'occupation du sol,...). Ces fiches permettent de stocker des informations qui décrivent les objets.

### 1.3. Modes de représentation de l'information géographique

Il existe deux modes de représentations de l'information géographique : « Le mode Raster » et « le mode vecteur ». Ce choix dépend de la disponibilité des données, et / ou du niveau de détail de représentation exigé.

#### 1.3.1. Mode Raster

C'est un mode à structure image ou maille, qui représente l'espace régulier par des unités élémentaire (*pixel*), ou cellule, ce qui forme une grille à maille, dans laquelle une valeur radiométrique est attribuée à chaque cellule qui représente un attribut lié à cette position.

Le mode Raster est considéré comme un modèle *space-filling*, dans lequel chaque emplacement dans la région de l'étude correspond à une cellule dans le raster [53]. La représentation de tout l'espace envisagé de cette façon nécessite un volume de stockage important. Mais ce mode présente une simplicité au niveau de la structure de données représentée sous forme d'une matrice dont chaque cellule est référencée par sa position en ligne et en colonne.

L'exactitude de la forme des objets avec le mode raster est limitée. Elle dépend fortement de la résolution spatiale. Cette dernière est définie par la taille réelle, exprimé en mètres (m), de l'unité surfacique élémentaire sur terre représentée par le (*pixel*). Cette taille, du plus petit objet représenté, est définie par les moyens d'acquisition des données.

Le raster est utilisé généralement pour représenter des phénomènes spatiaux continus ainsi que pour les applications qui nécessitent la superposition d'un grand nombre de couches spatiales.

Les structures des données du mode raster sont exceptionnellement fortes dans le cas où les limites et renseignements du point ne sont pas bien définis, [28].

**Avantages :** structure de données simples, affichage plus proche de la réalité continue, facile à utiliser dans le cas des différentes couches thématiques.

**Inconvénients** : représentation spatiale non exacte, grand volume des données, absence de la topologie explicite, ambiguïté dans la forme.

### 1.3.2. Mode Vecteur

C'est un modèle à structure objet, le plus répandu dans le domaine des SIG. Du point de vue géométrique le mode vecteur sert à décrire en langage numérique des unités de forme irrégulières caractérisées par des relations spatiales, [56].

Le mode vecteur garde la représentation de la cartographie traditionnelle, dont les objets spatiaux sont représentés par des entités de base telles que le point, la ligne et la région (polygone) qui sont référencées spatialement par un système de coordonnées cartésien.

Le point joue le rôle central, il représente une position sans dimension. A partir au moins de deux points connectés, une ligne est représenté. La surface est construite à partir des points/lignes connectées qui forment une boucle qu'on appelle aussi polygone. Cette structure qui semble complexe par rapport au raster, donne une forme exacte et délimite de façon concise les objets, avec une quantité minimale d'informations ce qui réduit le volume de stockage des données mais avec un cout élevé pour le traitement.

Dans ce mode, les relations entre les objets peuvent être stockées de façon explicite, ce qui donne une possibilité d'effectuer des opérations d'analyse spatiales de nature qualitative (par exemple l'analyse topologique) en plus des opérations d'analyse quantitatives (métrique).

**Avantages** : minimise le volume des données, présence des relations comme la topologie, supporte mieux la haute résolution que le mode raster, permet un affichage de bonne qualité.

**Inconvénients** : structure de données plus complexe, nécessite une grande capacité de traitement.

## 1.4. Composants d'un modèles géo-spatiales

On distingue 5 composantes d'un modèle géographique donné, [2] :

**Types d'objets** : Sont définis par l'ensemble des classes de différentes entités spatiales.

**Relations** : Décritent les différentes relations spatiales entre deux objets spatiaux ou plus.

**Attributs descriptives** : Sont des données sémantiques associées aux objets spatiaux. La modélisation thématique dépend du contexte du domaine étudié.

**Conventions** : Représentent l'ensemble des règles et contraintes, qui gèrent et contrôlent le contenu, la structure, l'intégrité, et le comportement du modèle. Une convention est appliquée à tout le modèle comme, par exemple , la convention qui affirme que chaque entité géographique doit contenir des objets qui ont un et un seul type géométrique. Elles donnent la règle qui interdit l'appartenance d'un objet de forme zone à une classe de type linéaire. [47].

**Opérations** : Sont l'ensemble des outils, existant dans le modèle, qui nous permettent d'extraire des informations pour effectuer des changements sur les objets. Certaines de ces opérations sont standards comme l'ajout, la recherche, la suppression et la combinaison entre ces opérations de base pour effectuer des traitements plus compliqués.

### 1.5. Analyse Spatiale

La force d'un système d'information géographique réside dans sa capacité à analyser les données spatiales [40]. Cette force repose sur l'information quantitative et qualitative des objets géographiques.

Une grande partie des opérations d'analyse porte sur les relations spatiales entre les objets géographiques, donc celles-ci représentent un élément clé dans la modélisation spatiale.

#### 1.5.1 Relations spatiales

Les relations spatiales peuvent être classées par catégorie en fonction de trois niveaux différents de représentation : le niveau géométrique, le niveau informatique, et le niveau utilisateur. Le niveau géométrique est une représentation abstraite en termes mathématiques des objets, il représente le niveau le plus primitif. Les deux autres niveaux se rapporteront toujours à la définition d'une relation spatiale au niveau géométrique [13].

Le niveau utilisateur est lié au contexte sémantique des objets géographiques, donc il est plus proche des langages naturels et lié à la perception humaine, qui est plus qualitative que quantitative.

Les relations spatiales, sont distinguées, d'une part, en relations métriques (quantitative) qui correspondent à la géométrie euclidienne, et sont définies dans un espace métrique. Et d'autre part en relations topologiques (qualitative) définies dans un espace topologique.

Il existe un troisième type de relations spatiales de nature qualitatives, qui est souvent confondu avec la topologie dans le contexte géographique. Ce type correspond à la géométrie projective, il décrit les relations d'orientation. Ces relations sont de nature qualitative car elles n'ont pas besoin de mesures précises pour être expliquées [24].

La relation projective représente une étape intermédiaire entre la métrique et la topologie. Donc, les trois espaces ne sont pas indépendants entre eux. Ils sont liés hiérarchiquement (figure 1.4). Si une propriété géométrique est évaluée dans l'espace topologique, la même propriété restera valable dans l'espace projectif et l'espace euclidien, mais l'inverse n'est pas nécessairement vrai, [13]. Cette hiérarchie est définie selon l'approche axiomatique dans les espaces géométriques. Un espace peut être construit à partir du précédent en ajoutant davantage d'axiomes [35], donc on peut dire que :

$$\text{Topologie} \subset \text{Projective} \subset \text{Mét}$$

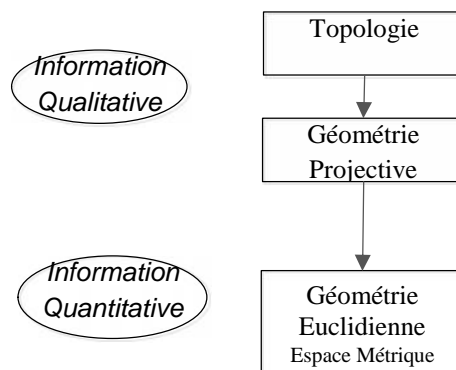


Figure 1.4. Hiérarchie des Relations Spatiales

### 1.5.2 Relation Topologique

La topologie est étymologiquement grecs : ‘topos-logos’, ce mot composé signifie littéralement ‘l’étude de lieu’, la topologie comme discipline mathématique a connu ses débuts avec la théorie des graphes, en 1736 grâce aux travaux d’Euler, sur ‘les ponts de Königsberg’ celui-ci a définis la topologie comme : « la part de la géométrie ne relevant pas des mesures et des distances ».

D’après *Augustus Möbius*, cette discipline mathématique s’intéresse à l’étude des propriétés de figures invariantes sous des transformations topologiques [23].

Une transformation topologique est une transformation continue et biunivoque qui préserve la topologie d’un objet quand il subit une déformation du genre translation, rotation, étirement ou rétrécissement sans déchirement ni faire de trous.

La topologie est définie mathématiquement sur un ensemble  $E$  comme une partie  $T$  de sous ensemble de  $E$  (appelés ouverts), qui vérifie les conditions suivantes :

- 1 /  $w \in T, E \in T$
- 2 / l’intersection de deux élément de  $T$  est un élément de  $T$
- 3 / la réunion des éléments de  $T$  est un élément de  $T$

Un espace topologique est un couple  $(E, T)$  ou  $E$  est un ensemble et  $T$  est une Topologie sur  $E$

La topologie joue un rôle importants dans les SIG, en terme de cohérence et d’intégrité des données, et lors des opérations d’analyse qui s’appuient sur les relations spatiales entres les objets. La topologie est utilisée pour décrire les relations d’adjacence, d’inclusion et d’intersection entre les objets géographiques. Ces relations sont calculées et validées à partir de la géométrie. Elles sont représentées explicitement par les primitives topologiques qui correspondent aux primitives géométriques, qui sont le nœud (point), l’Arc (ligne) et la surface (polygone).

Des opérateurs topologiques sont conçus pour déduire les relations topologiques entre deux objets A, B à partir de la géométrie par exemple : Intersect(A, B), Touch(A, B), Equal(A, B), Disjoint(A, B)...etc.. Ces opérateurs topologiques s'appuient sur des modèles comme le modèle de 4-intersections développé par [52], et le modèle de 9-intersections développé par [25].

**1.5.2.1 Modèle de 4-intersections**

Une relation entre deux objets A, B est calculée à partir d'un opérateur d'intersection entre les deux parties des deux objets A, B qui sont : l'intérieur (A°) et frontière (∂A). Ce modèle est représenté par une matrice 2\*2, dont chaque valeur vaut soit (∅) si l'intersection est vide ou non (≠ ∅) si l'intersection est non vide, ce qui donne 2<sup>4</sup>, 16 cas possibles pour la relation topologique entre deux objets A et B.

$$R(A, B) = \begin{matrix} A^\circ \cap B^\circ & A^\circ \cap \partial B \\ \partial A \cap B^\circ & \partial A \cap \partial B \end{matrix}$$

**1.5.2.2 Modèle de 9-intersections**

Une relation entre deux objets A, B est calculée à partir d'un opérateur d'intersection entre les 3 parties des deux objets A, B qui sont : l'intérieur (A°) et frontière (∂A), ainsi que l'extérieur (A<sup>-</sup>). Ce modèle est représenté par une matrice 3\*3, dont chaque valeur vaut soit (∅) si l'intersection est vide ou non (≠ ∅), ce qui donne 2<sup>9</sup> (512) cas possibles pour la relation topologique entre deux objets A et B.

$$R(A, B) = \begin{matrix} A^\circ \cap B^\circ & A^\circ \cap \partial B & A^\circ \cap B^- \\ \partial A \cap B^\circ & \partial A \cap \partial B & \partial A \cap B^- \\ A^- \cap B^\circ & A^- \cap \partial B & A^- \cap B^- \end{matrix}$$

Chacun des objets A, B peut avoir une dimension : 0(point), 1(ligne) ou 2(polygone). Par exemple la relation Disjoint (A, B)=1, est représentée par la matrice de 9-intersections, dans le tableau suivant :

Objet A/ Objet B	B°	B	B <sup>-</sup>
A°	0	0	1
A	0	0	1
A <sup>-</sup>	1	1	1

**Tableau1.1.**Matrice Disjoint (A, B)

**1.5.3 Relation projective**

**Climentini et Laurini** dans [13] ont effectué une distinction entre les relations topologiques et projectives. Comme les relations projectives sont plus précises que les relations topologiques, elles décrivent des relations qualitatives qui ne sont pas décrites par la topologie telles que : « à droite de », « devant », « arrière », « entouré par » et ainsi de suite.

Le modèle de départ qui représente les relations projective est le modèle de 4-intersections qui s'appuie sur une relation de base qui est la colinéarité entre trois points, pour décrire les différentes relations projectives entre 3 régions du plan [14].

## 1.6. Modèles 2D

Au niveau informatique les informations qualitatives et quantitatives sont reflétées par les structures des données. Il existe différents modèles conceptuels pour modéliser les différents aspects géométriques des objets géographiques. Les principaux modèles selon l'ordre chronologique d'apparitions sont :

### 1.6.1 Modèle Spaghetti

C'est un modèle qui décrit la partie spatiale des entités géographiques tout simplement par l'aspect métrique où aucune information qualitative n'est modélisée explicitement. Les primitives géométriques sont modélisées par leurs coordonnées géographiques. Chaque point est représenté par ses coordonnées x, y, les lignes et polygones sont représentés par un ensemble de points.

Ce modèle ne donne aucune information de voisinage entre les objets, les lignes et les polygones. Les objets sont représentés d'une manière indépendante, sans considération de l'intersection qui existe entre eux et sans contrainte. Cette simplicité de représentation marque un avantage qui réside dans l'accessibilité rapide. Ce qui est utile pour l'affichage. Mais le stockage des données avec ce modèle souffre de la redondance et crée une incohérence des données. Ce modèle reste insuffisant et inefficace pour l'analyse spatiale. Pour effectuer des opérations d'analyse, du type « la définition de la partie commune entre deux objets » on doit utiliser beaucoup de calculs mathématiques basés sur l'information métrique.

Le modèle spaghetti est anarchique et ne modélise que des lignes par un ensemble de coordonnés, en plus des points, la notion de polygone est absente [59], une structure surfacique est modélisée par un ensemble de points.

Cependant ce modèle a connu plusieurs changements, comme l'intégration de la notion de polygone. La figure.1.5 montre un diagramme de classe d'un modèle de spaghetti polygonale :

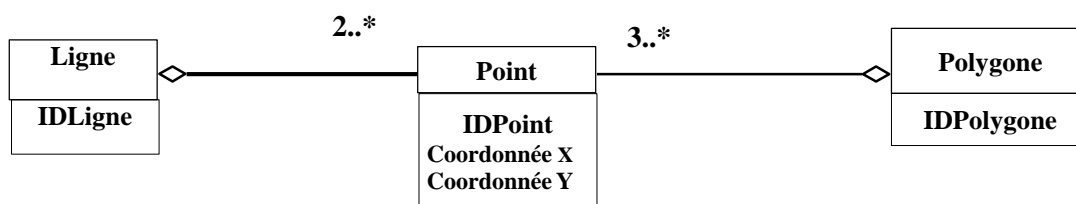


Figure 1.5. Modèle Spaghetti

### 1.6.2 Modèle Réseau

Ce modèle représente une version évoluée du modèle spaghetti par l'intégration des primitives topologiques de la théorie des graphes qui sont la notion de Nœud et d'Arc. Chaque arc est délimité par exactement deux Nœuds qui représentent ses extrémités (un nœud début et un nœud de fin), ce qui rend le réseau comme un graphe orienté. Les nœuds sont aussi utilisés pour représenter les objets ponctuels. L'intégration de l'information qualitative de Nœud et d'arc définit l'interconnexion dans le réseau entre ses éléments, ce qui permet de faire l'analyse sur les structures linéaires (en réseau) de façon plus efficace, comme le calcul du plus court chemin dans un réseau routier. Mais les structures surfaciques sont représentées seulement par la géométrie, comme ensemble de coordonnées (x, y), sans aucune information topologique comme dans le modèle Spaghetti. La structure de données de ce modèle est la suivante :

*Nœud (Coordonnée x : Réel, Coordonnée Y : Réel).*

*Arc (Nœud Début, Nœud Fin, Géométrie Ligne : {Coordonnées : (X, Y)}).*

*Polygone (Géométrie Polygone : {Coordonnées : (X, Y)}).*

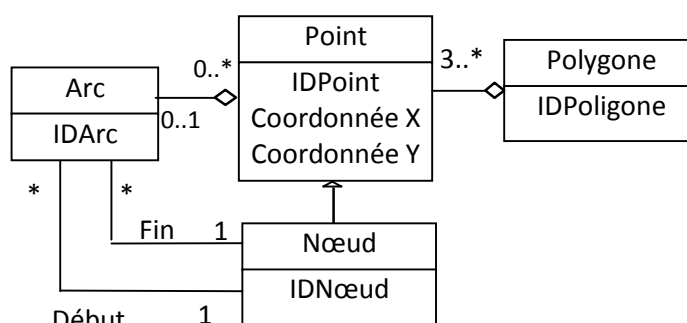


Figure 1.6. Diagramme de classe du modèle réseau

### 1.6.3 Modèle Topologique

Ce modèle intègre les notions de Nœud, Arc, et Face, avec toutes les relations qui peuvent exister entre eux. Ces relations sont stockées de façon explicite et sans redondance, ce qui permet de garantir la cohérence des données, et offre une efficacité lors de l'analyse spatiale.

Un modèle topologique général est présenté par un diagramme de classe dans la figure 1.7.

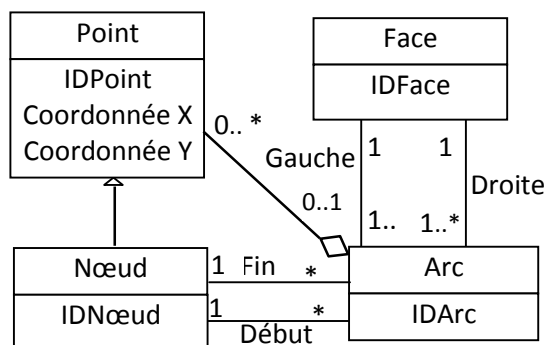


Figure 1.7. Diagramme de classe du modèle topologique

La cardinalité 1 du coté Nœud, et Face dans la relation avec l’arc correspond à l’ajout des Attributs Nœud Début/Fin, et Face (Droite, Gauche) dans la table d’arc dans le modèle logique équivalent :

*Arc (IdArc, Nœud-Debut, Nœud-Fin, Face-Droite, Face-Gauche, Géométrie-Ligne : {Ensemble des Points}).*

*Nœud (Coordonnée x : Réel, Coordonnée Y : Réel).*

*Face (Géométrie-Polygone : {Ensemble des Arcs}).*

Il y a plusieurs standards qui intègrent l’information topologique, comme le modèle topologique du standard Français EDIGéo [3], [4] présenté par un diagramme entité-association, dans la figure 1.8, [26], [17].

En plus des primitives de base, le modèle présente l’association (*présenté par*) avec les objets géographiques. Les cardinalités (0, n) et (1, n) montrent la représentation des entités géographiques à partir d’un ou plusieurs primitives de base. Dans la partie objet, une association (composé de) est défini avec les cardinalités (0, n) pour modéliser des objets complexes à partir des objets simples. Un nœud isolé est distingué, et peut appartenir à une surface. Du point de vue des dimensions des objets, chaque primitive de base est modélisée par ses limites à partir de primitives de dimensions inférieures, jusqu’au Nœud (Point) qui représente l’élément le plus primitif sans dimensions.

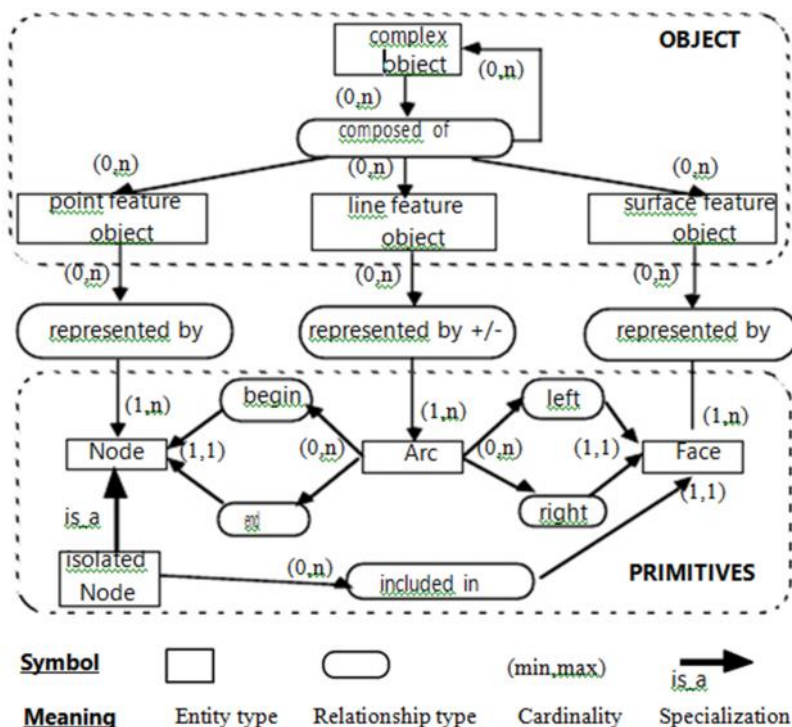


Figure 1.8. Modèle conceptuel topologique d’EDIGéo

**Conclusion**

Dans ce chapitre, nous avons décrit les SIG, et leurs fonctionnalités de base. Celles-ci sont nombreuses, multidisciplinaires, liées et influencées les unes par les autres. Parmi elles, la modélisation qui est considérée comme le cœur et le centre de masse d'un SIG, ainsi que l'analyse spatiale qui représente la raison d'être du SIG, dans laquelle réside sa puissance.

La divergence qui existe, entre le milieu extérieur qui alimente les SIG par les données et le milieu intérieur d'un SIG, qui intègre stocke et gère l'information géographique, est reflétée dans la modélisation par l'existence de deux modes différents de représentations des données spatiales, dont le mode raster est dicté par un milieu externe, qui est la plus grande source d'acquisition des données, et le mode vectoriel le plus répandu et plus avantageux, car il représente l'information géographique de façon concise et appropriée aux besoins des SIG.

D'un autre côté, l'importance de l'analyse spatiale qui représente une spécificité des SIG, réside dans sa capacité à satisfaire les différents besoins de l'utilisateur en termes de production de l'information géographique utile à partir des données acquises.

## Chapitre2 : Modélisation spatiale 3D

### Introduction

Bien que le monde réel soit en 3D, sa représentation numérique, ainsi que sur les supports de visualisation (écrans, papier) sont limitées à l'espace 2D. L'intégration de la troisième dimension dans le domaine géographique, pour une représentation réaliste, constitue un thème de recherche d'actualité. Les besoins en 3D dans les SIG ne sont apparus que dernièrement, après la généralisation des outils SIG 2D, et l'apparition des nouvelles technologies de l'acquisition des données géographiques en 3D. L'évolution des technologies de l'informatique en terme de capacité de stockage, de rapidité des processeurs dans le traitement et la qualité d'affichage via les écrans et les cartes graphiques de nouvelle génération ont permis aux logiciels du type CAO et DAO dont les SIG partagent certaines fonctionnalités, un grand progrès en terme de modélisation et visualisation 3D.

Dans ce contexte, nous présentons dans ce chapitre la notion de la 3<sup>ème</sup> Dimension, les domaines spatiaux concernés par celle-ci, ainsi que les modèles de base 3D existants, qui sont dans la majorité limités à l'aspect géométrique, et destinées à d'autres besoins différents que ceux du SIG. Plusieurs chercheurs dans la géomatique se sont inspirés de ces modèles pour créer des nouveaux modèles 3D adaptés à leurs besoins pour représenter les données géographiques.

### 2.1. La 3<sup>ème</sup> Dimension

Le terme dimension a plusieurs significations, qui sont parfois mal utilisées et confondues. Dans le contexte géographique, les dimensions de sous-espace occupé par l'objet donnent le nombre de dimensions propre à l'objet, comme point (0D), ligne (1D), polygone (2D) et Volume (3D) les coordonnées d'un objet dans un espace donné permettent de définir sa position et sa forme. **Lachance** utilise la notion de l'univers pour repérer les objets existants selon trois niveaux différents : l'univers global qui donne la position absolue de tous les objets sur la terre en 3D, l'univers du système qui donne la localisation des objets qui sont dans le même système comme le système de projection dans une base de données spatiale, et l'univers local, qui permet de donner une localisation régionale[5]. Le nombre de dimension affecté au deux derniers univers peut avoir plusieurs configurations possibles : *univers 1D, univers 2D, univers 3D, univers 4D,..... univers nD*. On note que si le nombre *n* est supérieur à 3, il ne s'agit pas de dimension de type géométriques par exemple le 4D représente l'aspect temporel. [40]

Les formes 3D sont définis par la compagnie ESRI (Environmental Systems Research Institute) comme étant des points, des lignes, des polygones qui, en plus des dimensions X et Y, possèdent une coordonnée Z [34], cette définition est conforme avec la notion précédente de dimension, qui est relative à l'espace, et non à l'objet. Donc il considère les objets en 3D s'ils sont repérés dans un espace 3D. Selon [40], cette définition englobe toutes les possibilités de présence de

la 3D, contrairement à d'autres définitions restreintes trouvés dans [10], [5] et [7], qui ne prennent en considération que les dimensions internes propre à l'objet et non pas celles de l'univers dans lequel les objets sont positionnés.

## 2.2.Modélisation de la 3D et applications

La représentation numérique de la 3D, a commencé par des fins de visualisation tridimensionnelle, par l'utilisation de la stéréovision, qui sert à la production des images, qui donne l'impression 3D lors de leur perception par l'être humain. La vision tridimensionnelle a connu un grand essor, avec ses applications dans des domaines artistiques et de loisirs comme le cinéma et les jeux vidéo. Par la suite, et dans ces dernières années, la 3D a touché divers domaines. Certaines applications ont vu le jour, pour satisfaire les différents besoins, tels que les domaines de CAD et DAO, ainsi que le domaine de l'information géographique dans lequel les besoins en 3D sont beaucoup plus orientés vers l'analyse. Ce qui rend la modélisation 3D dans les SIG plus compliqué, et pose encore des problèmes, à l'inverse des autres domaines (jeu vidéo, cinéma, etc..) dont les besoins sont limités souvent uniquement à la visualisation.

### 2.2.1 L'information géographique 3D

Le premier pas vers la 3D dans les bases de données géographiques, à consister à ajouter un une composante z sous forme d'attribut altitude à chacun des points de la base de données en 2D [60], le résultat est appelé 2.5D et non 3D car il subsiste encore de fortes contraintes liées au fait que l'altitude soit fonction de x et de y [19].

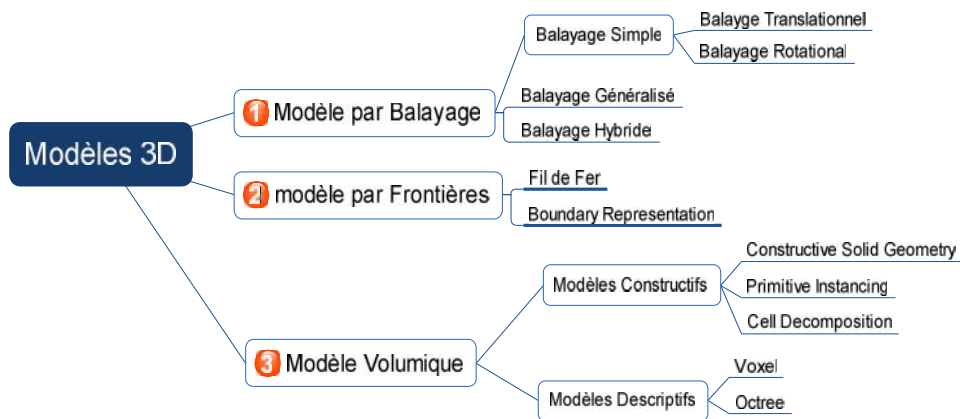
La modélisation de la véritable 3D de l'information géographique, n'est pas limitée à l'ajout de la coordonnée Z, ni à l'affichage de l'aspect 3D. La visualisation tridimensionnelle n'est pas suffisante dans les SIG qui doivent fournir d'autres fonctionnalités pour effectuer l'analyse directement en exploitant les données fournies dans le modèle 3D.

Cependant, il existe certains modèles 3D de base, qui sont développés pour diverses disciplines, dont chacune a ses avantages et ses limites.

## 2.3.Modèles spatiaux 3D

Les modèles spatiaux 3D sont classés dans des catégories, selon différentes perceptions de la 3d. Koussa [39] a distingué 5 familles classées selon les primitives de base utilisées, pour représenter la géométrie des objets du monde réel en : Modélisation filaire, modélisation surfacique, modélisation volumique, modélisation hybride (surfacique et volumique) et modélisation paramétrique.

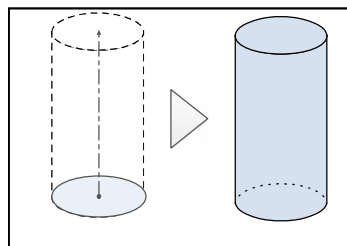
La figure 2.1 représente une classification des modèles 3D, en 3 catégories principales : les modèles par balayage, les modèles par frontières et les modèles volumiques. Les modèles sont représentés par les feuilles et les catégories sont représentées par les nœuds de l'arbre [19], [40].



**Figure 2.1.** Classification des modèles 3D

#### 2.4.Modèle par balayage

Ce modèle de représentation consiste à créer un objet 3D (Volume) de forme régulière à partir d'une surface 2D, par son déplacement le long d'une trajectoire. Par exemple pour le déplacement d'un cercle selon une trajectoire de forme ligne droite permet de créer un cylindre (figure2.2).



**Figure 2.2.** Représentation d'un cylindre par le modèle de Balayage

Ce type de balayage est simple. La trajectoire est sous forme linéaire, pour le balayage translationnel ou autour d'un axe pour le balayage rotationnel.

**Le balayage simple** de type rotationnel est très utile pour représenter les pièces dans la mécanique dont plusieurs composants sont usinés au moyen de tours [19], [54].

**Le balayage Hybride** consiste à combiner plusieurs objets volumiques créés par le balayage simple (translationnel et/ou rotationnel) à partir d'une ou plusieurs surfaces 2D, avec l'utilisation de l'opération ensembliste d'union, afin de créer des objets plus complexes.

Dans notre contexte géographique, et particulièrement dans les zones urbaines, le modèle par balayage est utilisé pour représenter quelques objets qui ont une forme simple et régulière.

**Le Balayage Généralisé**, représente une autre extension du modèle balayage pour représenter des objets plus complexes comme une pyramide par exemple, par l'utilisation d'une surface 2D qui varie au long du balayage.

L'utilisation du modèle par balayage pour représenter des entités géographiques de façon générale est limitée, car les objets du monde réel ont des formes irrégulières et très variables. Cependant il est utile et simple pour modéliser des objets géographiques, de forme simple et régulière parallélépipédique. Ce modèle est utilisé dans certains outils SIG de visualisation (*ArcView*, *ARCGis* de *Esri*) des bâtiments en 3D à partir de leurs représentations surfaciques et un attribut qui représente la hauteur de chaque bâtiment. C'est une représentation 2.5D, elle n'est pas qualifiée comme vraie 3D.

## 2.5.Modèle Volumique

Les modèles volumiques reposent sur la représentation des objets à partir des primitives de base de forme volume. Il existe deux sous-catégories des modèles volumiques qui sont les modèles volumiques constructifs et les modèles volumiques descriptifs.

**Les modèles constructifs** utilisent des primitives qui peuvent être plus complexes pour représenter des objets. Il existe plusieurs types dans cette catégorie :

### 2.5.1 Modèle « Constructive Solid Geometry » (CSG)

Le modèle CSG signifie en français : «Géométrie de Solide Constructif», consiste à définir un ensemble d'opérations, et un ensemble de primitives de base de type solide. La représentation d'un objet réel repose sur la construction de sa géométrie à partir de l'application de certaines opérations sur certaines primitives de base de l'ensemble des solides. Donc la représentation de l'objet selon ce modèle est défini par une composition hiérarchique en un ensemble de solides dont les nœuds de la hiérarchie représentent l'ensemble des opérations nécessaires pour combiner les solides de niveau inférieur dans la hiérarchie. En plus des opérations booléennes, il est utile de définir des opérations unaires de transformations : {rotation, translation, changement d'échelle}.

Donc un modèle CSG est un ensemble de primitives de base, (eg : *Solides = {Cube, Cône, Cylindre}*), et un ensemble d'opération booléenne  $Op = \{Union U, Intersection \ , Différence \}$ .

La figure 2.3 est un exemple de représentation par un arbre d'un Cube avec un trou cylindrique obtenu à partir de la différence entre une primitive de type Cube et une autre de type Cylindre.

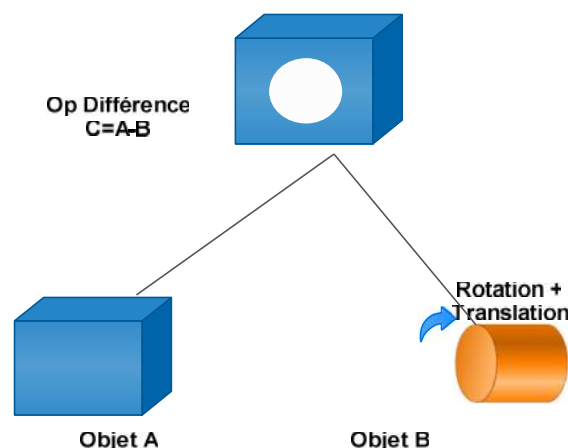


Figure 2.3.Arbre de représentation CSG d'un objet

L'application de certaines opérations sur certains solides donne parfois des objets non volumiques. Cette situation n'est pas appropriée dans un modèle volumique. La solution réside dans la régularisation des opérations utilisées afin qu'elles ne produisent que des objets volumiques ou rien.

Les modèles CSG sont basés sur la définition de la géométrie des primitives. Celle-ci est définie mathématiquement de façon concise. Donc ce modèle offre une représentation précise des objets. Cette précision facilite le calcul des opérations géométriques. Dans la figure 2.3, le volume de l'objet C est égale à la différence entre les deux volumes des deux primitives A et B.

Ce modèle ne préserve pas l'unicité des représentations pour tous les objets. Certains objets peuvent avoir plusieurs représentations différentes [54]. La modélisation avec un clonage à partir d'un nombre limité de primitives de forme régulière, n'est pas adéquate avec le caractère complexe et irrégulier des objets réels. Car le nombre des objets complexes possibles à représenter est limité par l'ensemble des solides prédéfinis dans le modèle. En plus, les objets qui sont plus complexes et représentables dans ce modèle nécessitent un grand nombre d'opérations, ce qui influe sur le temps de traitement.

Le modèle CSG est largement utilisée dans les domaines de conception 3D, comme les systèmes CAO. Il sert à créer des nouveaux objets, dont les formes sont simples et régulières et leur complexité est abordable (dans la mécanique).

Mais l'utilisation de ce modèle dans le domaine géographique, pose plusieurs problèmes. Car les objets, pour certains, sont complexes et sont caractérisés par l'irrégularité, déjà existante dans la réalité. Ils nécessitent une décomposition, selon l'ensemble des opérations et des primitives du modèle. Alors, la méthode employée n'est pas vraiment naturelle et on imagine mal des restituteurs ou bien des méthodes automatiques permettant les décompositions d'objets (en primitives et opérations) [19].

Le modèle CSG ne prend pas en compte l'aspect topologique des objets. Malgré cela, il existe des modèles géographiques qui utilisent ce modèle en combinaison avec d'autres modèles pour ajouter plus de détails lors de la visualisation [39].

### 2.5.2 Primitive Instancing

Le modèle **Primitive Instancing** signifie l'instanciation des primitives. Il est appelé aussi **modèle paramétrique** [39], qui consiste à décrire les objets avec des primitives et un ensemble de paramètres commun à un ensemble d'objets de même type. Par exemple un escalier qui est une construction constitué de marches peut être paramétré avec le nombre de ses marches, ainsi que la hauteur et le pas de marche. Un engrenage est paramétré par le nombre de dents et son rayon. Dans ce cas, les objets doivent être classés selon leurs formes semblables en primitives complexes et paramétrables.

La combinaison entre les primitives n'est pas possible. Chaque objet a sa propre structure interne. Cette méthode de définitions des objets à partir des primitives complexes n'est pas adéquate pour représenter l'aspect spatial des objets géographiques. Il est important de décrire les objets par des primitives de base (les plus simples) afin de réduire l'espace de stockage et permettre de définir explicitement le voisinage d'un objet.

Mais, au niveau sémantique, ce modèle présente un avantage, et permet de définir des paramètres de type quantitatifs et/ou qualitatifs. Afin de compléter la représentation des objets par leur aspect thématique en plus de l'aspect géométrique, il est possible d'utiliser des primitives complexes paramétrées de types thématiques pour regrouper des objets semblables dans le domaine des SIG.

## 2.6. Cell Decomposition

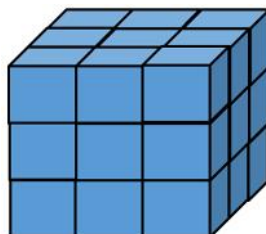
En français décomposition en cellules, ce modèle représente un modèle CSG allégé, par l'ajout d'une certaines primitives complémentaire comme le quart de cône, le quart de cylindre..., afin de réduire le nombre d'opérations permises dans ce modèle en une seule opération, qui est l'union pour le collage des primitives ou cellules qui compose l'objet.

Ce modèle présente un avantage par rapport à son prédécesseur CSG, en termes de rapidité d'affichage, et simplicité de représenter les objets avec une opération simple de collage sur les primitives de base, ce qui permet de classier cette extension de CSG, avec **les modèles volumiques descriptifs** ou **représentation par spatial partition**.

Les modèles volumiques descriptifs, permettent de décrire un objet volumique par son sous-espace occupé par l'objet, à partir de l'union d'un ensemble de volumes élémentaires, qui sont contigus mais sans chevauchement. Dans ces modèles, les objets sont représentés par des vrais volumes.

### 2.6.1 Modélisation par Voxel

La modélisation par voxel, ou l'énumération spatiale est un modèle volumique descriptif qui décrit les objets volumiques avec un ensemble de volume élémentaire régulier, ou voxel qui signifie « volume element » par analogie de « pixel » en 2D.



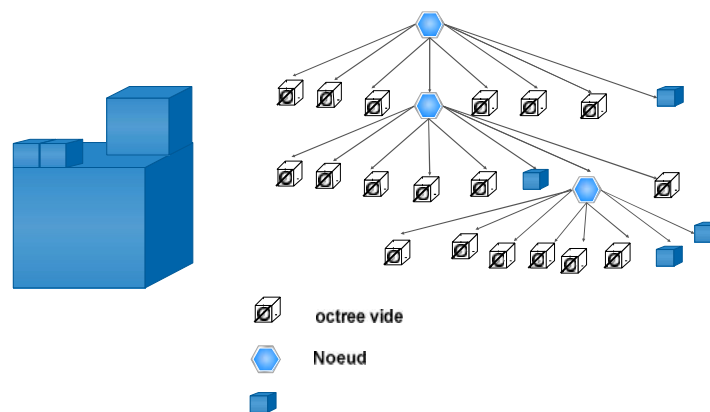
**Figure 2.4.** Ensemble de voxels

La figure 2.4 représente un ensemble de voxels qui composent un cube. Ce modèle est une extension 3D du modèle raster 2D. Donc ce modèle présente une simplicité de modélisation des objets 3D avec des formes régulières unitaires (cube) qui permet de mieux présenter la réalité d'un volume.

Mais l'exactitude de représentation est liée à la résolution, et les limites des objets ne sont pas réalistes. Le problème majeur de cette représentation réside dans le volume de stockage énorme, ce qui nécessite une méthode de compression pour réduire l'espace de stockage.

### 2.6.2 Modélisation par Octree

C'est une méthode de réduction de volume, du modèle précédent par l'utilisation d'une primitive volumique de taille variable qui est l'octree. A l'aide d'une structure d'arbre l'espace est partitionné en voxels de taille différente. Chaque niveau de la hiérarchie représente une partition en voxels de taille différente, résultante de la division du nœud de niveau supérieur en 8 voxels fils. Chaque voxel prend une des trois valeurs possible, soit il représente une feuille de l'arbre (donc il représente un espace plein et appartient à l'objet), soit il prend la valeur zéro (donc il représente un espace vide), ou encore il représente un nœud qui est divisé en 8 voxels fils. Donc la création de l'arbre des octrees est basée sur une procédure récursive, à chaque itération les voxels qui représentent les nœuds sont divisés et produisent des voxels de taille inférieure dans le niveau bas de l'arbre.



**Figure 2.5.** Présentation d'un objet en octree et son arbre hiérarchique

La figure 2.5 illustre la représentation d'un objet en octree ainsi que son arbre syntaxique, dont le tronc de l'arbre représente l'espace globale, une feuille représente un octree plein ou vide, et un nœud représente un espace à diviser en voxels plus petits dans le niveau inférieur.

L'avantage de la modélisation par octree réside dans la réduction de volume de stockage nécessaire pour représenter un objet par voxel. Il est considéré comme une méthode de compression du modèle par voxel. En effet, l'objet illustré dans la figure 2.5 est représenté seulement par 4 primitives (cube) en modélisation par octree, en comparaison avec la modélisation par voxel qui représente le même objet avec 74 voxels.

Le modèle par voxel et particulièrement son extension octree est utilisé dans l'imagerie 3D, ainsi que dans les SIG pour le mode raster en 3D [43], [44]. Mais il est limité à la visualisation 3D et la modélisation des solides [2].

D'une manière générale, les modèles volumiques reposent sur la définition des primitives de 3D (volumique), afin de représenter les objets. Mais cette philosophie de modélisation n'est pas appropriée dans le contexte de la représentation des objets dans le monde réel (géographiquement). Car dans le domaine géographique, il est intéressant de montrer explicitement comment les objets partagent leurs espaces entre eux et afin d'éviter la redondance de représentation et d'optimiser les opérations d'analyse.

## 2.7.Modèles par frontières

Ces modèles consistent à représenter l'objet non pas par son volume mais par son contour. Le contour d'un objet n'est plus décrit par des primitives volumiques, il s'agit des éléments de dimensions inférieures comme les surfaces et les lignes.

Il existe deux types de modèles par frontières qui sont le modèle filaire (fil-de-fer) et le modèle Boundary représentation (B-rep).

### 2.7.1 Modèle filaire (fil-de-fer)

Le modèle fil-de-fer est considéré comme le premier modèle vecteur en 3D. Il consiste à définir le contour d'un objet seulement par des segments de lignes qui sont délimités par des nœuds d'intersections. Ainsi un objet est représenté seulement par des points et des lignes. C'est la représentation la plus simple, qui permet de qualifier ce modèle comme le modèle le plus simple à implémenter, et qui utilise l'espace de stockage réduit. Mais le modèle filaire ne représente pas explicitement les surfaces. L'utilisateur doit interpréter visuellement les surfaces à partir de la représentation linéaire des objets.

La figure 2.6, montre par un exemple de représentation d'un cube. L'ambiguïté du modèle fil-de-fer est illustrée, au niveau visuel, dans l'affichage des arcs, sans présence des surfaces qui cachent le contour interne de l'objet. Ainsi celui qui regarde tombe dans une confusion et ne peut pas distinguer entre les différents côtés de l'objet. L'exemple montre deux interprétations visuelles différentes possibles [12], en plus plusieurs interprétations sont possibles aux niveaux des données [19].

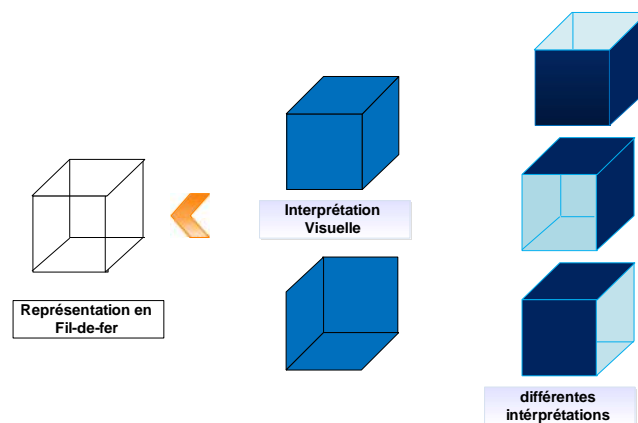


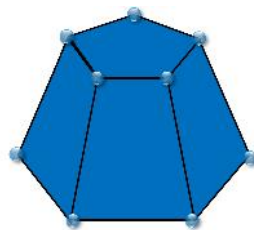
Figure 2.6. Présentation ambigu par le modèle fil-de-fer

Pour enlever cette ambiguïté, il est important d'étendre ce modèle par l'ajout des surfaces pour représenter les contours des objets.

### 2.7.2 Modèle Boundary représentation (B-Rep)

B-rep est une extension du modèle précédent, qui permet d'enlever l'ambiguïté qui existe dans la présentation par le modèle filaire. Ce modèle consiste en l'introduction de la surface comme un élément constituant du contour d'un objet modélisé par ses frontières. Alors un objet est modélisé par les surfaces constituant son contour. Celles-ci sont présentées chacune par les lignes du contour. Ces derniers sont limités chacune par deux points d'extrémités. La figure 2.7 représente un exemple de représentation par le modèle Boundary représentation.

Le présent modèle dans sa forme simple représente les objets par leurs frontières qui se composent à partir des faces, sans considération de la notion de volume, les faces doivent être planes car elles sont modélisées par des segments de ligne droite, ce qui nécessite un grand nombre de surfaces pour modéliser approximativement les surfaces non planes.



**Figure 2.7.** Modélisation B-rep

Afin de représenter les surfaces non planes à l'affichage, quelques modèles de ce genre n'autorisent que les faces planaires de forme triangulaire, ce qui nécessite une procédure automatique de triangulation des surfaces.

Malgré ces lacunes, ce modèle est une extension logique en 3D du modèle géographique vectoriel en 2D, dont les objets sont représentés de la même façon avec leurs primitives de base de dimension inférieure. Donc ce modèle peut être élaboré afin de représenter les données géographiques en vecteur 3D.

Cependant, il est indispensable d'ajouter la notion de volume, avec l'orientation des surfaces, afin de déterminer l'espace interne des volumes et ses faces externes, en plus de l'association de la topologie en 3D qui est absente dans sa version simple.

### Conclusion

Dans ce chapitre nous avons présenté la notion de dimension afin de définir la 3D dans le contexte des SIG. La 3D a connu un grand succès dans plusieurs domaines de grand public, comme le cinéma et les jeux vidéo, ainsi que d'autres domaines plus spécifiques dont les systèmes CAO qui jouent

un rôle important pour la modélisation des nouveaux produits en 3D dans plusieurs activités professionnels comme la mécanique et l'architecture.

Les besoins en 3D dans ces domaines sont reflétés dans les modèles 3D adéquats qui sont présentés dans ce chapitre, avec leurs avantages et leurs inconvénients, en plus de leurs philosophie de représenter les formes des objets en comparaison avec celle des SIG.

Le modèle par balayage présente une simplicité en termes de passage à la 3D ou plus précisément 2.5D. Les modèles volumiques descriptifs sont considérés comme une extension de la 3D des modèles Raster 2D dans le domaine géographique. Le modèle B-rep est considéré comme une extension 3D naturel du modèle vectoriel 2D, mais n'ayant pas la modélisation de la topologie 3D, l'analyse 3D devient extrêmement compliquée. Ainsi la majorité des modèles 3D présentés dans ce chapitre sont des modèles géométriques. Ils ne sont pas destinés à la représentation des données géographiques.

Les modèles 3D dans les SIG doivent prendre en considération l'aspect topologique qui décrit les relations entre les objets. Le stockage de cet aspect des objets réels de façon explicite est très important pour améliorer les opérations d'analyse spatiale.

Nous abordons la modélisation 3D des données géographiques dans le prochain chapitre.

## Chapitre 3 : Etudes des modèles géographiques en 2.x-D et 3D

### Introduction

Les SIG sont utilisés dans plusieurs domaines utilisant l'information spatiale. Ils ont offert plusieurs fonctionnalités et divers outils pour répondre aux différents besoins en information géographique pertinente à la prise de décision dans différentes disciplines. Mais ils reposent sur une modélisation limitée en 2D alors que la réalité est en 3D. Cette restriction à la 2D est dictée en quelques sortes par les limites en technologie de stockage et de traitement informatique à l'époque de l'apparition des premiers SIG et des supports d'affichages (papier ou écran).

Cependant, de nos jours après la généralisation des SIG en 2D, des nouveaux besoins en informations géographiques 3D sont apparus actuellement qui nécessitent des modèles plus proches de la réalité, dans divers domaines d'applications. Par exemple les applications qui modélisent des phénomènes spatiaux qui se diffusent en 3D comme le bruit [36], l'inondation d'eau et la pollution d'air [61] et l'environnement urbain...etc. [58]. Ces besoins de la modélisation 3D sont accentués par l'apparition des nouvelles techniques d'acquisitions (Lidar, etc...).

Aussi le développement dans le domaine informatique, avec l'apparition des ordinateurs de plus en plus puissants au niveau du stockage et de traitement des processeurs ainsi qu'au niveau des moyens d'affichage. En outre, l'apparition des modèles et logiciels en 3D dans l'infographie et les systèmes DAO et CAO ont facilité l'utilisation de la 3D.

Afin de répondre à ces besoins, les nouveaux modèles des SIG en 3D doivent avoir, en plus de certaines fonctionnalités, les mêmes fonctions que les SIG 2D [2]. Donc ils doivent fournir des fonctionnalités d'acquisition, de représentation de stockage, ainsi que d'analyse en plus de l'affichage en 3D des données géographiques. Par rapport à l'infographie le rôle de la visualisation n'est que secondaires vis-à-vis des fonctions de traitement et d'analyse qui représentent le cœur des SIG. Alors les modèles 3D doivent être aptes à représenter les objets selon les différents aspects géométriques quantitatifs et qualitatifs ainsi que l'aspect sémantique. Dans ce dernier réside l'intérêt de l'objet au niveau de l'utilisateur.

Suite aux différentes spécificités des SIG, d'autres modèles en 3D sont apparus pour modéliser les différents aspects des objets réels. Chacun, des modèles, a proposé une façon de stocker et représenter les données géographiques dans un contexte large ou restreint à un domaine spécifié d'application des SIG.

L'objet du présent chapitre est de présenter certains travaux dans ce contexte de modélisation des données géographiques dans l'espace 3D.

La représentation de la 3D dans le domaine géographique a commencé avec la représentation du relief, par des courbes de niveaux, dans les cartes topographiques dessinées sur papier. Après

l'apparition des SIG, et l'utilisation des bases de données pour stocker les données géographiques, le premier passage à la 3D a commencé par l'ajout de la coordonnée Z comme attribut où à chaque point (X, Y) correspond un seul attribut Z. Cette technique est utilisée afin de représenter les différentes élévations du relief dans les modèles numériques du terrain (MNT). Les modèles qui sont basés sur cette technique ne sont pas des modèles vraiment en 3D, ils ne sont qu'en 2.5D.

### 3.1 Modèles 2.5-D

Le modèle 2.5D permet de représenter la 3D dans un modèle 2D dont les données spatiales sont stockées dans la base de données avec seulement les deux coordonnées (X, Y) et l'attribut z. Ainsi, le modèle 2.5-D consiste en la visualisation tridimensionnelle de chaque position selon une valeur d'attribut qui représente généralement la hauteur, ou l'altitude par rapport à la référence. Cette technique peut servir à afficher une valeur non géométrique pour représenter en 3D un attribut sémantique en illustrant leur distribution au long d'une étendue géographique donnée.

Le modèle 2.5-D autorise au maximum une valeur Z à chaque position (x, y). Donc la représentation des murs verticaux, d'objets artificiels ou naturels comme bâtiments ou les rocs n'est pas faisable [30].

Quelques logiciels SIG commerciaux utilisent le modèle 2.5-D en combinaison avec le modèle de balayage pour la représentation des bâtiments, selon leur hauteur afin d'afficher ces derniers comme des blocs en 3D, mais les toits ne sont pas modélisés.

La représentation 2.5-D est utilisée principalement pour modéliser le relief de terrain. Deux modèles sont utilisés, le modèle *TIN* (*Triangular irregular Network*) et le modèle *Raster DEM* (*Digital Elevation Model*) ou la structure GRID.

#### 3.1.1. Modèle TIN

Le modèle TIN est utilisé pour la représentation du terrain à partir des primitives surfaciques sous forme de triangles irréguliers. L'ensemble des triangles sont interconnectés sans intersections. La création d'un TIN nécessite une méthode de triangulation où chaque triangle définit un plan différent, par exemple la méthode de triangulation de Delaunay [20] garantit que le cercle circonscrit par un triangle ne contient aucun point d'un autre triangle.

La différence principale entre un Raster DEM et une structure TIN réside dans la distribution de points. Dans un raster les points DEM sont arrangés régulièrement, alors que dans TIN, un algorithme détermine les points nécessaires pour la représentation du terrain [57]. Donc le volume de stockage des données est plus petit dans le cas de TIN que le Raster DEM.

### 3.1.2. Modèle 2.75-D

Larue dans [41] a introduit un modèle qui s'appelle 2.75D afin de résoudre le problème lié à l'insuffisance du modèle 2.5D pour représenter correctement les objets géographiques avec une épaisseur. Le 2.75D autorise le stockage de deux valeurs  $z$  pour une position  $(x, y)$ . Avec les deux valeurs minimale et maximale d'altitude on peut avoir l'épaisseur des objets.

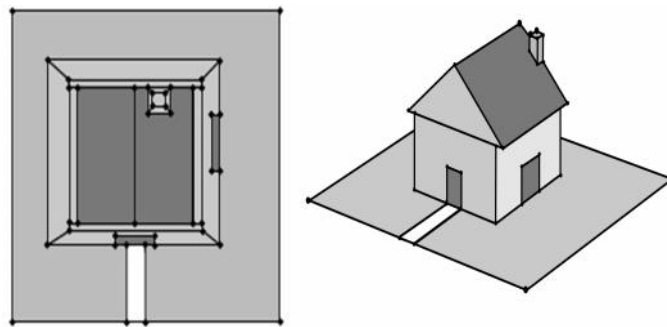
Mais l'inconvénient de ce modèle reste le même que celui du 2.5D. Deux altitudes seulement en  $z$  sont attribuées au même point qui appartient à une surface au niveau du terrain. Les deux altitudes ne donnent que l'aspect visuel de la 3D mais aucune relation topologique selon l'axe  $z$  n'est représentée.

### 3.1.3. Modèle 2.8-D

Gröger et Plümer dans [30] proposent une modélisation (2.8-D map) qui est une extension légère des modèles 2D ou 2.5-D qui couvre beaucoup d'opérations en 3D qui préserve la simplicité algorithmique et conceptuelle du 2D car il existe une représentation plane 2D équivalente qui a la même topologie. Ce modèle néglige les détails internes (par exemple un mur qui sépare un garage d'une maison) et ne considère que l'enveloppe externe d'un bâti.

Quant au modèle 2.8D, Il fournit une visualisation tridimensionnelle avec une topologie bidimensionnelle. Un objet modélisé par ce modèle est considéré comme un objet 2D drapé dans un espace 3D (tissus tendu sur un espace 3D sans trou ni pli). Par exemple un arc ne peut être relié qu'à deux face seulement, mais en 3D réel il peut être relié à plusieurs faces.

La figure 3.1 illustre un exemple d'une visualisation d'une représentation 2.8-D avec son équivalent topologique planaire en 2D [30].



**Figure 3.1.** Exemple de 2.8D avec son équivalent topologique en 2D [31]

La modélisation 2.5D est une représentation intermédiaire entre la 2D et la 3D [39]. Les modèles 2.x-d (2.5D, 2.75D, 2.8D) ont hérité leur simplicité des modèles 2D. Ils ont l'avantage d'utiliser les mêmes moyens de stockage et de gestion des modèles 2D sans modification importante. Mais, ils sont pauvres en termes de représenter l'aspect qualitatif et/ou quantitatif de la géométrie selon l'axe verticale  $Z$ . Ces modèles représentent une continuité des modèles 2D, mais ils ne sont pas aptes à représenter le caractère volumique des objets géographiques.

Ces limites des modèles 2.x-d ont poussé la communauté des chercheurs des domaines concernés par l'information géographique en 3D, à s'orienter vers la conception des nouveaux modèles dans le but de modéliser les objets avec le type volume et de définir les relations topologiques en 3D. Certains travaux se sont inspirés des modèles géométriques utilisés dans les logiciels CAO (voir le chapitre précédent) avec des adaptations selon les spécifications géographiques et les besoins de leurs applications.

### 3.2 Modèles 3D

La modélisation géographique 3D a connu un grand manque, suite aux obstacles qui ont rencontré cet axe de recherche. Ces divers problèmes, dus aux différentes sources, impliquent l'apparition de plusieurs modèles 3D, dont certains modèles sont nés afin de modéliser quelques aspects des objets réels, de répondre à certains besoins, et de résoudre certains problèmes liés à un contexte d'application particulier. Tandis que d'autres travaux ont essayé de trouver une solution universelle avec des modèles de base qui sont devenus des standards comme le modèle CityGML.

#### 3.2.1. 3D FDS (Formal Data Structure)

Le modèle 3D FDS, ou structure formelle de données [39], est proposé par *M. Molenaar* [46], l'un des pionniers dans cet axe de recherche. Le 3D FDS est un modèle topologique vectoriel simple parmi les premiers modèles en 3D. Dans ce modèle, la topologie est associée à la géométrie par des liens entre les primitives qui représentent un objet volumique de type Solide en plus des objets Point, Ligne et Surface. Un solide est construit à partir des primitives de type face. Des relations de construction et d'inclusion sont utilisées pour relier les primitives entre elles. Le 3D FDS, dans ses conventions, a fixé pas mal de contraintes pour structurer les données (par exemple l'exigence sur les surfaces qui doivent être planes).

Ce modèle est considéré comme une généralisation du modèle topologique 2D et particulièrement la version 2D du modèle FDS proposé dans [45]. Malgré ses limites, ce modèle a des capacités dans la description thématique et géométrique 3D par objets [63]. C'est une approche du vecteur pour construire la géométrie, exprimée principalement par les relations entre les frontières [48], ce qui ressemble au modèle B-rep. Il représente un modèle de base pour les modèles topologiques. Le Standard ISO prend son inspiration à partir de ce modèle mais recommande une indépendance totale entre le modèle topologique et le modèle géométrique [38].

La figure.3.2 représente le modèle conceptuel de 3D FDS, qui repose sur le niveau objet par la définition de trois types d'objets : Point, Ligne, et Solide, ainsi que le niveau primitif qui représente les 3 primitives de base. Il a introduit une nouvelle primitive Bordure qui représente l'association entre Face et Arc. Ce dernier est relié à plusieurs faces, ce qui est indésirable pour représenter la topologie par une relation de cardinalité (N, N).

**FLick** dans [27] étend le 3D FDS en introduisant le Solide comme primitive topologique [31].

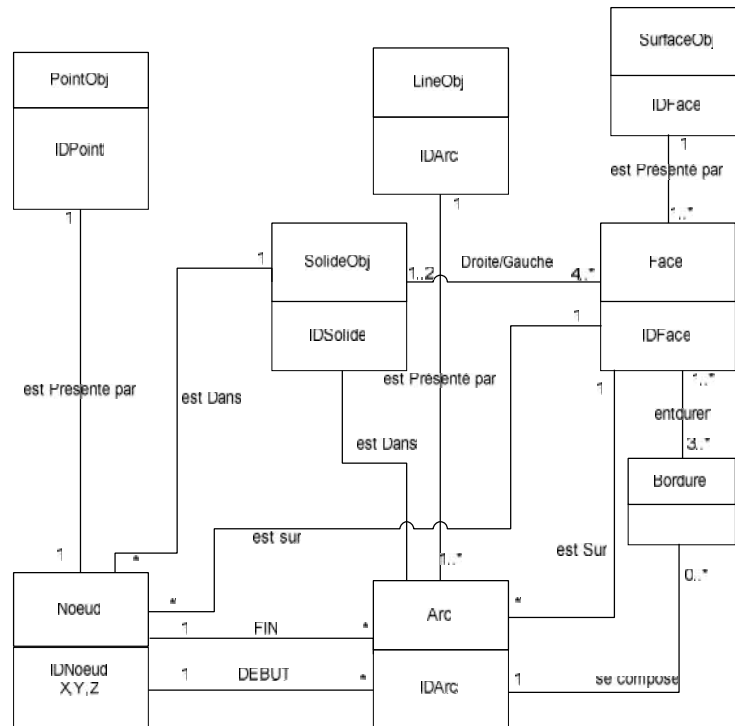


Figure 3.2. Conception du Modèle 3D FDS, [46] cité dans [40]

### 3.2.2. Modèle multidimensionnel MD (2.5D, 2D, 3D)

Afin de résoudre le problème de liaison selon l'axe z vu dans les modèles 2.5D, de **Cambrai** [17] dans son modèle a introduit une relation qualitative de type « placé sur » ou « placed on », entre le modèle 2.5D de relief et les objets qui ont des altitudes et sont localisés sur le relief, afin d'avoir une perception globale en 3D. Donc le relief est modifié. Les objets 2D et 3D sont placés sur le relief en 2.5D. Il considère ce modèle comme un modèle multidimensionnel MD (2.5D, 2D et 3D), mais les trois éléments sont définis de façon indépendante et dans des espaces différents. Donc la relation qualitative utilisée ne présente pas le caractère continu des relations topologiques entre ces éléments superposés comme des couches sans décrire les parties géométriques communes entre eux.

### 3.2.3. Modèle 3DGT

**Zeitouni** dans [62] a distingué entre l'adjacence verticale et horizontale, ceci conduit à son modèle topologique pragmatique qui décrit les connexions entre objets indépendamment de leurs natures. Le modèle proposé s'appelle **3DGT**, il tient compte des liens verticaux et horizontaux, comme la relation « *placed on : placé sur* » présentée dans le modèle précédemment (le modèle **MD**), et les relations « *beside : à côté de* », « *in : dans* » utilisées dans le modèle **BDE** proposé dans [21] et [22].

Le modèle **3DGT** représente des objets de nature hétérogène en intégrant des points, lignes, surfaces (comme dans 2D GIS), et volumes. Il prend en considération le relief, comme dans le modèle MD. La topologie est utilisée comme un moyen de plus haut niveau d'abstraction sémantique que la

géométrie et un filtre pour éviter le stockage en mémoire des éléments géométriques non nécessaires. Ainsi, un mur, par exemple, est considéré comme une surface de séparation mais il est décrit par un volume géométrique de même pour un tronçon de route qui est considéré au niveau topologique comme un arc mais au niveau de la géométrie il s'agit d'une surface. Donc la description géométrique peut être liée aux objets au lieu des primitives. Ce modèle nécessite un ensemble de contraintes tels que la contrainte pour un arc horizontal qui est relié à une seule face au maximum dans chaque côté (*droite, gauche, au-dessus, en-dessous*).

La figure 3.3 [62] montre la conception du modèle au niveau des primitives et les relations topologiques qui existent entre elles. Ces primitives sont liées au niveau objet pour la création des différents objets géographiques simples et complexe, de façon similaire au modèles topologique 2D et particulièrement le standard français *EDIGéo* (voir section 1.9.3, figure 1.9). Ce modèle est très riche en informations topologiques mais il ne présente aucune explication sur l'implémentation du modèle qui nécessite un processus de décomposition et d'intégration des données selon la structure proposée de façon à garantir la consistance. Les formes des objets réels ne sont pas toutes compatibles avec cette structure. En outre, la séparation entre la topologie et la géométrie est utilisée ce qui nécessite une méthode mathématique pour calculer cette topologie à partir de la géométrie acquises en 3D directement, ou construire la topologie 3D à partir du 2D.

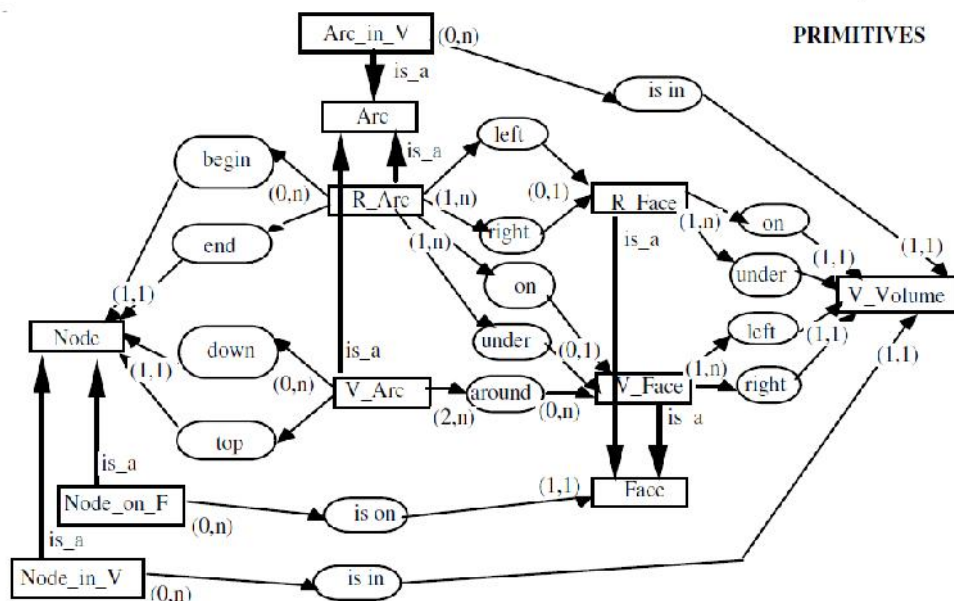


Figure 3.3. Primitives topologique du modèle 3DGT [62]

### 3.2.4. Modèle TEN (Tetrahedral Network) ou 3D TIN

Le TEN acronyme de 'Tetrahedral Network' il s'agit d'une extension du modèle TIN [2]. Ce modèle est proposé par Pilouk [50]. Il utilise 4 primitives de base : tétraèdre, arc, triangle et nœud. Un solide est composé de tétraèdres, une surface de triangle, une ligne d'arc, et un point de nœud [64]. Le

modèle représente les objets de forme complexe qui ne sont pas identifiés par le modèle 3D FDS. Les formes volumiques complexes sont décomposées en un ensemble de Tétraèdres sans intersection entre eux. Cette décomposition en volume convient aux données géologiques, mais ne convient pas pour les objets artificiels tels que les objets du milieu urbain où l'intérieur des objets (Bâtiment par exemple) ou un espace ouvert doit être décomposé en volumes tétraédriques [39].

Le modèle TEN, présente beaucoup d'avantages au niveau géométrique et topologique. Il a été considéré par les chercheurs en sciences de la terre comme une structure de données utile pour quelque temps [55], mais il exige un grand espace de rangement et impose un découpage indésirable pour des objets réels de régions urbaines.

Beaucoup de chercheurs ont considéré ce modèle comme un cas particulier des modèles basés sur le concept de "simplexe" et des "*complexes simpliciaux*", pour modéliser la topologie. Dans ce modèle, chaque primitive se compose d'un ensemble de primitives de dimension inférieure.

Ces concepts de la topologie algébrique sont utilisés dans la modélisation pour décrire l'abstraction des relations entre les différents objets. Chaque n-simplexe est entouré par (n+1) de (n-1)-simplexe différents, dont 0-simplexe représente un point, 1-simplexe représente une ligne et 2-simplexe une surface (triangle), et 3-simplexe représente un solide (tétraèdre). Un ensemble de simplexes contigus sans chevauchement forme un "*complexe simplicial*" ou n-cell qui représente un objet complexe de n dimensions. Un modèle est proposé dans [11] qui s'appelle "le complexe du simplicial" utilise les termes *0-simplex*, *1-simplex*, *2-simplex*, et *3-simplex* pour désigner les objets géographiques de type nœud, ligne, surface, et volume, Son modèle peut être étendu aux n dimensions [1], [65].

### 3.2.5. Modèle SSM (Simplified Spatial Model)

SSM Acronyme de *Simplified spatial model* qui signifie Modèle Spatial Simplifié a été conçu pour servir des applications orientées web [11]. SSM est proposé dans [63]. Ce modèle est simplifié par rapport au modèle 3D FDS car il n'utilise pas l'arc pour la construction des objets. Il est basé sur l'hypothèse que les arcs peuvent être omis de la représentation du modèle sans risque. Cette simplification permet de réduire le Volume de stockage par l'élimination de la relation entre Arc et Face qui est de type (n, n). Celle-ci n'est pas appropriée à représenter la topologie qui nécessite l'ajout d'une table supplémentaire au niveau du modèle logique adéquat.

La figure 3.4 représente le modèle SSM qui utilise seulement deux primitives Nœud et Face et évite la primitive de type arc. Mais au niveau objet, les structures linéaires sont toujours présentes et utilisent 4 éléments qui représentent les objets de types : Point, Ligne, Surface et Solide.

Ce modèle offre la possibilité de faire des requêtes spatiales ainsi que la visualisation 3D des résultats sous web. Les relations topologiques entre les objets de type Volume sont détectées à partir

des relations qui existent entre ses frontières (faces et nœuds). **La Figure 3.5** illustre les différentes relations possibles entre deux volumes. Mais ce modèle n'est pas apte à détecter toutes les relations : par exemple les relations de directions comme (*Above*, *under*) sont calculées à partir des valeurs métriques.

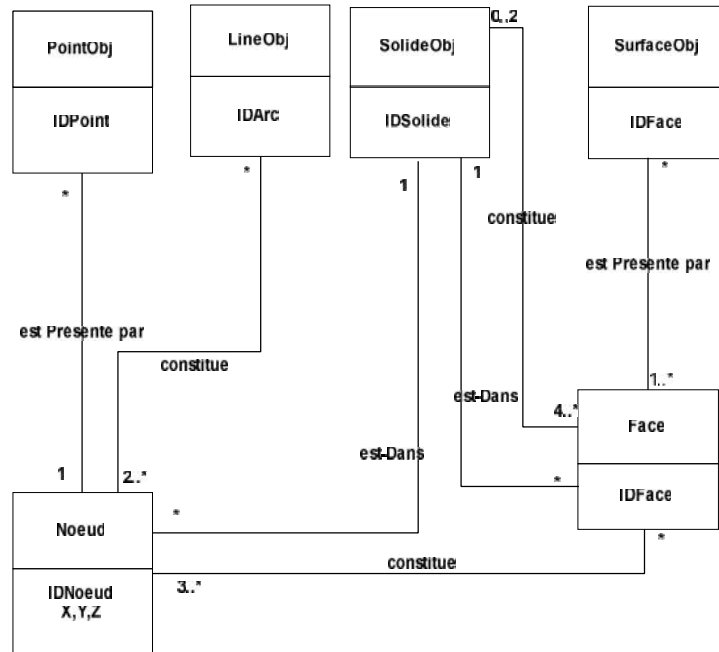


Figure 3.4. Simplified Spatial Model SSM, cité dans [40]

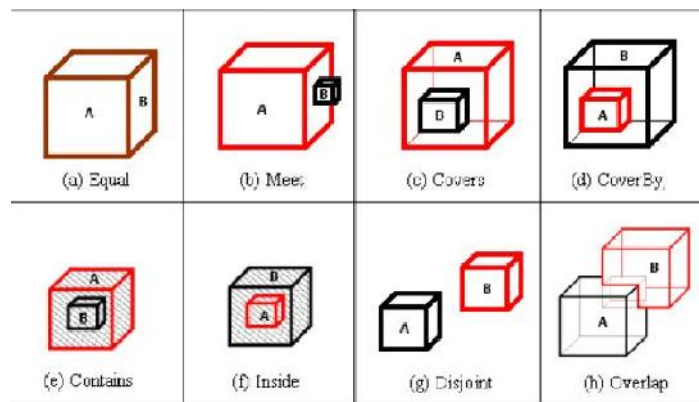


Figure 3.5. Relation Volume-Volume [63]

Le problème de construction et de stockage des relations topologiques en 3D représente un défi qui n'as pas, encore, de solution simple et efficace tel qu'en 2D. Le modèle SSM a été testé pour la visualisation et l'analyse en 3D des données urbaines.

### 3.3 Modèles 3D Urbain

Actuellement l'extension des zones urbaines nécessite une bonne maîtrise et gestion adéquate pour mieux maîtriser l'ensemble des phénomènes liés à l'urbanisme (gestion de la circulation routière, brut, pollution, incendie, etc..). Pour cela, les chercheurs ont mis au point des modèles de gestion dans le

domaine urbain. Dans ce chapitre nous allons présenter certains de ces modèles à desquels nous nous sommes inspirés.

### 3.3.1. UDM (Urban Data Model)

UDM est l'Acronyme de Urban Data Model signifiant Modèle de donné Urbain [39]. Ce modèle est proposé dans [16], Il représente un objet par ses frontières, comme le modèle SSM il repose sur seulement deux primitives Nœud et Face. Un objet volumique est représenté par les faces qui l'entourent. Celles-ci doivent être planes et convexes. Cette contrainte est garantie par la triangulation des faces complexes, car un triangle est toujours convexe. Un objet de type surface est représenté par des faces. Une Face est reliée avec au moins 3 nœuds dont l'ordre des nœuds oriente la face. La primitive de type arc n'est pas représenté explicitement dans ce modèle mais les objets linéaires sont représentés par les Nœuds, avec au moins 2 nœuds. Le modèle UDM est représenté dans La figure 3.6.

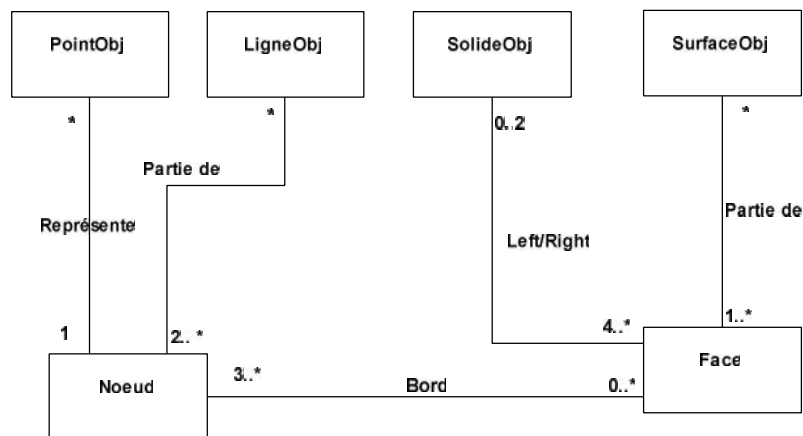


Figure 3.6. Urban Data Model [16]

### 3.3.2. Modèle Standard CityGML

Le Modèle City GML est un standard international à base XML pour la représentation et l'échange des données des modèles de ville virtuelle 3D entre les différentes applications. CityGML est adopté par l'Open Geospatial Consortium comme norme. Il est basé sur les normes ISO 19100 et est implémenté comme un schéma d'application de balisage géographique de l'OGC Géographic Markup lanugage (GML) [49]. CityGML a été développé par the Special Interest Group 3D (SIG 3D). La spécification définit à la fois une modélisation UML et un encodage du modèle au format GML [37].

CityGml prend en compte 4 aspects différents : géométrie, topologie, sémantique et apparence dans les modèles de ville virtuelle (city model). La figure 3.7 représente le modèle géométrique de CityGml avec les primitives utilisées et les classes d'objets complexes construites à partir de ces primitives.

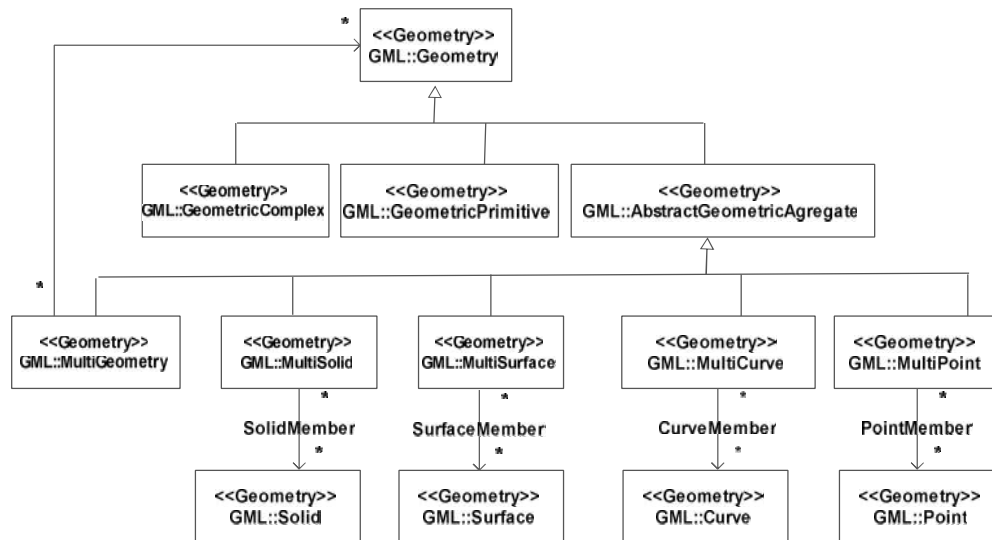


Figure 3.7. Modèle Géométrique de CityGML

En plus de l'aspect géométrique et topologique, *CityGml* offre un modèle sémantique de bâtiments très riches, qui offre 5 niveaux de détails différents (*LoD* ou *Level Of Detail*), dont le niveau *Lod0* représente un *MNT*, *LoD1* représente un bâtiment par un seul block et toit de forme plat. *LoD2* représente les toits avec plus de détails et avec des surfaces thématiquement variés. *LoD3* est un modèle d'architecture qui ajoute les détails comme les Balcon, les ouvertures telles que les fenêtres et les portes. *LoD4* représente les structures internes des bâtiments comme les chambres et les fournitures en deux types mobiles et installations. Le concept de *LoD* définis par *CityGml* est illustré dans la **figure 3.8** [37].

Il existe d'autres modèles d'échanges et formats standards des données spatiales comme *COLLADA* acronyme de (*COLLABORATIVE DESIGN ACTIVITY*), qui utilise des polygones, matériaux et des animations pour modéliser des objets 3D dans un format *XML* [15]. *KML* est l'acronyme de "*Keyhole Markup Language*", qui est aussi un format *XML* de description et d'échange de données spatiales, créé par la société Google et utilisé par des logiciels de cette société, tels que *Google Earth*, *Google Maps*, etc.

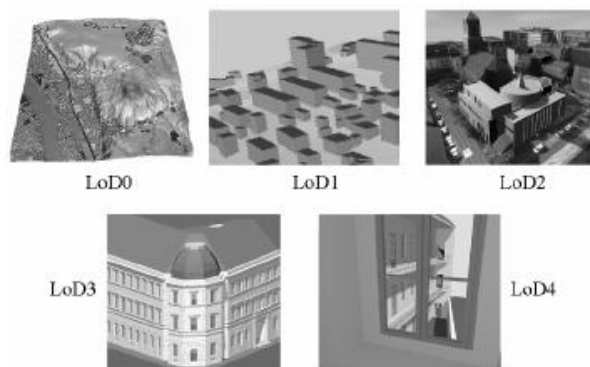
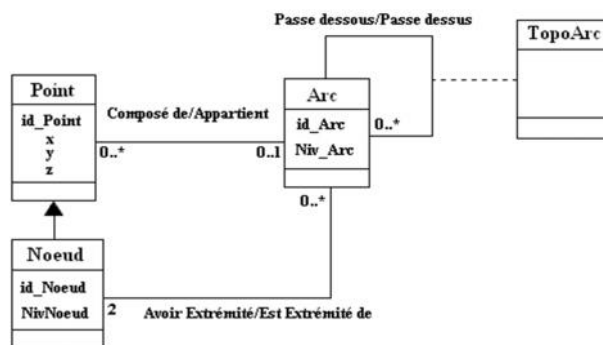


Figure 3.8. Level Of Détail (LOD) définis par CityGml

### 3.3.3. Modèle 3D topologique pour le Réseau Routier

*B.Bessaa* a développé un modèle 3D spécifique pour le réseau routier [8]. Ce modèle présente la particularité d'être bien adapté aux caractéristiques du réseau routier, en plus il demande peu d'investissements vu qu'il est une extension d'un modèle 2D existant. A l'aide du concept de niveau introduit dans ce modèle il est possible de créer un passage vers la 3D à partir d'une base de données représentée dans un modèle 2D. Ce concept sert à attribuer des niveaux aux arcs après la phase de digitalisation des arcs en 2D. L'affectation des niveaux se fait manuellement et suit des règles d'affectation. Les arcs qui sont sur le sol (la terre) prennent toujours le niveau 0, les arcs qui sont totalement en l'air prennent des valeurs pairs (2, 4, 6,...), ceux qui sont au sous-sol prennent des valeurs négatives (-2, -4, ...) et les arcs intermédiaires qui lient les arcs de niveaux pairs prennent la valeur impair intermédiaires (1, 3, 5,...). Après l'affectation des niveaux aux arcs, l'étape suivante se fait automatiquement et consiste à générer la topologie 2D qui décompose certains arcs aux niveaux des points d'intersection. Ces intersections, qui sont créées lors de cette étape, peuvent être réelles, comme elles peuvent être virtuelles lorsqu'il s'agit, au niveau sémantique, d'une intersection entre des tronçons de niveaux différents par exemple une intersection entre un pont et une route qui passe en dessous. L'étape suivante consiste à affecter des niveaux aux différents nœuds où chacun prend le même niveau que l'arc incident de niveau pair. Ensuite, il faut éliminer les nœuds inutiles qui représentent des intersections virtuelles et relier les arcs découpés par cette intersection lors de la génération du modèle topologique. Pour garder l'information qualitative, les arcs concernés et le point d'intersection sont sauvegardés dans la base de données. La dernière étape pour compléter le modèle en 3D consiste à calculer les altitudes des différents points. Dans la figure 3.9, nous présentons le diagramme de classe du modèle topologique 3D des objets linéaires.



**Figure 3.9.** Diagramme de classe du modèle topologique 3D (Structures linéaires)

Les altitudes des points appartenant aux arcs de niveau 0 (sol) sont acquises directement à partir du *MNT*, et les altitudes des points restants en se basant sur l'hypothèse suivante : « *Les arcs de niveau non nul (autre que les arcs se situant sur le sol) ont une pente constante* », le calcul des altitudes des points intermédiaires constituant un arc se fait comme suit :

$$H_i = \frac{\sum_{j=1}^i L_j H}{L}$$

Où :

H est la hauteur entre le nœud de départ et le nœud de destination ce qui nécessite la connaissance des altitudes des différents nœuds

L la longueur de l'arc.

$L_i$  est la longueur d'un tronçon  $i$  de l'arc.

L'avantage de ce modèle est qu'à partir d'un modèle 2D existant, un modèle 3D est généré avec peu d'investissements. Mais il est limité à un type spécifique d'objets géographiques (objets linéaires).

### Conclusion

La modélisation 3D dans les SIG est une démarche qui nécessite la modélisation des objets de façon proche à la réalité, en garantissant les mêmes fonctions fournies par les SIG en 2D.

Dans cet axe de recherche, deux approches sont possibles : la première consiste à ajouter la troisième dimension aux modèles 2D ce qui permet d'utiliser les mêmes moyens de stockage et gestion des modèles 2D sans modification importante. Les modèles 2.x-D sont limités en termes de représentation des différents aspects de la géométrie selon l'axe vertical et ne prennent pas en compte les objets volumiques. La deuxième approche représente les modèles 3D qui sont conçus par la majorité à partir des modèles géométriques utilisés dans les logiciels CAO. Ces modèles sont conçus principalement pour la visualisation 3D. Ils sont faibles en analyse spatiale. Pour que celle-ci soit très facilitée il faut l'existence explicite des relations topologiques dans la base de données. Plusieurs chercheurs ont introduits la topologie 3D dans ces modèles selon leurs besoins. Beaucoup de modèles hybrides sont apparues mais ils ne sont pas assez satisfaisants. L'incompatibilité entre ces modèles et tous ce qui existent comme moyens de stockage et structure de données en 2D, a retardé la 3D dans le domaine géographique.

Dans ce contexte, notre travail consiste à définir des passerelles de passage des modèles vectoriels topologiques 2D vers le 3D. Le chapitre suivant sera consacré à la conception et l'implémentation de ses passerelles en se basant sur le concept de niveau pour faire un passage du modèle topologique 2D vers un modèle topologique 3D qui permet de représenter les structures surfaciques et volumiques du domaine urbain (bâtiments) en plus des structures linéaires qui représentent le réseau routier.

## Chapitre 4 : Conception et implémentation des passerelles du modèle vectoriel topologique 2D vers le 3D

### Introduction

Dans le domaine de l'information géographique, la véritable 3D n'est pas encore atteinte. La plupart des modèles 3D sont inspirés de l'infographie et des systèmes CAO. Ils ne décrivent pas suffisamment tous les aspects géométriques et sémantiques des objets géographiques. Ces modèles ne tiennent pas compte de la spécificité de ces derniers [62].

Les logiciels commerciaux actuels sont limités à quelques fonctions de visualisation en 3D et ne fournissent pas de fonctions d'analyse en 3D.

Sur le plan d'acquisition et de stockage des données, les moyens d'acquisition en 3D ne sont pas en large disponibilité ou disponibles avec un coût élevé. Le manque en 3D est marqué aussi au niveau des standards et moyens de stockage des données.

Par contre, les modèles 2D sont généralisés. Ils offrent tous les moyens d'abstraction, de stockage et d'échange de l'information ainsi que les fonctions d'analyse spatiale. A cet effet, les premiers modèles 3D sont des extensions des modèles 2D. Malheureusement, ce type de modèles est limité à 2.x-D seulement (2.5d, 2.75d, 2.8d) et ils ne permettent pas la modélisation des objets volumiques et des relations topologiques selon l'axe vertical.

Par conséquent, Il est intéressant de développer des modèles 3D qui exploitent les données 2D déjà existantes avec le moindre coût [30]. La conception de ce type de modèles permet d'exploiter aussi les moyens de stockage et structures des données 2D, ils doivent fournir, aussi des outils de passage à la topologie 3D en ajoutant l'information selon l'axe verticale tout en gardant la topologie en 2D.

Dans ce sens, le travail que nous avons réalisé a consisté en la conception des passerelles du modèle vectoriel topologique 2D vers le 3D, en se basant sur l'exploitation du concept de *niveau*, introduit par *M .Bessaa* dans son modèle topologique 3D pour le réseau routier [8]. En effet, l'extension de ce modèle sur d'autres structures urbaines de types surfaciques et volumiques n'est possible que par la redéfinition du concept de *niveau* dans un contexte plus large, afin de fournir un moyen de représenter l'aspect topologique 3D des objets, représentés dans un espace 2-D quelque soient leurs dimensions puisque la notion de *niveau* dans les travaux précédents dans notre laboratoire est limitée aux structures réseaux.

Afin de généraliser la notion de *niveau*, pour modéliser les objets volumiques, nous avons commencé par étudier le modèle '*niveau\_ réseau*' déjà réalisé, faire ressortir ses points forts et ses limites pour ensuite passer à la généralisation et le traitement des objets en structure et topologie 3D.

#### 4.1. Notion de Niveau

Le *niveau* est un attribut de type entier affecté aux objets spatiaux particulièrement de type linéaire dans une base de données définie dans un espace 2D. La valeur (*Valniv*) de *niveau* d'un objet représente un code qui désigne sa position verticale par rapport aux autres objets. Le *niveau* de valeur *Valniv* égale à zéro (0) est affecté aux objets qui sont en même *niveau* que le sol, tandis que les objets en position au-dessus ou en-dessous du *niveau* 0 ont des *niveaux* positifs ou négatifs selon leur position par rapport au *niveau* 0.

Ce concept permet de quantifier la topologie selon l'axe z pour les objets définis sur un plan afin de distinguer entre ceux qui sont réellement au niveau du plan et ceux qui sont en l'air mais sont représentés seulement par leur projection sur le même plan. Il permet, aussi de préserver la topologie 2D et d'exploiter cette dernière pour ajouter la topologie selon l'axe vertical et particulièrement la relation au-dessus/en-dessous.

#### 4.2. Passerelle 2D vers 3D pour les structures linéaires

Le passage du 2D vers le 3D est défini dans le modèle topologique de réseau routier en 4 étapes (figure 4.1) [8]. Ces étapes représentent un processus semi-automatique en interaction avec l'utilisateur qui joue un rôle important lors du passage à la 3D depuis la digitalisation jusqu'à la création de la base de données 3D. Pour améliorer le modèle, il faut automatiser le processus de passage à la 3D en minimisant le nombre d'interventions manuelles. Le traitement automatique doit prendre en compte tous les cas particuliers et remédier aux problèmes d'ambiguïté.

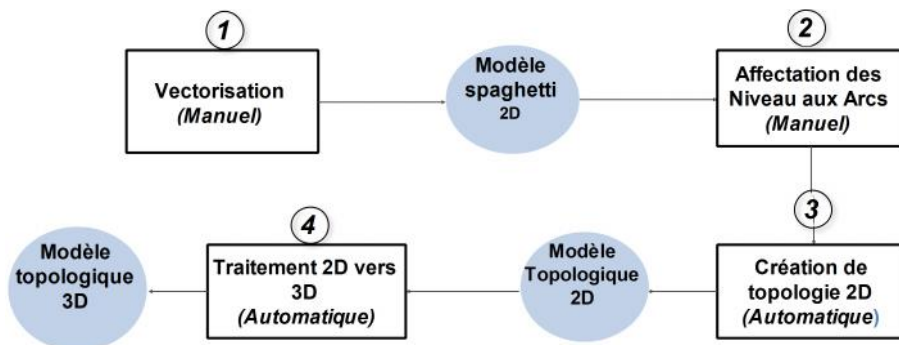


Figure 4.1. Passerelle topologique 2D vers 3D pour les structures linéaires

##### 4.2.1. Vectorisation

Le choix des tronçons par l'utilisateur lors de la vectorisation influe sur la cohérence des résultats obtenus. En effet, l'utilisateur marque la fin d'un arc et le début d'un autre arc lorsqu'il y a un changement de *Valniv* ou pour créer un arc qui représente un objet linéaire différent au niveau sémantique. Le résultat de la vectorisation est une base de données en modèle spaghetti qui ne prend pas en compte les relations topologiques entre les arcs.

### 4.2.2. Affectation des niveaux

L'étape suivante consiste en l'attribution des valeurs de *niveaux* aux arcs. Cette opération se fait de façon manuelle et respecte un ensemble de règles d'affectation. Les arcs au niveau du sol ont le *niveau* de *Valniv* 0. Les arcs qui sont au-dessus du sol ont des *niveaux* de *Valniv* pair (Pour les arcs au sous-sol on prend des valeurs négatives (-2, -4, ...), et les arcs intermédiaires entre deux *niveaux* pairs ont des *niveaux* de *Valniv* impairs.

Le nombre d'arcs qui représentent les tronçons de réseau routier qui ont un *niveau* 0 est plus important que le nombre des arcs des autres *niveaux*. L'utilisateur affecte seulement les *niveaux* aux arcs différents de zéro. Ensuite le *niveau* 0 est affecté au reste des arcs automatiquement sans violation des règles d'affectation pour assurer la cohérence dans le modèle.

La figure 4.2 présente un cas d'ambiguïté lors de l'affectation des *niveaux* aux arcs. Un arc de *niveau* pair est relié à deux arcs qui ont des *niveaux* impairs différents. Cette situation est due au caractère relatif de *niveau*. Afin de résoudre ce problème il faut fixer une nouvelle règle d'affectation des *niveaux*. Il est plus logique de garder le maximum d'information en affectant, le *niveau* de *Valniv* la plus grande à cet arc. De ce fait, l'arc présenté dans la figure 4.2 doit prendre la valeur de *niveau* 4 et pas 2. Ce choix permet de garder l'information d'existence de deux arcs de *niveaux* pairs différents en-dessous de l'arc envisagé.

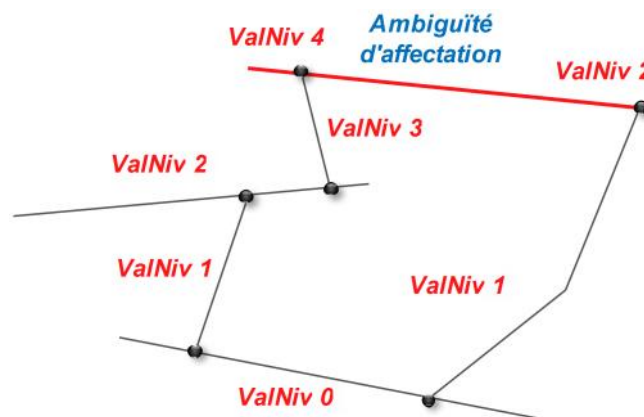


Figure 4.2. Ambiguïté d'affectation de niveau

Afin de connaître l'existence d'un autre passage dans le cas de présence d'un saut de *niveau*, la nouvelle règle consiste à l'attribution des *niveaux* aux arcs de *niveau* pair selon le *niveau* de *Valniv* impair le plus grand parmi les arcs reliés à celui-ci. Cette convention enlève l'ambiguïté et garde le maximum d'informations topologiques selon l'axe vertical.

### 4.2.3. Génération de la topologie 2D

La 3<sup>ème</sup> étape consiste en la création de la topologie en 2D. Au cours de cette étape les différents nœuds sont créés. Les nœuds représentent les limites des arcs, et peuvent être partagés par plusieurs arcs. Cette étape entraîne alors la décomposition de certains arcs au niveau des intersections

entre eux. Ces intersections peuvent être réelles ou virtuelles. L'utilité des nœuds virtuels est de trouver le point où les arcs qui ont des *niveaux* différents sont liés par une relation de type au-dessus/en-dessous.

#### 4.2.4. Traitement 2D vers 3D

Cette étape de traitement est décomposée en plusieurs phases :

- Affecter les *niveaux* aux différents nœuds automatiquement selon les règles suivantes :
  1. Les deux nœuds extrémités d'un arc de *niveau* pair prennent le *niveau* de l'arc.
  2. Les nœuds reliant des arcs de *niveaux* pairs différents prennent la valeur minimale des différents *niveaux*.
- La détection des nœuds virtuels qui sont créés à cause d'une intersection entre deux arcs de *niveaux* différents. Ensuite on doit relier les arcs découpés par cette intersection, puis supprimer le nœud virtuel et créer une relation (au-dessus/ en-dessous) entre les deux arcs à la position de l'intersection. Ceci représente le cas simple lorsque le nœud virtuel est le résultat d'intersection entre deux arcs pairs de *niveaux* différents. Dans certains cas particuliers à rare existence, le même point représente une confusion entre des nœuds réels et virtuels. Ces cas particuliers ont besoins d'un traitement différent. Si le nœud existe réellement en plusieurs *niveaux* différents il faut fusionner le point d'intersection en deux nœuds ou plus, qui ont la même position horizontale (x, y) mais avec des *niveaux* différents. La figure 4.3 présente des cas particuliers. Dans le cas 1 le nœud existe au *niveau* de valeur 2 et n'existe pas au *niveau* de valeur 0. Le traitement de ce cas consiste à créer un nœud de *niveau* 2 et à relier les deux arcs au *niveau* 0, car le nœud n'existe pas réellement. Dans le cas d'existence d'un arc ou plus avec un *niveau* impair qui partage le nœud d'intersection avec des arcs de *niveaux* pairs différents, il est nécessaire de déterminer le *niveau* pair où cet arc est connecté dans cette position. Les extrémités d'un arc impair ne peuvent pas être dans le même *niveau*. La règle qu'un arc de *niveau* impair est orienté toujours vers le *niveau* de *Valniv* supérieur permet de résoudre ce problème. Alors, pour chaque *niveau*, il faut déterminer s'il s'agit d'une intersection réelle ou virtuelle.

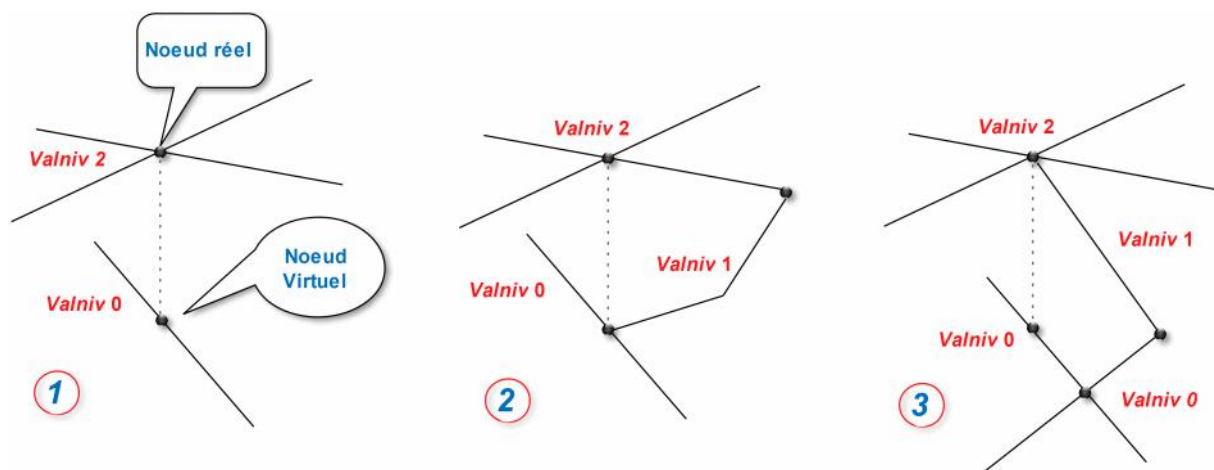
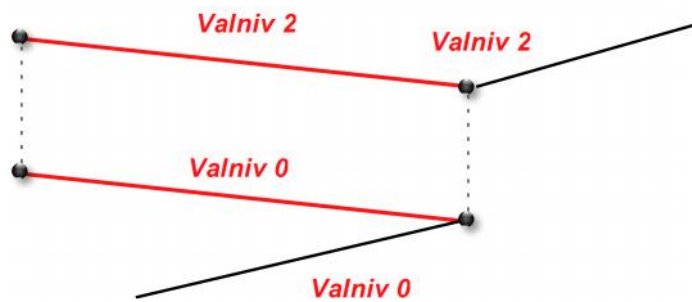


Figure 4.3. Confusion entre nœud-réel et nœud-virtuel

- La dernière phase dans cette étape consiste à compléter le modèle 3D par l'ajout automatique de la coordonnée z aux différents points. Cette coordonnée est relative au *niveau 0* (c'est la hauteur par rapport au sol).

Le modèle présenté est simple et utile dans le cas des objets linéaires. Car la projection d'une ligne définie dans un espace 3D sur un plan 2D est toujours une ligne. Sauf dans le cas des lignes verticales qui donnent un point sur le plan de projection orthogonale. Ce cas n'a pas de sens sémantique appréciable dans notre contexte pour le réseau routier et ce type d'objet n'a pas de présence décisif dans cette thématique. Cependant, les lignes projetés peuvent être confondues s'ils ont une même forme dans la même position horizontale, mais avec des *niveaux* différents (Figure 4.4.). Cette situation nécessite la duplication de la même ligne sur plusieurs niveaux pour faire le passage topologique vers l'espace 3D. Mais comment découvrir qu'il y a une confusion sur même ligne qui représente plusieurs niveaux ?



**Figure 4.4.** Confusion entre des lignes avec des niveaux différents

Afin de traiter ce cas, il est nécessaire de définir les objets comme des entités sémantiques avec des attributs en 2D. A ce niveau, un seul arc topologique peut être relié à plusieurs objets linéaires de *niveaux* différents. Ce qui permet de dupliquer le même arc sur les différents niveaux des objets lors du passage à la 3D. Une ligne qui représente un pont par exemple et qui a un *niveau 2* est confondue avec un tronçon de route qui se situe au *niveau 0*. Lorsqu'on passe en 3D, on crée l'arc qui représente le pont avec le *niveau 2* en plus de l'arc qui représente le tronçon de la route en-dessous avec le *niveau 0*. La possibilité de multiple représentations, pour le même arc dans l'espace 2D offerte par l'affectation de *niveau*, présente une continuité des modèles 2.x-D et résout les problèmes qui limitent ce type de modèles à une seule ou à quelques valeurs de z pour la même position en (x, y). Ainsi l'information topologique selon l'axe z est ajoutée. Cette information est relative par rapport au *niveau 0*. Par ailleurs, il est possible de représenter, dans un espace 2D, plusieurs objets qui ont la même position horizontale mais avec différents positions en vertical.

La Topologie et la sémantique se sont les aspects des objets réels les plus proches de la perception humaine. Ils ont plus d'intérêt au niveau de l'utilisateur. Lors de la vectorisation, l'utilisateur dessine les objets selon leur rôle sémantique et il prend en compte l'affectation des *niveaux*. Il en résulte de cette vectorisation que l'existence des objets est basé sur la sémantique et la

variation des *niveaux*. Des règles et des contraintes sur la relation entre objets et *niveaux* peuvent être définies afin de simplifier plus le processus manuel. Il est plus facile, pour l'être humain d'affecter un *niveau* à un pont que d'affecter un *niveau* à un arc sans attributs sémantiques. L'exemple de pont dans les structures routières représente un bon cas pour expliquer le rôle de niveaux sémantiques pendant l'affectation du *niveau*. L'exemple d'un pont qui a toujours une valeur de *niveau Valniv* plus grande qu'un tronçon situé en-dessous (un *niveau* pair plus grand que 0 doit être affecté à une ligne qui représente un pont).

Afin d'introduire ce concept de 'niveaux' dans les modèles d'objets complexes, la partie suivante de notre travail consiste à généraliser ce dernier pour permettre au modèle de représenter d'autres types d'objets tels que les bâtiments, qui ont déjà la notion de niveau (étages) associé à leurs paramètres sémantiques.

Les passerelles topologiques du 2D vers 3D sont plus difficiles à concevoir pour les objets de types surfaciques et volumiques que pour les objets linéaires qui ont un nombre de dimension interne en 1D. Ceux-ci ont une représentation simple et similaire soit dans l'espace 2D ou dans l'espace 3D avec un minimum modification topologique et de perte d'information. Tandis que les objets surfaciques (2D) et volumiques (3D) ont une représentation 2D qui génère un changement topologique et géométrique ainsi qu'une perte d'information et dans certains cas, la réduction de la dimensionnalité. Donc il est nécessaire de définir un processus de passage basé sur la reconstruction de la topologie 3D à partir des modèles 2D. Lors de ce passage, le concept de *niveau* est introduit, où des Valeurs *Valniv* de niveaux sont associées comme des paramètres aux objets définis par l'aspect sémantique.

### **4.3. Description du Modèle Topologique 3D**

Dans le milieu urbain l'objet surfacique est souvent utilisé pour représenter les bâtiments en 2D, qui sont à l'origine des volumes dans l'espace réel 3D. Les modèles topologiques dans l'espace 2D représentent les objets surfaciques (2D) à partir de leurs contours linéaires 1D. Une relation topologique arc-face est définie selon la contrainte qu'un primitif '*arc*' est adjacent, au maximum à deux faces (droite/gauche). De façon similaire dans l'espace 3D les objets volumiques 3D sont représentés par leurs frontières surfaciques (2D). Chaque primitive face est reliée à, au plus deux objets volumiques. Cette relation face -volume interprète l'orientation des objets surfaciques. D'autre part, un arc dans un espace 3D peut être relié à plusieurs faces comme un nœud en 2D qui peut être relié à plusieurs arcs. Les modèles topologiques 3D peuvent être considérés comme une extension des modèles topologiques en 2D dans lesquels les objets, quelque soient leurs dimensions internes (0D, 1D, 2D, ou 3D), sont représentés dans un espace 3D. La modification de l'espace de représentation des objets, définis en 2D, exige une nouvelle représentation dans un modèle topologique 3D. Lors de cette transformation le processus de passage préserve les relations topologiques 2D, et construit la topologie

3D selon l'axe verticale. Pour atteindre cet objectif, le concept de *niveau* est associé aux objets surfaciques en plus des objets linéaires et ponctuels. La figure 4.5 présente un modèle topologique 3D simple et plus général, dans lequel le *niveau* représente un attribut associé aux différentes primitives topologiques.

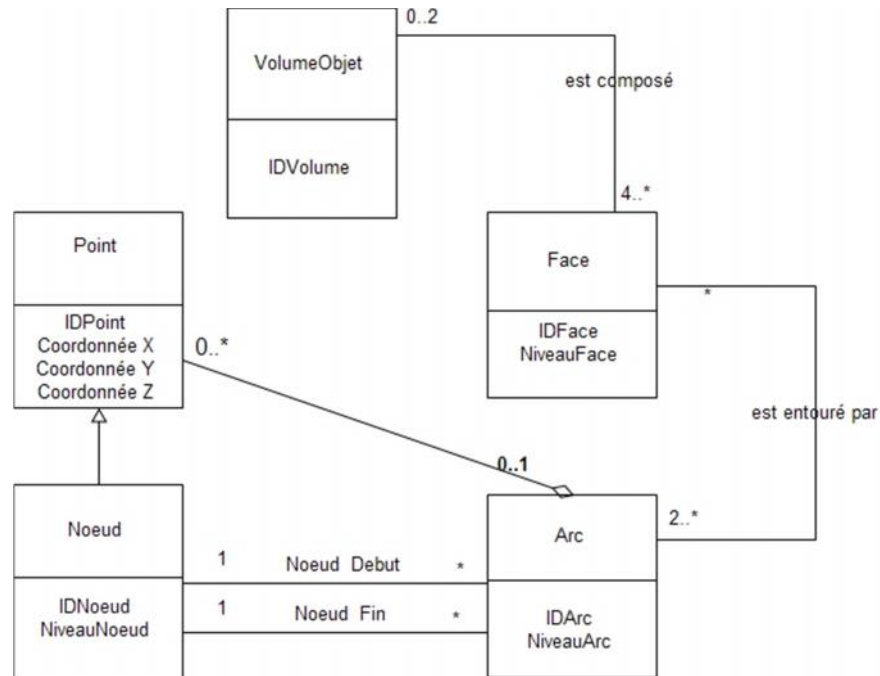
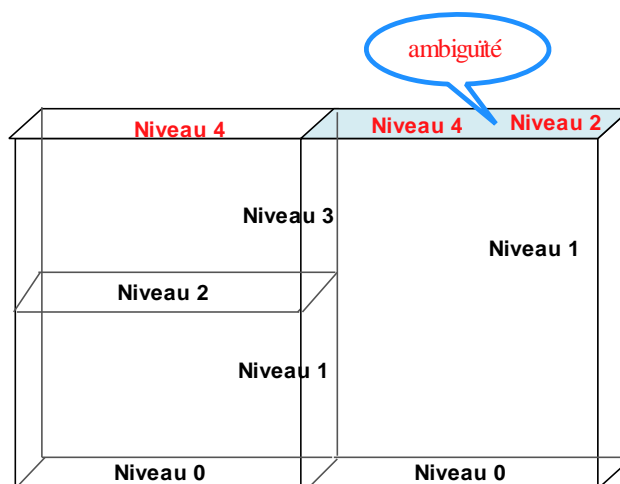


Figure 4.5. Modèle Topologique 3D

Dans un contexte urbain, par exemple pour représenter un bâtiment, les cardinalités de la relation d'adjacence entre face et arc de type plusieurs (\*) peuvent être simplifiées à 4, ce qui signifie dans le sens de l'orientation, une seule face dans chaque direction (droite, gauche, au-dessus ou niveau supérieur, en-dessous ou niveau inférieur).

#### 4.3.1. Niveaux des primitives en 3D

Le *niveau* associé à une primitive face peut être pair ou impair. Les faces qui ont des *niveaux* pairs sont composées d'arcs de même *niveau*. Les faces de *niveaux* impairs représentent des objets intermédiaires entre deux *niveaux* pairs différents. Comme un arc intermédiaire a deux nœuds extrémités qui ont deux *niveaux* différents, une face intermédiaire est reliée avec deux lignes de *niveaux* pairs différents. Une face intermédiaire est limitée aussi par deux arcs intermédiaires avec le même *niveau* impair. Il reste les nœuds qui prennent toujours un *niveau* pair.



**Figure 4.6.** Exemple d'affectation des niveaux aux primitives faces

La figure 4.6 présente un exemple d'association des *niveaux* aux faces en 3D. C'est un cas particulier qui pose une ambiguïté lors d'affectation des *niveaux* à une face de *niveau* pair. Cette face est reliée à deux faces intermédiaires de *niveaux* impairs différents.

Afin de garantir la cohérence du modèle une règle ou convention doit être fixée en présence d'ambiguïté qui consiste à retenir le *niveau* le plus grand. Cette situation a été déjà vue précédemment avec les structures linéaires. Cette règle a permis de garder le maximum d'informations topologiques selon l'axe z. La cohérence de cette convention avec le modèle topologique 2D est garantie. L'adjacence horizontale entre deux faces est décrite par une relation topologique 2D dans laquelle les deux faces ont le même *niveau* pair.

#### 4.4. Passerelle topologique 2D vers la 3D (Structures Surfiques : Bâtiments)

La difficulté de passage pour les objets volumes et surfaces réside dans la perte d'informations et de dimension avec leur représentation sur un plan et le passage du 2D vers le 3D consiste en la reconstruction topologique des objets en 3D à partir de leur représentation réduite en 2D.

##### 4.4.1. Objet Volumique en (2D et 3D)

Un objet volume (3D) est représenté dans un espace 2D par un objet surfacique 2D ce qui réduit son nombre de dimension à 2. Dans le sens topologique la présentation en 2D réduit aussi la topologie 3D à une topologie en 2D. Cependant, le processus de reconstruction des volumes est basé sur sa représentation surfacique en 2D dans ce sens ou un volume en 3D est représenté à partir de ses frontières surfaciques dans un modèle topologique 3D [39]. La restitution de la topologie 3D d'un volume consiste en la restitution des primitives surfaciques qui ont une relation topologique avec le volume.

#### 4.4.2. Objet Surfaccique (2D et 3D)

Les faces sont représentées dans un espace 2D selon leurs positions par rapport au plan de projection. Dans la plupart des cas, les faces sont représentées par le même nombre de dimensions (2D) dans l'espace 2D ou 3D. Il existe un cas particulier où la face est en position orthogonale par rapport au plan de projection. Dans ce cas exceptionnel, la représentation en 2D est restreinte à un objet linéaire (1D). Le processus de passage en 3D de ce cas particulier consiste à construire une face intermédiaire de *niveau* impair à partir de sa représentation linéaire en 2D.

Selon la position et le *niveau* des faces, 3 cas possibles sont distingués :

- Les faces qui ont un *niveau* pair sont représentées de la même façon, en 2D ou en 3D, par des faces (2D). Les faces qui ont le même *niveau* pair gardent leurs relations topologiques 2D entre elles dans l'espace 3D. Le passage 3D consiste à intégrer cette face au niveau topologique approprié.
- Les faces qui ont un *niveau* impair : il existe deux cas selon la position verticale :
  - Les faces qui sont en position orthogonale (verticale) par rapport au plan de projection, elles seront représentées par des lignes seulement en 2D. Le passage en 3D consiste à la construction et l'intégration dans la topologie 3D d'une face intermédiaires de *niveau* impair. En effet, un *niveau* impair en 3D est associé à cet objet qui est représenté par une ligne seulement en 2D.
  - Les faces qui ont une position inclinée sont représentées dans l'espace 2D par des faces horizontales. La distinction entre les arcs qui ont le *niveau* pair inférieur et ceux qui ont le *niveau* pair supérieur permet de définir l'orientation de la face en 3D et mettre cette face dans sa position selon l'axe z. Les autres arcs restants de la face prennent le même *niveau* (impair ou intermédiaire) que la face. Les arcs qui ont des *niveaux* pairs différents pour une face intermédiaire et n'ont pas un nœud en commun, ils sont reliés par les arcs intermédiaires (*niveau* impair).

Les 3 cas possibles sont différents en '*niveau*' s'il est pair ou impair et en '*nombre de dimensions*' de représentation en 2D en comparaison avec celle en 3D. Afin d'apporter plus de spécification sur notre modèle il est utile de distinguer entre les faces qui ont le *niveau* pair et celle qui ont le *niveau* impair ainsi qu'entre les arcs. Ces deux cas représentent deux types de primitives qui sont différentes en orientation et en topologie dans l'espace 3D.

#### 4.4.3. Affectation des niveaux en 2D

L'affectation des *niveaux* en 2D est effectuée au niveau des objets afin d'éviter la confusion et l'ambiguïté en cas d'attribution des *niveaux* aux primitives, car deux objets de *niveaux* pairs différents peuvent être représentés en 2D avec des primitives en commun s'ils ont une partie qui a la même position horizontale. Cette possibilité qu'une primitive peut être reliée à plusieurs objets est traduite par les cardinalité (1,\*) dans la relation d'agrégation entre primitives et objets sur la figure 4.7.

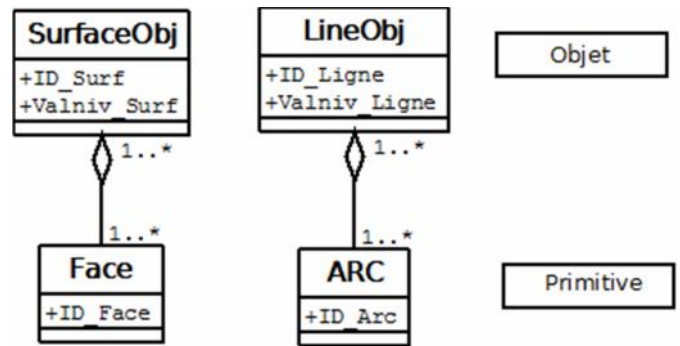


Figure 4.7. Composition des objets avec des primitives topologiques

L'affectation des *niveaux* aux objets permet d'avoir la différence de la position selon l'axe z pour deux objets qui partagent la même représentation dans l'espace 2D.

#### 4.4.4. Aspect Sémantique

Un objet représente une entité sémantique qui est associée à un ensemble d'attributs pour décrire son rôle dans la réalité. Dans notre travail, les entités définies au niveau sémantique comme (mur, plancher, toit, pièce...) sont représentées à partir des objets en 2D. Un objet qui est construit par l'être humain a l'avantage d'avoir une certaine régularité, celle-ci réside dans sa forme, son nombre de dimension, et l'orientation en vertical, ainsi que d'autres paramètres qui reflètent son rôle sémantique dans la réalité. Cette particularité est utilisée, dans notre cas, pour décrire un objet au niveau sémantique. Les objets qui appartiennent à la même classe thématique ont les mêmes paramètres.

La détermination de l'ensemble des paramètres, nécessaires pour faire le passage à la 3D pour chaque classe d'objets sémantiques, permet de faire un passage topologique à la 3D de façon automatique. L'aspect sémantique représente l'aspect le plus proche à la perception humaine. L'intervention manuelle se fait au niveau sémantique. Le modèle sémantique paramétré permet d'automatiser le processus de passage du 2D vers le 3D.

#### 4.5. Processus de passage topologique du 2D vers le 3D

Le processus de passage topologique du 2D vers le 3D est défini selon le schéma présenté dans la figure 4.8 en 4 étapes :

1. Importation des données et vectorisation : cette étape se fait manuellement, le résultat de cette étape est une base de données en modèles spaghetti 2D.
2. La création d'une base de données topologique 2D automatiquement. Une base de données topologique est générée à partir des données 2D en modèle spaghetti. Cette base de données génère les primitives topologiques : Face, Arc et nœud et les relations topologiques 2D qui existent entre eux.

3. La création des objets à partir des primitives topologiques et l'affectation des attributs sémantique ainsi que les valeurs de *niveau*.
4. Traitement du 2D vers le 3D : reconstruction topologique 3D à partir de la base de données topologique 2D en affectant les différents paramètres et particulièrement le *niveau* aux objets.

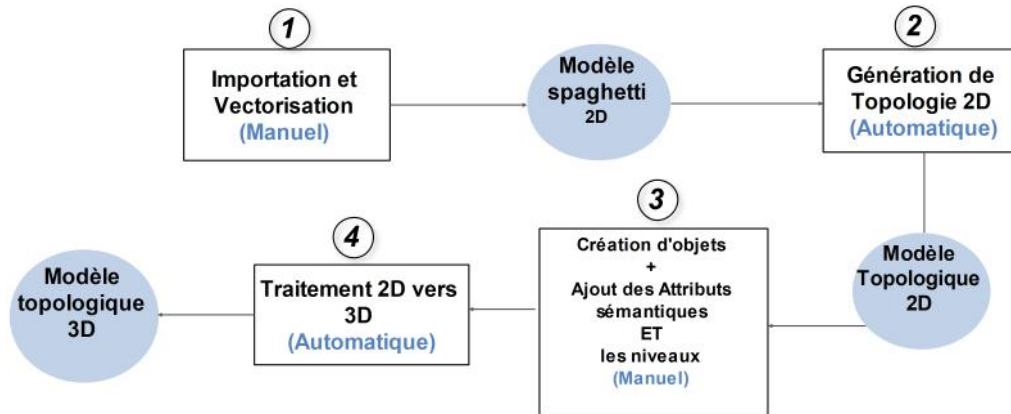


Figure 4.8. Passerelle topologique 2D vers 3D pour les structures Surfacades

#### 4.5.1. Importation et vectorisation

La première étape consiste en l'importation des données à partir d'une source externe. Un fichier (format DXF) vectoriel 2D généré par un logiciel CAD, ou un plan 2D scanné sont deux exemples de source des données. La vectorisation consiste en la création des données vectorielles et les stocker dans une base de données selon le modèle spaghetti 2D.

#### 4.5.2. Génération de topologie 2D

L'étape suivante est la génération de la topologie 2D. Cette étape est automatique, elle permet la génération de la topologie en 2D, dans laquelle un objet est décrit par des primitives de base de dimension inférieure qui constituent leurs frontières. L'intégration des données dans un SGBD-S, qui gère la topologie, facilite cette tâche. Un SGBD-S fournit des moyens topologiques pour générer la topologie automatiquement.

#### 4.5.3. Création des objets selon l'aspect sémantique

Cette étape est très importante. C'est l'étape où l'utilisateur crée les différents objets en 2D qui représentent des entités sémantiques. Un ensemble de primitives définissent un objet, chaque entité exige un ensemble de valeurs pour paramétrer l'objet créé. Ces paramètres sont utilisés par le processus de passage en 3D. Nous proposons dans le cas d'un bâti, par exemple, l'ensemble d'entités sémantiques suivantes :

**L'entité Mur** : Un mur ordinaire est représenté en 2D avec une ligne de *niveau* pair. Un *niveau* pair est affecté manuellement à la ligne en 2D, qui donne le *niveau* à la ligne, de *niveau* supérieur de ce mur. L'objet ligne 2D est la projection orthogonale d'une face de type intermédiaire. Alors un mur

est représenté en 3D avec une face intermédiaire orthogonale au plan de projection. Cette face prend le *niveau* impair suivant par rapport au *niveau* de plan de projection. Le passage du 2D vers le 3D pour un mur consiste en la création d'une face intermédiaire de *niveau* impair à partir d'une ligne en 2D. D'Autres attributs qui caractérisent un mur comme la hauteur et l'épaisseur sont ajoutés.

**L'entité plancher** : un plancher est représenté par une surface normale de *niveau* pair. Le passage à 3D garde la même topologie qu'en 2D dans le même *niveau*.

**L'entité plafond ou toit** : un plafond ou toit est représenté avec une surface normale de *niveau* pair supérieur par rapport au *niveau* du plan de projection. Le passage du 2D vers le 3D consiste en l'intégration de la même surface 2D dans le *niveau* pair supérieur.

**L'entité Escalier** : est représenté avec une surface horizontale en 2D. Sa représentation en 3D est une surface intermédiaire inclinée par rapport au plan de projection. Le passage du 2D vers le 3D de ce cas exige la distinction entre les arcs de *niveau* pair du plancher et les arcs de *niveau* pair supérieur.

**La classe pièce** : une pièce est représentée avec une surface en 2D plus un *niveau* qui représente le *niveau* pair de sa face supérieure. Une pièce est représentée par un volume en 3D. Le passage du 2D vers le 3D consiste à créer un volume à partir d'une face horizontale. La topologie d'un volume exige l'orientation des faces. Les Faces de *niveau* pair ont un volume au-dessus qui est le volume au *niveau* impair supérieur et un volume en-dessous au sens de *niveau* impair inférieur. Les Faces intermédiaires ont un volume à droite et à gauche. Elles sont orientées selon leur limite inférieure qui est une ligne de *niveau* pair. Sur le plan sémantique une pièce est représentée en 3D à partir d'un plancher, des murs et un toit.

#### **4.5.4. Traitement du 2D vers le 3D**

Cette étape est la plus importante. Elle permet de construire la base de données topologique 3D. Nous utilisons l'attribut *niveau* et la représentation des objets en 2D pour reconstruire les objets selon un modèle topologique 3D. Les valeurs qui caractérisent chaque objet sémantique sont utilisées pour construire la topologie 3D au niveau d'objets et primitives. Le traitement du 2D vers le 3D sert à intégrer les objets reconstruits selon leurs *niveaux* et leurs dimensions. Le processus consiste à relier les objets qui ont le même *niveau* pair par des relations topologique en 2D. Les objets intermédiaires qui ont un *niveau* impair jouent le rôle de liaison entre deux *niveaux* pairs différents. Cette nouvelle topologie est représentée par un modèle topologique 3D adéquat. La dernière étape consiste en l'ajout des altitudes aux différents points pour donner l'aspect géométrique 3D aux objets.

### 4.6. Modèle Topologique 3D (extension de la topologie 2D)

Le modèle topologique 3D est considéré comme une extension du modèle topologique 2D. La distinction entre les deux types de topologie offerte par le concept de *niveau* est traduite explicitement par la création des nouvelles primitives plus spécifiées dans le modèle topologique 3D (figure 4.9).

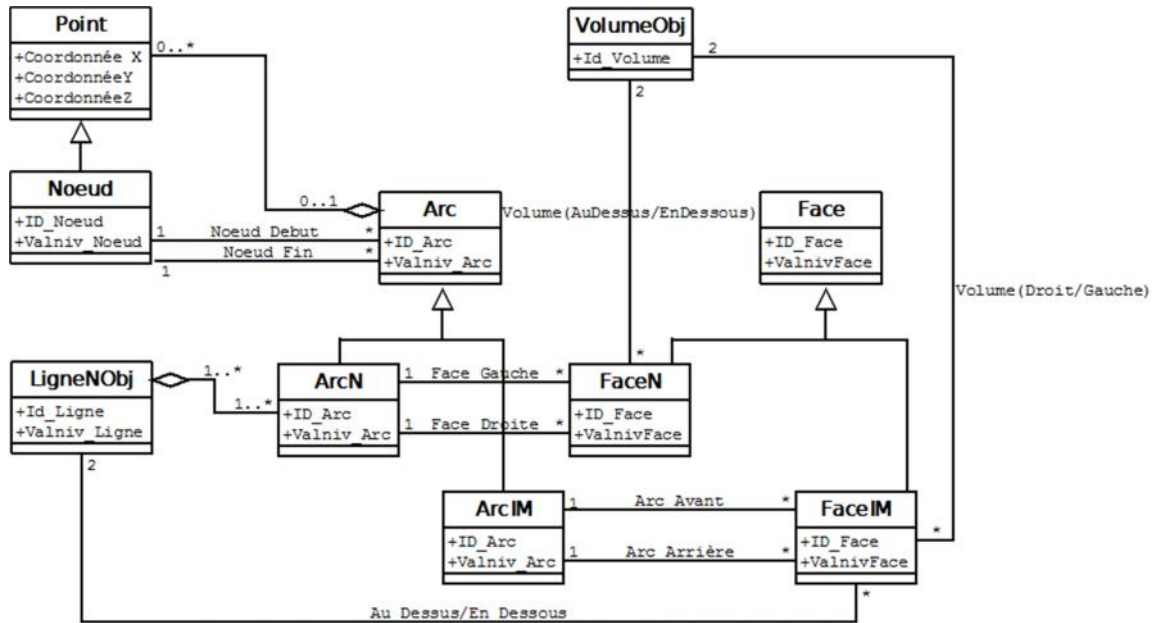


Figure 4.9. Modèle Topologique 3D (Extension 2D)

Les deux primitives *ArcN* et *FaceN* sont des héritiers plus spécifiés des primitives *Arc* et *Face*. La relation entre les deux primitives *ArcN* et *FaceN* adjacentes qui ont le même niveau (pair) est équivalente à une relation topologique en 2D entre *Arc* et *Face*, dans laquelle chaque *ArcN* est relié à, au plus deux *FaceN*. Il s’agit, dans ce cas-là, d’une topologie équivalente à celle en 2D entre les objets de même niveau pair.

Les objets intermédiaires entre les niveaux pairs, comme les murs sont représentés par les primitives de type *FaceIM* qui est une face avec un niveau impair. *FaceIM* est en relation topologique avec deux niveaux pairs différents, à travers deux lignes de type *LineObj*. Les deux lignes délimitent cette face verticalement l’une de niveau inférieur (au-dessous) et l’autre de niveau supérieur (au-dessus). Ces deux lignes sont liées aux deux arcs intermédiaires de niveau impair. Un arc intermédiaire est distingué par une nouvelle primitive *ArcIM*. Les deux arcs *ArcIM* (avant, arrière) qui délimitent une *FaceIM* ont des extrémités initiales dans le niveau inférieur et finales au niveau supérieur. Les deux extrémités initiale et finale (Nœud Début/Fin) de la ligne de niveau pair inférieur sont les deux extrémités initiales des deux arcs intermédiaires *ArcIM* avant et arrière respectivement. Les deux extrémités initiale et finale (Nœud Début/Fin) de la ligne de niveau pair supérieur sont les extrémités finales des deux Arcs *ArcIM* (Avant/arrière) respectivement.

L’ajout de ces nouvelles primitives topologiques apporte plus de spécificité dans les relations entre les objets et permet de garder la topologie 2D sans aucune modification. Au niveau physique,

l'implémentation de ce modèle est possible avec les moyens de stockage limités à la topologie 2D. En effet, la figure 4.9 montre bien que les primitives topologiques 2D sont conservées.

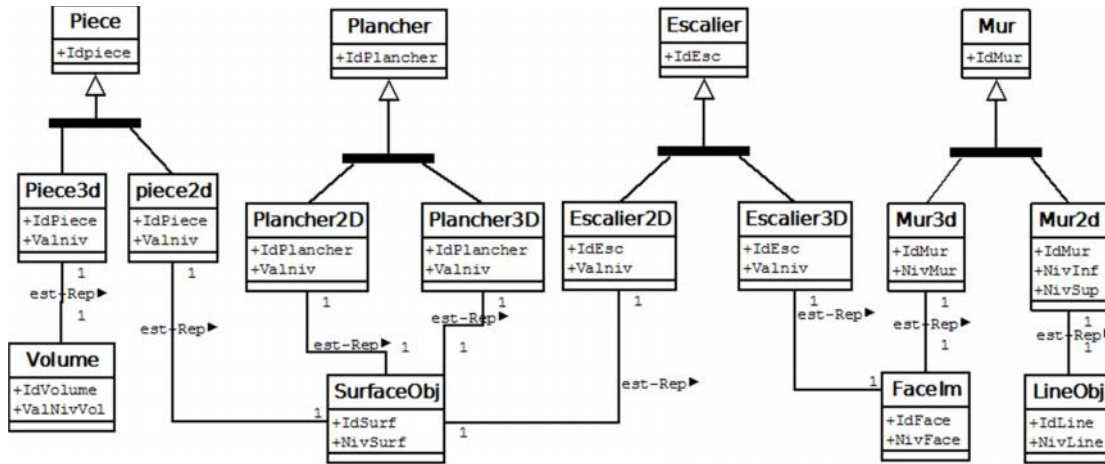


Figure 4.10. Modèle Sémantique (présentation 2D et 3D)

La figure 4.10 représente le modèle sémantique associé au modèle proposé. Chaque entité sémantique est représentée avec un objet topo\_ géométriques en 2D et en 3D.

#### 4.6.1. Description Sémantique et précision Topologique

Des valeurs de *niveaux Valniv* différentes sont affectées aux objets en 2D à chaque plan de projection de ces derniers qui représente le plan de chaque niveau (étage) dans le contexte sémantique. Ceci définit l'unité sémantique de variation des valeurs de *niveaux* par un étage (niveau). Dans ce cas, le sous espace occupé par un étage est décrit topologiquement par un seul *niveau* de valeur impair et un changement de *niveau* est effectué pour chaque étage. Ce sous espace est délimité exactement par deux valeurs de *niveaux* 'pair inférieur' et 'supérieur'. L'affectation des *niveaux* dépend donc du niveau détail (LOD :Level Of Detail) sémantique utilisé.

*La topologie de l'entité escalier est représentée par une seule face (FaceIm) de Valniv impaire intermédiaire entre les deux niveaux de Valniv pairs qui correspondent au plan de plancher et le plan de plafond de l'étage considéré. Les détails d'escalier ne sont pas pris en compte. Dans une représentation plus détaillée des valeurs de niveaux locaux ou sous-niveaux sont affectées aux objets de la même façon que le processus d'affectation des valeurs de niveau. Le sous-niveau de ValSniv 0 correspond au niveau de Valniv pair inférieur et le sous-niveau de valeur ValSniv m (2n+1) impair maximal est affecté à l'objet intermédiaire lié au niveau de Valniv pair supérieur.*

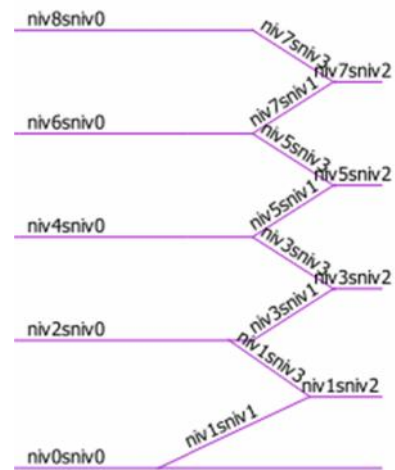
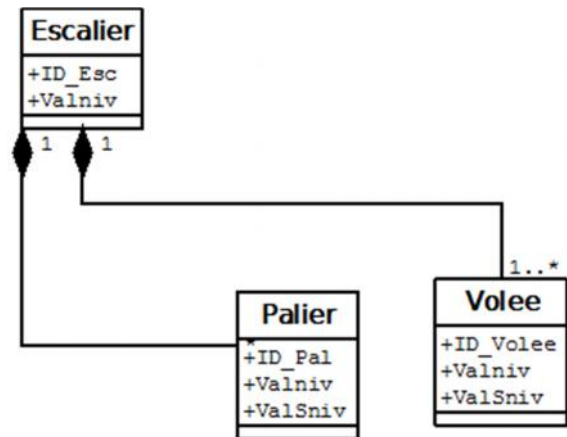


Figure 4.11. Affectation des niveaux/Sous-niveaux (coupe verticale d'escalier)

La figure 4.11 montre les différents pairs des valeurs de (*Valniv/ValSniv*) des (*Niveaux /Sous-Niveaux*) selon l'axe vertical au *niveau* d'escalier.

Les détails d'escalier sont ajoutés comme des entités sémantiques qui sont des faces de même *niveau* impair de l'entité escalier. Des *sous-niveaux* en plus sont affectés à ces entités selon leurs positions topologiques par rapport à l'objet de *niveau* pair inférieur. Les faces qui représentent les volées ont des valeurs de *sous-niveaux* pairs et sont représentées avec des *faceN* dans l'espace 3D qui garde la même topologie en 2D. Les faces qui représentent les paliers ont des valeurs de *sous-niveaux* impairs donc elles sont représentées par des objets *faceIM* en 3D et délimitées par deux (niveaux/sous-niveau) pairs différents.



**Figure 4.12.** Composition d'escalier

La figure 4.12 montre la composition d'escalier à partir de ses détails (palier, volée).

Le concept de résolution sémantique (LOD) est un standard de l'OGC pour la visualisation des modèles de villes en 3D où plusieurs LOD sont proposés. Dans le présent cas, nous avons limité notre étude actuelle à une hiérarchie à deux niveaux seulement. D'autres niveaux de détails peuvent être ajoutés pour décrire, par exemple, les marches d'escalier ou les ouvertures (portes et fenêtres), etc.

#### 4.7. Mise en œuvre des passerelles 2D>3D

Les systèmes de gestion des bases de données SGBD-S représentent un moyen utile pour le stockage des données dans le domaine de la géomatique. Parmi les fonctionnalités de base d'un SIG, retrouve le stockage et l'analyse des données spatiales. Cette analyse repose sur la manipulation des données géographiques stockées dans un format approprié, car les données géographiques sont volumineuses il est plus commode d'utiliser un SGBD-S pour stocker les données géographiques. Ces dernières sont caractérisées par un aspect sémantique et un aspect spatial. La sémantique est représentée par des types classiques alphanumériques, mais l'aspect spatial nécessite une extension spatiale. L'extension spatiale d'un SGBD-S fournit des types spatiaux qui représentent la géométrie comme point, ligne et polygone. Les SGBD-S offrent un ensemble d'opérations pour manipuler les objets spatiaux. L'index spatial est utilisé pour garantir un accès rapide et non séquentiel aux données spatiales. Les données géographiques sont caractérisées aussi par le stockage explicite des relations topologiques entre les objets dans les modèles topologiques. Le stockage de l'information qualitative est très important pour faire l'analyse spatiale. Les SGBD-S relationnels sont utilisés pour stocker la topologie via les relations entre les différentes tables.

##### 4.7.1. SGBD-S et 3D :

Plusieurs modèles topologiques 3D sont apparus ces dernières années mais ils demeurent théoriques sans implémentation explicite. Cependant quelques chercheurs ont lié leur recherche avec une implémentation dans un SGBD [33]. La majorité de ces modèles ne sont pas compatibles avec les

SGBD-S conçus à l'origine pour le stockage des données selon les modèles topologiques 2D, Ils sont donc limités en termes de représentation de la topologie 3D. Quelque SGBD-S ont intégré la coordonnées Z dans leurs types de données mais ils n'ont pas fourni le stockage de la topologie et les fonctions d'analyse nécessaires en 3D. Une autre insuffisance en 3D dans les SGBD-S consiste en l'absence d'un type spatiale volume pour stocker les objets en 3D et par conséquent, un objet volumique est représenté par un ensemble de primitives de type polygone ou par un seul objet de type multi-polygones.

#### **4.7.2. Choix du SGBD-S :**

Il existe de nombreux SGBD qui ont une extension spatiale : PostgreSQL et son extension spatiale PostGIS, MySQL avec son extension spatiale MyGIS, SQLite avec son extension spatiale Spatialite, et Oracle Spatiale, etc. Nous avons choisi PostgreSQL/PostGIS. C'est un SGBD objet-relationnel libre caractérisé par les points suivants:

- Il est reconnu au niveau mondial comme le premier SGBDS Open Source respectant la norme SQL et les spécifications de l'OGC (Open Geospatial Consortium).
- Il utilise l'index spatial pour améliorer les performances d'exécution des opérations spatiales.
- Il utilise la géométrie et la topologie 2D mais il supporte les coordonnées en 3D.
- Il dispose de nombreux outils d'administration. L'outil *psql* permettant de saisir des requêtes SQL en ligne de commande. L'outil populaire est *PgAdmin* qui est un outil graphique libre.
- Il possède un schéma topologique en 2D : **Topology** avec un ensemble de fonctions topologiques [51]. Ces dernières sont utilisées pour gérer les différentes primitives topologiques de façon cohérente.
- En plus du langage de requête SQL, il possède son propre langage procédural PL/PGSQL.

Pour tous ses avantages nous avons donc choisis PostgreSQL/POSTGIS comme SGBD-S. Le schéma *topology* de POSTGIS (version 2.0.4) nous permet de créer des schémas topologies en 2D.

Dans le contexte de notre travail, les algorithmes de passage 2D>3D sont traduits en fonctions PL/PGSQL en combinaison avec le langage python. Le langage Python est assez simple d'utilisation mais surtout commun à plusieurs logiciels SIG comme ArcGIS dont une console Python est incorporée.

#### **4.8. Passerelle pour le réseau routier (structure linéaire)**

Le modèle de réseau routier proposé par Mr Bessaa a été déjà utilisé dans un projet de magister pour réaliser un SIG en ligne avec une application de guidage qui calcule le plus court chemin entre deux points dans le réseau routier dans une zone caractérisée par la diversité de son infrastructure routière: région ouest d'Alger dont les coordonnées UTM 31N sont : (503640.304 4064939.718, 507620.395 4066414.078) [6].

Pour notre part, nous avons proposé un algorithme de passage à la topologie 3D basé sur le traitement de tous les cas étudiés et détaillés ci-dessus, ainsi que la détection automatique des intersections virtuelles afin de supprimer les nœuds qui n'existent pas dans la réalité et reconstruire les nœuds selon le niveau où ils existent réellement.

#### Algorithme 4.1. Traitement topologique 2D>3D de la topologie réseau

**Entrée** : Topologie 2D Table des Nœud : Tnode(IdN,NivN), Table d'Arcs : TArc(IdA,IdAPr,NivA,Fnod,Tnod).

**Sortie** : Schéma Topologique 3D : Table des Nœuds : Tnode(IdN,NivN), Table d'Arcs : TArc(IdA,IdAPr,NivA,Fnod,Tnod), TArcTopo(IdA,Point)

**Fonction AjouterNoeudNive(Node Nœud , Niv :Entier Pair) : // Fonction ajouter un nœud selon un niveau%**

**Debut Fonction**

**Si** Node.NivN<> Niv **alors** :

NewNoeud :=AjouterNoeud(Node,Niv) // Ajouter même nœud avec niveau Niv

**Si Non** :

NewNoeud :=Node ;

**Fin Si**

Rendre NewNoeud ;

**Fin Fonction.**

**DEBUT**

**Pour chaque** nodeAct **dans** Tnode **faire** : // examiner chaque nœud dans la table

Nbnv := Nombre des niveaux pairs des arcs reliés à nodeAct ; // Calcul du Nombre des niveaux pairs dans ce nœud

EnsNive := [NivPinf.....NivPsup] ; // ensemble ordonnées des niveaux pairs des arcs relié à NodeAct

**Si** Nbnv>=2 **Alors** // Tester l'existence de plusieurs niveaux pairs à ce nœud

**Pour** NivPair allons de NivPinf à NivPsup **faire** : // Examiner chaque niveau pair de EnsNive

NbAp :=Nombre des arcs pairs de niveau NivPair reliés à NodeAct ; // Le nombre des arcs par niveau pair

Bool :=Faux ;

**Pour chaque** ArcI Arc de niveau impair Nivimp relié à NodeAct **Faire** : // Chercher l'existence des Arcs Impairs%

**Si** ((Nivimp=NivPair+1) **ET** (ArcI.Fnode=NodeAct)) **OU** ((Nivimp<NivPair) **ET** ( ArcI.Tnode=NodeAct)) **Alors** :

// On Suppose que l'orientation des Arcs impair est Toujours de niveau pair inferieur vers le niveau supérieur

**Si** Bool =Faux **Alors** :

Bool :=Vrai ;

NewNoeud :=AjouterNoeudNive(NodeAct ,NivPair) ;

**Finsi**

ArcI.Extrémité :=NewNoeud # Extrémité =Fnode /Tnode %

**FinSi**

**Fin Pour**

**Si** Bool =Vrai **Alors** : // l'intersection existe vraiment avec des arcs impairs

**Pour Chaque** Arc ArcP de niveau pair = NivPair **Faire** :

**Si** ArcP.Extrémité =NodeAct **alors** :

ArcP.Extrémité :=NewNoeud ; // Extrémité = Fnode / Tnode selon le cas

**Fin Si**

**Sinon** **Si** (NbAp=2) **ET** (IdApr1=IdApr2) **Alors** : // Arc1 et Arc2 sont les 2 arcs connectés à NodeAct

ArcI=Union (Arc1, Arc2) // Relier les deux arcs

SupprimerArc(Arc2) // Supprimer l'entrée de Arc2

AjoutidArc(ArcI, TopoArc,NodeAct.(X,Y)) ; // ajouter l'arc et point à la topologie d'arc dans ce point

**Si** NodeAct.NivN= Nivpair **Alors** : // le niveau examiné est le niveau du nœud

**Si** Nivpair<> NivPsup **Alors** :

NoeudAct.nivN= EnsNive.NivSuivant ; // Modifie le niveau du nœud examiné

---

```

    Sinon                                     // le dernier niveau est atteint
        SupprimerNoeud(NodedAct) ; // Supprimer le nœud de la table des noeud
    Fin Si
    Fin Si
    Sinon : // le nœud s'existe entre deux arcs digitalisés ou extrémité d'un seul arc ou intersection de plus de 2 arcs
        NewNoeud :=AjouterNoeudNive(NodeAct ,NivPair) ;
        Pour Chaque Arc ArcP de niveau pair = NivPair Faire :
            Si ArcP.Extrémité =NodeAct alors :
                ArcP.Extrémité :=NewNoeud ; // Extrémité = Fnode / Tnode
            Fin Si
        Fin Pour ;
    Fin Si
    Fin Si
    Fin Pour
    Fin Si
    Fin Pour
    Fin Si
    FIN.

```

---

#### 4.9. Mise en œuvre des passerelles 2D>3D pour les bâtiments (structures surfaciques)

L'implémentation a été réalisée avec le langage de programmation python. L'application réalisée se compose d'un ensemble d'extensions intégrées dans le logiciel QGIS. Le module Python *Psycopg2* est utilisé pour intégrer PostgreSQL/POSTGIS avec python. Ce module représente un adaptateur de base de données de PostgreSQL/POSTGIS pour le langage de programmation Python. Les passerelles 2D vers 3D représentent un processus interactif avec l'utilisateur pour construire la base de données topologique en 3D. Ce processus passe par différentes phases :

##### 4.9.1. Importation et vectorisation

Nous avons utilisé les plans des niveaux d'un bâtiment en 2D comme source de données. Les plans de niveaux sont importés à partir des images scannés ou à partir des formats standards DXF. La digitalisation permet de créer la couche vectorielle en modèle spaghetti (structure linéaire) sans topologie. Nous avons utilisé les outils des logiciels SIG comme Mapinfo dans cette étape. Les vecteurs linéaires sont exportés vers des fichiers de format mif.



**Figure 4.13.** Exemples des Plans de niveaux

### 4.9.2. Génération de topologie 2D

L’outil Traducteur Universel de mapinfo permet de convertir la couche vecteur en modèle spaghetti (modèle utilisé par MapInfo) à un autre format qui gère la topologie par exemple le format E00 (Esri ArcInfo), la topologie 2D est créée dans ce format. Ce dernier doit être reconverti dans un format lisible par mapinfo. Ce processus de conversion/reconversion peut provoquer un décalage des coordonnées de la base de données initiale. Nous avons utilisé PostgreSQL/POSTGIS qui possède un outil POSTGIS Shapefile Import/Export. Cet outil permet d’importer les fichiers Shapefile à une base de données *POSTGIS* mais dans un schéma *public* qui ne génère pas la topologie. Il est nécessaire donc d’utiliser le module *Topology* qui offre beaucoup de fonctions pour la gestion et l’insertion dans des tables topologiques 2D. Nous avons utilisé ces fonctions pour créer l’extension *CreateTopo2D*, elle sert à importer un fichier de format d’échange de *mapinfo* (.mif) contenant des lignes (modèle spaghetti) dans un schéma topologique sous *POSTGIS*. Un schéma topologique est créé pour chaque plan de niveau. Chaque ligne est intégrée dans la topologie comme un ou plusieurs arcs dans la table à l’aide de la fonction topologique *addege*, suivie de la création de tous les polygones possibles à partir des arcs. Les fonctions *AddFace* et *st\_polygonize* sont utilisées pour la création des faces. Les schémas résultants contiennent les 3 tables de primitives *Face(Face)*, *Edge(arc)*, *Node(noeud)* avec les relations topologiques entre elles ( Figure4.14).

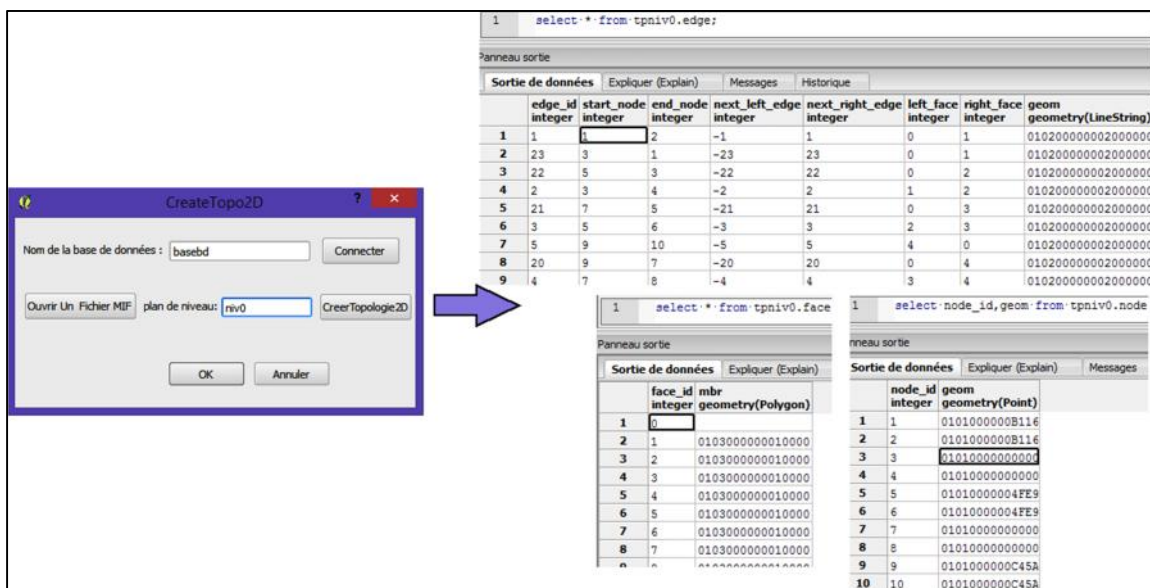


Figure 4.14.Création de Topologie 2D

### 4.9.3. Création et paramétrage des objets (entités sémantiques)

Des extensions QGIS sont créées (Figure 4.15) pour cette phase pour gérer l'aspect sémantique et elles permettent de:

- Créer les différentes tables sémantiques comme :

---

*Mur* ( *Id\_mur*, *ValNiv-Inf*, *ValNiv-Sup*, *geom topogeometry*) // *NivInf/NivSup* : *Valniv* pair de niveau inférieur/supérieur

*Plancher* (*Id\_Plancher*, *ValNiv*, *mbr topogeometry*) // *Valniv=2n* (valeur pair)

*Escalier* (*Id\_Escalier*, *ValNiv*, *geom topogeometry*) // *Valniv=2n+1* (valeur impair)

---

Le type *topogeometry* est utilisé pour la composition d'un objet à partir des primitives topologiques. Par exemple, la topo-géométrie est utilisée pour créer un mur à partir des arcs, l'escalier à partir des faces et ainsi de suite. La fonction topologique *AddTopoGeometryColumn* est utilisée pour créer une colonne de type *TopoGeometry* dans les tables.

- Création et insertion des entités dans la table à partir des primitives topologiques selon le type correspondant :

La fonction topologique *CreateTopoGeom* permet la création d'un élément *TopoGeometry* à partir des primitives topologiques. Les identificateurs des primitives sont ajoutés à partir des éléments sélectionnés par l'utilisateur sur le visionneur de la base de données spatiale.

La figure 4.15 représente un exemple (création mur) de l'interface graphique d'une des extensions réalisées pour cette phase. Le bouton Sélection sert à afficher la couche de primitive topologique adéquate pour sélectionner les primitives qui composent un élément topo géométrique représentant l'objet sémantique.

Le programme active automatiquement la couche voulue si elle existe, sinon, il charge la table adéquate comme une nouvelle couche à partir de la base de données. Le champs *ValNiv* permet d'introduire les valeurs de niveau correspond à l'objet. Une valeur *ValNiv* s'affiche par défaut qui représente le niveau correspond au schéma topologique envisagé.

L'utilisateur change la valeur de niveau s'il est nécessaire dans le cas où plusieurs niveaux ont la même architecture, autrement les différents niveaux ont le même nom sauf le dernier caractère qui correspond au numéro d'ordre du niveau par exemple ('niv0', 'niv1', 'niv2' ... 'nivn') ce qui donne les valeurs par défaut de niveau pair (0,2,4,.....2\*int('n')), *ValNiv* impair (1,3,5,.....2\*int('n')+1), *ValNiv* pair suivant (2,4,6,.....2\*int('n')+2).

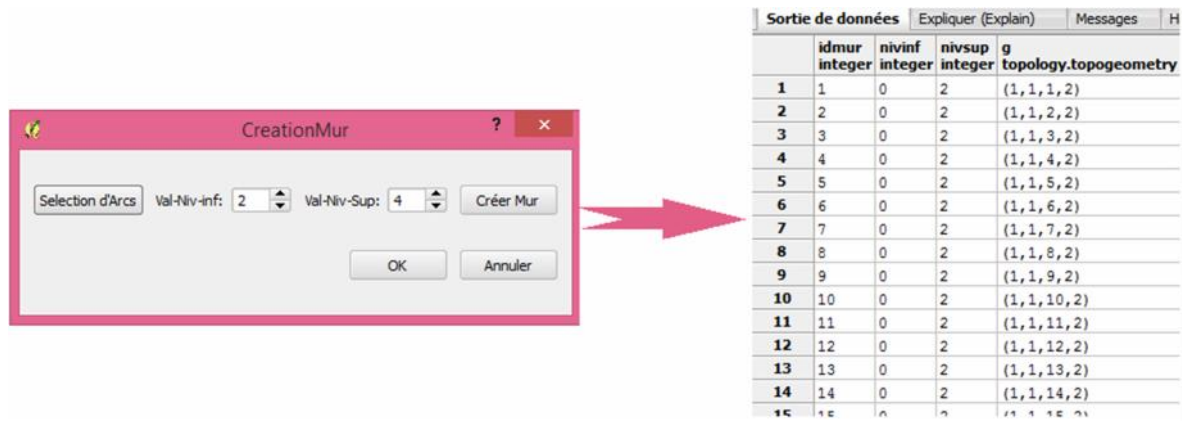


Figure 4.15. Création de table sémantique (exemple : mur)

Le bouton *créer* sert à créer la table si elle n'est pas encore créée sinon il insère les valeurs des niveaux et les éléments sélectionnés comme une nouvelle ligne dans la table adéquate.

**4.9.4. Ajout de détails sémantiques :**

Après la création des entités sémantiques, l'extension *ajout détails* permet la création de table des composants sémantiques des objets (volée d'escalier). Les primitives sélectionnées doivent appartenir géométriquement à un des objets de niveau supérieur dans la hiérarchie. L'objet agrégat est détecté à l'aide de la fonction géométrique Contient (~) entre les deux géométries de la face sélectionnée et l'objet escalier. L'élément de détail de sous-niveau impair correspond au volé d'escalier et l'élément de détail de sous-niveau pair correspond au palier d'escalier voir figure 4.16. La table volée créée : *Voleesc(idvolee, idesc, valsniv, geom topogeometry)*

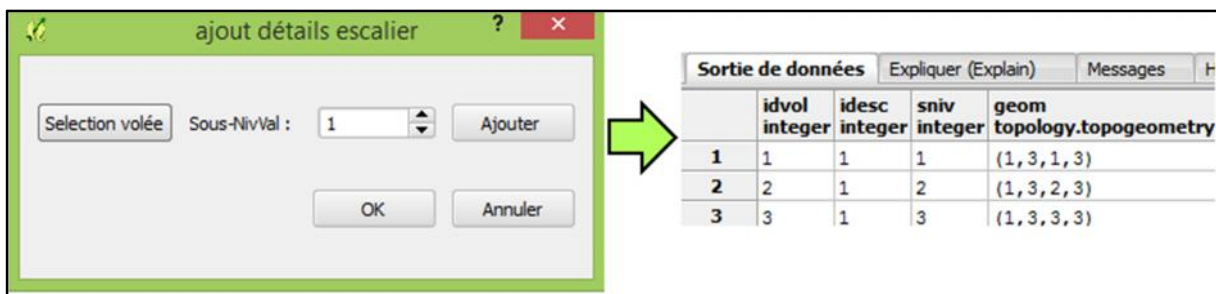


Figure 4.16. Ajout de détails sémantiques (volé d'escalier)

L'utilisation de sous-niveau pour décrire la hiérarchie de détails permet l'ajout des détails de façon cohérente entre les deux niveaux de détail.

#### 4.9.5. Construction topologique 2D vers 3D

Une extension *CréationTableTopo3D* a été réalisée pour permettre la création des différentes tables Topologiques 3D. Les tables topologiques 3D pour les primitives topologiques qui ont des *ValNiv* pairs comme face-N(Face), et Arc-n (edge) sont créées dans des schémas topologiques 2D pour chaque valeur de niveau pair. Ces différents schémas représentent des tables intermédiaires de calcul qui n'ont pas un intérêt dans la base de données finale et ne sont pas visibles au niveau de l'utilisateur final. Elles sont utilisées au niveau du passage vers le 3D pour la génération automatique de la topologie 2D entre les différentes primitives qui ont le même niveau pair lors de la reconstruction topologique en 3D (de même pour les primitives 'détails' de sous-niveau pairs). La topologie entre ces primitives est équivalente à celle en 2D. De ce fait, un ensemble de tables est créé dans cette phase (pour les attributs avec préfixe '#' représentant des clés étrangères, les clés primaires sont soulignées) :

---

```

Volume (Volume_id, valniv, geom3d geometry(polyhedralsurfaceZ)) // polyhedralsurfaceZ est un nouveau type qui
représente la géométrie volumique dans Postgis.
Face(face_id, valniv, mbr geometry(polygon), mbr3d geometry(polygonZ), #volume_inf, #volume_sup)
Edge(edge_id, valniv, #start_node, #end_node, #left_face, #right_face, geom geometry(linestring),
geom3d geometry(linestringZ))
Node(node_id, valniv, geom geometry(point), geom3d geometry(pointZ))
LigneN(id_ligne, valniv, geom topogeometry(multilinestring), geom3d geometry(linestringz))
Edgeim(edgeim_id, valniv, #start_node, #end_node, geom3d geometry(linestringZ))
Faceim(faceim_id, valniv, # start_edge, #end_edge, #inf_ligne, #sup_ligne, #right_volume, #left_volume ,
geom3d geometry(polygonz))

```

---

Les tables sémantiques en 3D sont créées pendant le processus de construction topologique 3D pour chaque entité.

La méthode de reconstruction topologique 3D associée aux différents objets pour les structures surfaciques se fait selon les 3 possibilités de représentation d'un objet surfacique en 2D.

*Algorithme de Construction Topologique 3D dans chaque cas des objets surfaciques :*

- La fonction topologique *AjouterTopoLigne* permet l'insertion d'une géométrie linéaire dans une topologie et Retourne l'ensemble des Identificateurs des Arcs résultants.
- La fonction *Composer\_ligne* permet la création d'une ligne à partir d'un ensemble de points ou de géométrie de type linéaire.
- La fonction *AjouterTopoPolygone* : Insertion d'une géométrie surfacique dans une topologie et Retourne l'ensemble des Identificateurs des faces résultantes.

*InsèreTpLigne* : Insérer Ligne-N dans une topologie et retourne ligne\_id.

*Insère TpEdgeIM* : Insérer edge-IM dans la topologie et retourne edgeim\_id.

*InsèreTPFaceIM* : insérer face\_IM dans la topologie et retourne faceim\_id.

---

## 1/ Objets surfaces avec un niveau pair :

**Algorithme 4.2.** Reconstruction Topologique 3D d'un objet surfacique de niveau pair

---

**ConstrTopo3D ( OjbetSurface Table, valniv = 2n)#exemple :Entité Plancher**

**ENTREE** : Table Objet2D, Tables topologie 2D, Tables topologie 3D

**SORTIE** :Table Objet3D, Mise à jour des Tables Topologie 3D

**DEBUT** :

*Créer Table objet3D(idobj, valniv)# Création de table 3D de ce type d'objet*

*AjoutTopoGéométrieColonne(valniv, objet3D,mbr geom ,polygon)*

*Pour chaque objet surfacique 2D dans OjbetSurface avec valniv = ValnivF Faire :*

*ExtraireGeometrieSurface(mbr topogeometrie) : Rendre Geometry FaceN(polygon)*

*AjouterTpPolygone(FaceN, valniv,Table3dnivn.FaceN) : Rendre ensemble de face\_id (Ens fid)*

*Insérer à objet3d(valniv,mbr) les valeurs :( ValnivF,CréerTopoGéométrie(Ens fid))*

*FinPour*

**FIN**

---

## 2/ Face impair en position orthogonale (exemple : les murs)

Reconstruction topologique à partir d'une représentation 2D de type ligne et les valeurs de niveau pair inférieur et supérieur :

**Algorithme 4.3.** Reconstruction Topologique 3D d'un objet surfacique de niveau impair orthogonale

**ConstrTopo3D (Objet-ligne Table, Valniv-inf = 2n, Valniv-sup=2m) :**

**ENTREE :** Table Objet2d, Table topologie 2D, Tables topologie 3D

**SORTIE :** Table3Dobj Table, mise à jour des tables Topologiques 3D.

**DEBUT :**

Créer Table3dobj(idobj, valniv =2n+1,#faceim\_id,Geom3D polygonZ)

Pour chaque element d'Objet-ligne faire :

Val-niv=valnivinf+1

rendre Fid =max (faceim\_id) // nouvelle FaceIm

Insérer à Table3Dobj (valniv,faceim\_id) les valeur(valnivinf+1,Fid)

ExtraireGeometryLigne(topogeometry) : Rendre GeometryLigne

AjouterTopoLigne(edge,GeometryLigne,Valnivinf) : Rendre ensemble Ens edge\_id

InsèreTpLigne à LigneN(valniv,geom) valeurs(valnivinf, CréerTopoGéometrie(Ens edge\_id): Rendre ligneI=ligne\_id

AjouterTopoLigne(edge,GeometryLigne,Valnivsup) : Rendre ensemble Ens edge\_id

InsèreTpLigne à LigneN(valniv,geom) valeurs(valnivsup, CréerTopoGéometrie(Ens edge\_id) : Rendre ligneS=ligne\_id

Rendre SNode=Node\_id de node où le nœud est debut/fin de ligneI

Rendre FNode=node\_id de node où le nœud est debut/fin de ligneS

InsèreTpEdgeIM à edgeim(valniv,start\_nod,end\_node) valeurs( Val-niv,SNode,FNode): Rendre edgeimS/E= edgeim\_id

InsèreTpFaceim FaceIm(valniv,... sup\_ligne)Valeurs(Val-niv, edgeimS,edgeimE,ligneI,ligneS) :Rendre Fid=Faceim\_id

Insérer à Table3dobj(valniv,faceim\_id) les valeur(val-niv,Fid)

FinPour ;

**FIN.**

**3/ Faces qui ont une position inclinée en 3D (exemple : Escalier)**

Reconstruction topologique 3D à partir d'un objet surfacique et la détermination des arcs des niveaux pairs inférieur/supérieur

**Algorithme 4.4.** Reconstruction topologique 3D d'un objet surfacique de niveau impair incliné

**ConstrTopo3D ( element de Objet-surface Table, val-niv = 2n+1)**

**ENTREE :** Table Objet2D, Table topologie 2D, Tables topologie 3D

**SORTIE :** Table3Dobj TABLE, mise à jour des tables topologiques 3D

**DEBUT :**

Créer Table3dobj(idobj, valniv,#faceim\_id,Geom3D polygonZ) // Valniv =2n+1

Pour chaque element d'Objet\_surface faire :

Lire (Ens\_edge(Inf/sup)={edge\_id/ID des arcs de ligne selectionnée},ValnivI/ValnivS) Valniv pair : inf 2m=2n/ sup 2m m>n

Composer\_ligne(Ens{geom/geometry où edge\_id dans Ens\_edge(Inf/Sup)}) : Rendre GeometryLigne

AjouterTopoLigne(edge,GeometryLigne,ValnivI/ValniS) : Rendre ensemble Ens edge\_id

InsèreTpLigne LigneN(valniv,geom)valeurs(valnivI/S, CréerTopoGéometrie(Ens edge\_id)) : Rendre ligneI/S= ligne\_id

Rendre SNode\_id de node où le nœud est debut/fin de ligne\_id= ligneI

Rendre FNode\_id de node où le nœud est debut/fin de ligne qui a ligne\_id= ligneS

InsèreTpEdgeIM à edgeim(valniv,start\_node,end\_node) valeurs( 2n+1,SNode,FNode) : Rendre edgeimS/E=edgeim\_id

InsèreTpFaceim FaceIm(valniv,... sup\_ligne)Valeurs(Val-niv, edgeimS,edgeimF,ligneI,ligneS) :Rendre Fid=Faceim\_id

Insérer à Table3Dobj (valniv,faceim\_id) les valeur(val-niv,Fid)

Fin Pour

**FIN.**

Dans ce dernier algorithme, l'objet sémantique adéquat est l'escalier. Pour donner plus de précision à l'algorithme par la reconstruction de la topologie des objets sémantiques en détails (volée, palier) des lignes sont ajoutées à l'algorithme précédent comme suit :

### Algorithmes 4.5. Reconstruction topologique en 3D d'escalier en détails

#### **ConstrTopo3dEscalierDetails ( ) :**

**ENTREE :** Table 2D : Voleesc(idvolee, idesc, valsniv, geom topogeometry), Escalier(Id\_Escalier, valniv, geom topogeometry), Tables topologie 2D, Tables topologie 3D

**SORTIE :** Table Escalier3d, Voleesc3d, Palier3d, mise à jour des tables topologiques 3D (détaillés : niveau, sous-niveau)

#### **DEBUT**

Créer Escalier3D (id\_Escalier, valniv, #faceim\_id, Geom3D polygonZ) // Valniv = 2n+1, Algorithme 4.4

Créer Table Voleesc3D (idvolee, #idesc, Valsniv, #faceim, Geom3D polygonZ) // si Valsniv=2n pour type volee

Créer Table Palier3D (idpalier, idesc, Valsniv, Geom topogeometry, Geom3D polygonZ) // Valsniv=2n+1 pour palier.

Pour chaque element esc de table Escalier Faire : // Escalier Correspond à Objet\_surface dans l'algorithme 4.4

Le même code que l'Algorithme 4.4 pour ajouter un Escalier3d sans détails

Rendre Id\_escalier=IdEsc3, Valniv, Faceimid3=faceim\_id de Escalier3d // Valniv=2m+1

Rendre max= max(Valsniv) de Voleesc où idesc=esc. id\_Escalier

Pour chaque element volee de Voleesc ou idesc=esc.id\_Escalier Faire :

Si volee.Valsniv est pair (Valsniv mod 2=0) Alors : // Valsniv=2n

ExtraireGeometrieSurface(volee.geom topogeometry) : Rendre Geometry FaceN(polygon)

AjouterTopoPolygone(FaceN, Valsniv, Table3dnivmn.FaceN) : Rendre ensemble de face\_id (Ens fid)

Insérer à palier3d (valsniv, geom) valeurs ( valnivF, CréerTopoGéometrie(Ens fid))

Sinon Si Valsniv=1/Valsniv=max Alors // volee.Valsniv est impair (Valsniv mod 2 <> 0) Alors : // Valsniv=2n+1

Select Linf/Lsup=inf\_ligne/sup\_ligne de faceim où faceim\_id=faceimid3

Actualiser FaceIm affecter a. inf\_ligne /sup\_ligne = Linf/Lsup où faceim\_id=faceimid

Rendre Snode/Fnode=node\_id où node\_id est Linf.debut/Lsup.debut

InsèreTpEdgeIM à edgeim(valniv, vaslniv, start\_node, end\_node) valeurs(esc. valniv, volee.Valsniv, Snode, Fnode) et retourne arc-start=edgeim\_id

Rendre Snode/ Fnode=node\_id où node\_id est Linf.fin/Lsup.fin

InsèreTpEdgeIM à edgeim(valniv, vaslniv, start\_node, end\_node) valeurs(esc. valniv, volee.Valsniv, Snode, Fnode) et retourne arc-end=edgeim\_id

InsèreTpFaceim FaceIm(valniv, valsniv.. sup\_ligne)Valeurs(esc. valniv, volee.Valsniv, arc-start, arc-end Linf, Lsup) rendre faceid=faceim\_id

Insérer à Voleesc3d (idesc, Valsniv, faceim) valeurs(esc.Id\_Escalier, volee.Valsniv, faceid)

Sinon :

geom=intersect(volee.geom, Voleesc.geom) où Voleesc. idesc=id\_Escalier et Valsniv=volee.Valsniv -1/Valsniv +1

AjouterTopoLigne(edge, geom, Valsniv) : Rendre ensemble Ens edge\_id

InsèreTpLigne LigneN(valsniv, geom)valeurs(valsniv, CréerTopoGéometrie(Ens edge\_id)) : Rendre Linf/Lsup=ligne\_id

Actualiser FaceIm affecter inf\_ligne/sup\_ligne = Linf/Lsup où faceim\_id=faceimid

Rendre Snode/Fnode=node\_id où node\_id est Linf.debut/Lsup.debut

InsèreTpEdgeIM à edgeim(valniv, vaslniv, start\_node, end\_node) valeurs(esc. valniv, volee.Valsniv, Snode, Fnode) et retourne arc-start=edgeim\_id

Rendre Snode/Fnode= node\_id où node\_id est Linf.fin/Lsup.fin

InsèreTpEdgeIM à edgeim(valniv, vaslniv, start\_node, end\_node) valeurs(esc. valniv, volee.Valsniv, Snode, Fnode) et retourne arc-end=edgeim\_id

InsèreTpFaceim FaceIm(valniv, valsniv, start\_edge, end\_edge, inf\_ligne, sup\_ligne)Valeurs(esc. valniv, volee.Valsniv, arc-start, arc-end Linf, Lsup) rendre faceid=faceim\_id

Insérer à Voleesc3d (idesc, Valsniv, faceim) valeurs(esc.Id\_Escalier, volee.Valsniv, faceid)

Finsi

Finsi

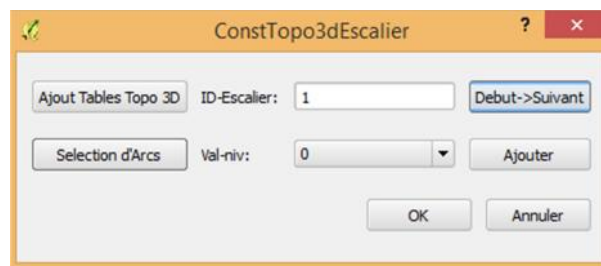
*Finpour*

*Finpour*

**FIN.**

'//': Pour un commentaire. '/' : Répéter le même code avec différents paramètres.

La figure 4.17 est une fenêtre qui représente l'extension *ConstTopo3DEscalier* permettant la reconstruction de la topologie en 3D de l'escalier. Le bouton de sélection et le champ *ValNiv* sont utilisés pour la détermination des arcs de niveau inférieur/supérieur de chaque objet surfacique (escalier).



**Figure 4.17.** Construction Topologique 3D

Le résultat de cette phase est une base de données qui contient les relations topologiques entre les différentes primitives (*Face-im*, *Face*, *Arc*, *Arc-im*, *Nœud*) en 3D bien que la géométrie soit en 2D. Ce qui manque, dans cette phase, c'est la géométrie en 3D (l'ajout de coordonnée Z) ainsi que la création des objets volumes et les différentes relations topologiques entre les objets volumiques et les primitives de type surfacique (*Face-n*, *Face-im*).

#### 4.9.6. Création de la géométrie Z

Le modèle proposé dans ce travail est un modèle topologique qui décrit les relations topologiques entre les différentes primitives en 3D. Ce passage topologique de 2D vers 3D est garanti sans l'utilisation de la géométrie selon l'axe z, seulement les valeurs de niveaux (sous-niveau) sont utilisées pour décrire la topologie en vertical. Cette notion de niveau (sous-niveau) est exploitée aussi sur le plan de coupe verticale pour l'ajout des coordonnées selon l'axe Z.

Le plan de coupe verticale est stocké dans un schéma topologique. La Table Ligne-N est créée dans ce schéma *LigneN(idligne, valniv, vals niv, geom topogéométrie(line))*.

L'extension *Création Ligne-n* offre la possibilité d'insérer les arcs qui ont la même valeur de niveau (sous-niveau) comme une ligne-n dans la table.

La distance entre un nœud qui appartient à une ligne-n de *ValNiv* et le même point (X) sur la ligne-n de *ValNiv* 0 donne la hauteur Z de ce point par rapport au niveau 0. La coordonnée y sur la coupe verticale correspond à l'axe Z et les distances y sont donc ajoutées par la suite comme coordonnées Z à la géométrie 2D pour obtenir la géométrie 3D des nœuds.



**Algorithme 4.6.** Construction topologique des volumes

**Construction Topo Volume** (Table obj-surface(idobj, valniv, mbr topogeometry, mbr3d geom (polygonz))):

**ENTREE** : Tables Topologiques 3D, Table volume (vide)

**SORTIE** : Mise à jour des Tables Topologiques 3d(Ajout topologie volume), Table Volume (Base de données topologique 3D complète)

**DEBUT** :

Créer table plafond (id-plaf, valniv, mbr topogeometry, mbr3d geometry(polygonz))

Insérer à Volume (volume\_id=0) // représente le volume globale ou l'extérieur

Rendre MaxNiv=max(valniv) dans la table des face // la valeur max des niveaux des faces

Rendre MinNiv=min(valniv) dans la table des face // la valeur min des niveaux des faces

Actualiser Face affecter à volume-inf, volume\_sup les valeurs (0,0) où face\_id=0

Actualiser Face affecter à volum-inf valeur 0 où Valniv=MiniNiv

Actualiser Face affecter à volum-sup valeur 0 où Valniv=MaxNiv

Pour chaque élément pl de table objet-surf // la table plancher

Insérer à Volume(valniv =pl. valniv +1)

PlafEnsF=[ ] //Ensemble vide des Face\_id qui compose un objet-surf supérieur(plafond)

Rendre Vol=max(volume\_Id) de Volume // l'identificateur de dernier volume inséré

Rendre VolGeom= pl.mbr3d de objet-surf où idobj=pl.id-obj // idobj ou id\_plancher

Rendre les id(s) ElementTopoGeom(pl.mbr) // obtenir les ids des faces de la topogéométrie

Pour chaque face d'elements id(s) : Update face affecter volume\_sup=Vol où face\_id=id Finpour

Rendre l'ensemble des ligne-n enligne qui sont contenue dans le contour de pl et qui représentent ligne\_inf pour les Face-im qui représentent des obj-surf(murs)

Pour chaque ligne dans enligne faire :

Rendre faceim=faceim\_id de faceim où ligne=inf\_ligne et faceim dans mur.faceim

Rendre line de table ligne-n où faceim.sup\_ligne=line(ligne\_id)

Rendre valnivS=line.valniv

Rendre S-id les id(s) ElementTopoGeom(line) // les ids des ligne-n supérieur de face-im

Rendre les id(s) ElementTopoGeom(ligne) // obtenir les ids des faces de la topogéométrie

Rendre Arc où Arc est dans ids et arc.debut=ligne.debut

Si arc.left\_face/right\_face est contenue dans pl Alors:

Actualiser faceim Affecter à left\_volume/right\_volume valeur Vol où faceim.inf\_ligne=ligne

Pour chaque arcc dans s- id faire:

Actualiser Face affecter à face.volume-inf=Vol où face\_id=arcc.left\_face/right\_face et volume\_inf est null

Ajouter face-id à PlafEnsF

Fin pour

Finsi

Pour chaque id(s) faire :

Si id.left\_face/right\_face=0 alors :

Actualiser faceim Affecter à left\_volume/right\_volume valeur 0 où faceim.inf\_ligne=ligne

Finsi

Finpour

Ajouter faceim.geom3d à VolGeom

FinPour

Insérer à plafond(valniv,geom) valeurs (valnivS, CréerTopoGéométrie(PlafEnsF)) où id-plaf

Insérer à Volume(geom3d) valeur(VolGeom) où volume\_id=Vol //VolGeom : polyhedralsurfacez

Finpour

Pour chaque faceim :

Rendre vol=volume\_sup de right\_face/left\_face par rapport à ligne\_inf de faceim

```

Actualiser faceim affecter à left_volume/right_volume la valeur vol
FinPour
Pour Chaque faceim où Valsniv=max(Valsniv) faire : // Volée d'escalier relié le palier d'arrivée
    Rendre vol=volume_inf de right_face/left_face par rapport à ligne_sup de faceim
    Actualiser faceim affecter à left_volume/right_volume la valeur vol
Fin pour
Pour chaque face de Valsniv<>0 Faire : // Face de sous-niveau pair (palier Intermédiaire d'escalier)
    Rendre vol=left_volume/right_volume de faceim où face est left_face /right_face des arcs de ligne_inf de faceim
    Actualiser face affecter à volume_sup la valeur vol
    Rendre vol=left_volume/right_volume de faceim où face est left_face /right_face des arcs de ligne_sup de faceim
    Actualiser face affecter à volume_inf la valeur vol
FinPour

```

FIN.

L'extension réalisée à la base de cet algorithme permet la création des objets de type volume (polyhedralsurfaceZ) et de compléter la topologie 3D par l'ajout des champs Volume Gauche/ Droit pour les face-im (Face de niveau impair ou intermédiaire) et Volume Inf/Sup pour les faces-N (face de niveau pair).

face_id integer	volume_inf integer	volume_sup integer	mbr3d geometry(PolygonZ)
1	2	0	01030000800100000
2	3	0	01030000800100000
3	5	0	01030000800100000
4	4	0	01030000800100000
5	0	0	
6	1	0	01030000800100000
7	18	8	01030000800100000
8	19	9	01030000800100000
9	23	9	01030000800100000
10	9	12	01030000800100000
11	21	12	01030000800100000
12	22	0	01030000800100000
13	6	8	01030000800100000
14	12	8	01030000800100000
15	13	8	01030000800100000
16	15	8	01030000800100000
17	20	8	01030000800100000
18	7	9	01030000800100000
19	16	9	01030000800100000
20	14	10	01030000800100000
21	8	10	01030000800100000

faceim_id integer	valniv integer	end_edge integer	sup_edge integer	inf_edge integer	right_volume integer	left_volume integer	geom3d geometry(PolygonZ)
1	2	3	4	2	19	1	01030000800100000
2	3	3	6	3	20	1	01030000800100000
3	24	3	48	24	41	3	01030000800100000
4	26	3	52	26	43	3	01030000800100000
5	23	3	46	23	40	0	01030000800100000
6	18	3	36	18	35	5	01030000800100000
7	19	3	38	19	36	5	01030000800100000
8	20	3	40	20	37	7	01030000800100000
9	7	3	14	7	24	4	01030000800100000
10	14	3	28	14	31	2	01030000800100000
11	4	3	8	4	21	1	01030000800100000
12	15	3	30	15	32	2	01030000800100000
13	6	3	12	6	23	1	01030000800100000
14	11	3	22	11	28	6	01030000800100000
15	12	3	24	12	29	6	01030000800100000
16	8	3	16	8	25	4	01030000800100000
17	22	3	44	22	15	7	01030000800100000
18	5	3	10	5	22	1	01030000800100000
19	16	3	32	16	33	7	01030000800100000

idvolume integer	valniv integer	geom text
1	0	
2	1	3 POLYHEDRALSURFACE Z (((403.00001 913 379.99999,248.00001 913 379.99999,248.00001 1246 379.99999,403.00001 12
3	2	3 POLYHEDRALSURFACE Z (((819.00001 1246 379.99999,1069.00001 1246 379.99999,1069.00001 1023 379.99999,819.0000
4	3	3 POLYHEDRALSURFACE Z (((1188.99999 1246 379.99999,1546 1246 379.99999,1546 913 379.99999,1188.99999 913 379.9
5	4	3 POLYHEDRALSURFACE Z (((819.00001 872 379.99999,819.00001 569 379.99999,677 569 379.99999,403.00001 569 379.9
6	5	3 POLYHEDRALSURFACE Z (((1188.99999 551.99999 379.99999,1019.00001 551.99999 379.99999,1019.00001 872 379.9999
7	6	3 POLYHEDRALSURFACE Z (((677 327 379.99999,403.00001 327 379.99999,248.00001 327 379.99999,248.00001 569 379.9
8	7	3 POLYHEDRALSURFACE Z (((1380 327 379.99999,1188.99999 327 379.99999,1095.99999 327 379.99999,687 327 379.9999
9	8	1 POLYHEDRALSURFACE Z (((403.00001 872 0,403.00001 1246 0,687 1246 0,1188.99999 1246 0,1188.99999 872 0,687 87
10	9	1 POLYHEDRALSURFACE Z (((403.00001 551.99999 0,403.00001 872 0,687 872 0,1188.99999 872 0,1188.99999 551.99999
11	10	1 POLYHEDRALSURFACE Z (((1188.99999 872 0,1188.99999 1246 0,1546 1246 0,1546 872 0,1546 551.99999 0,1306 551.9
12	11	1 POLYHEDRALSURFACE Z (((1760.00001 551.99999 0,1546 551.99999 0,1546 872 0,1546 1246 0,1760.00001 1246 0,1760
13	12	1 POLYHEDRALSURFACE Z (((403.00001 327 0,403.00001 551.99999 0,687 551.99999 0,1188.99999 551.99999 0,1306 551
14	13	7 POLYHEDRALSURFACE Z (((248.00001 1246 996,690 1246 996,690 1112 996,690 902 996,248.00001 902 996,248.00001
15	14	7 POLYHEDRALSURFACE Z (((718.00001 1112 996,690 1112 996,690 1246 996,718.00001 1246 996,840 1246 996,840 1112
16	15	7 POLYHEDRALSURFACE Z (((857.99999 1004 996,840 1004 996,840 1112 996,840 1246 996,857.99999 1246 996,1045.999
17	16	7 POLYHEDRALSURFACE Z (((1087 902 996,1045.99999 902 996,1045.99999 1004 996,1045.99999 1246 996,1087 1246 996
18	17	7 POLYHEDRALSURFACE Z (((1546 902 996,1546 551.99999 996,1380 551.99999 996,1115.99999 551.99999 996,1115.9999
19	18	7 POLYHEDRALSURFACE Z (((784 00001 802 802 784 00001 510 802 787 80000 510 802 748 80001 510 802 748 80001 802

Figure 4.19. Topologie 3D VOLUME-FACE (face-n et face-im)

La figure 4.19 représente le résultat d'une requête de sélection sur les 3 Tables Volume, Face et FaceIM pour montrer les relations topologiques entre elles ainsi que la géométrie 3D de chaque type

d'objet. Pour la table volume la géométrie est en format WKT ce qui montre son type géométrique polyhedralsurfacez.

#### 4.9.8. Visualisation et interrogation topologique en 3D

La visualisation 3D n'est pas l'objectif principal de ce travail. Nous avons utilisé les outils de visualisation 3D existants pour montrer les résultats des requêtes d'analyse sur la base de données topologique 3D.

L'outil Qgis 2threejs offre la possibilité d'affichage 3D. Ce plugin QGIS permet de visualiser des données en 3D dans le navigateur, grâce à WebGL via Three.js. Les tables géométriques 2D peuvent être affichées en 3D par l'extrusion selon un attribut z qui représente la hauteur.

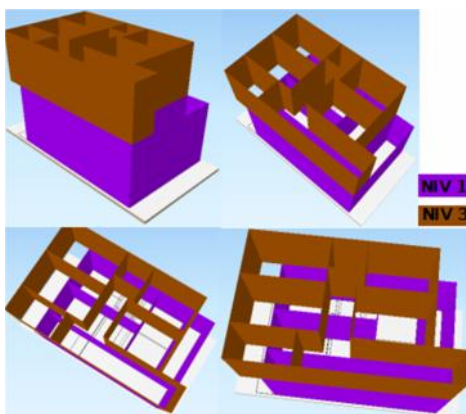
Les figures 4.20 et 4.21 représentent la visualisation 3D des murs de la base de données topologiques groupés selon leurs valeurs de niveaux et avec une couleur différente pour chaque niveau. La figure 4.20 correspond à la requête :

```
SELECT geom3d, valniv FROM mur3d WHERE valniv=3 OR valniv=1
```

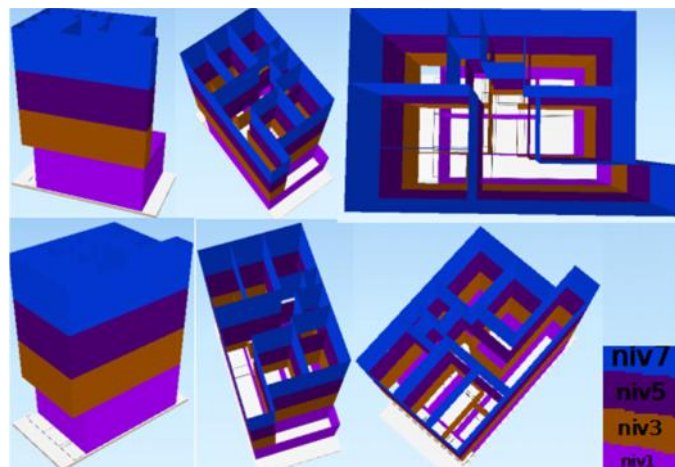
La figure 4.21 représente les murs de tous les niveaux :

```
SELECT geom3d, valniv FROM mur3D
```

Pour pouvoir visualiser en 3D les entités contenues dans la base, il n'existe actuellement que très peu d'outils. Pour visualiser automatiquement le résultat d'une requête qui porte sur la colonne géométrique 3D nous avons utilisé le format X3D (eXtensible 3D) qui est un standard ISO pour la représentation 3D dans un navigateur web, et le successeur de VRML. Le résultat d'une requête est utilisé pour construire un fichier X3D. Ce fichier est affiché avec un navigateur qui supporte ce format.



**Figure 4.20.** Visualisation 3D des murs des niveaux 1 & 3



**Figure 4.21.** Visualisation 3D des murs par niveau

La figure 4.22 (a et b) représente l’affichage du fichier généré par le résultat de la requête qui retourne toutes les faces internes représentées en deux niveaux de détails:

```
SELECT mbr3d FROM face WHERE volume_sup <> 0 AND volume_inf<>0 UNION
SELECT geom3d FROM faceim WHERE rihtg_volume <> 0 AND left_volume <> 0
```

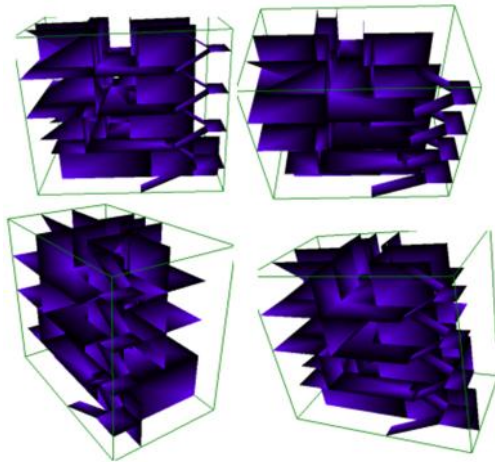


Figure 4.22. (a) : Faces d’intérieur

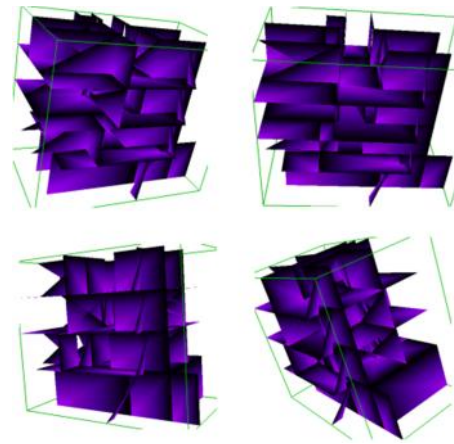


Figure 4.22. (b) : Faces d’intérieur

La figure 4.23 représente tous les faces externes de bâtiment qui correspond à la requête :

```
SELECT mbr3d FROM face WHERE volume_sup=0 OR volume_inf=0 UNION
SELECT geom3d FROM faceim WHERE rihtg_volume=0 OR left_volume=0
```

La figure 4.24 représente tous les faces qui sont reliées à un niveau donné x (par exemple la valeur de niveau d’une face sélectionnée en 2D) :

```
SELECT mbr3d FROM face WHERE valniv=x UNION
```

```
SELECT geom3d FROM faceim JOIN lignen ON (faceim.inf_ligne=lignen.ligne_id OR
faceim.sup_ligne=lignen.ligne_id) AND lignen.valniv=x // x=2n
```

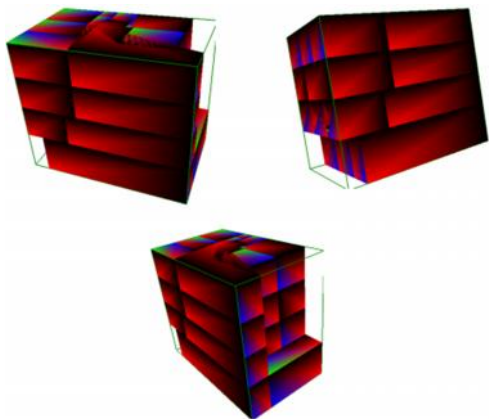


Figure 4.23. Faces d’extérieur

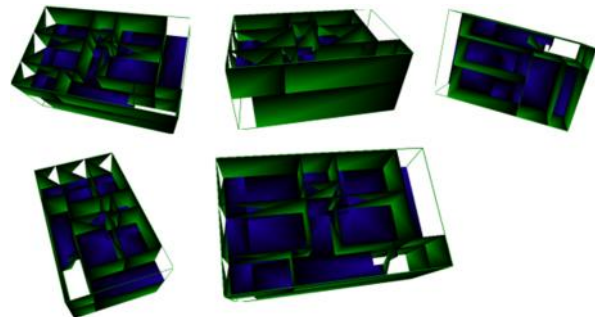


Figure4.24. Faces liées à un niveau pair donné

## **Conclusion**

Le concept de niveau utilisé dans le présent modèle permet d'étendre un modèle 2D pour représenter les objets dans un modèle topologique 3D de façon simple. En effet, ce modèle montre bien que nous gardons toujours toutes les primitives topologiques 2D. Nous avons complété l'ensemble du système par les primitives qui sont liées au 3D à savoir les notions de *niveau*, *volume* et *face*. Le processus de passage vers le 3D avec ce modèle exploite les valeurs de niveau associés aux différents objets, ainsi que les attributs sémantiques qui permettent de définir la méthode de reconstruction topologique 3D appropriée. Le concept de niveau de détail a été intégré par l'ajout des sous niveaux topologiques pour décrire plus de détails.

Au niveau physique, le modèle a été implémenté avec les outils SIG existants en 2D et les moyens de stockage et de manipulation topologique 2D en utilisant le SGBD-S PostGeSQL/PostGis. La base de données résultante est une base de données Topologique 3D, permettant l'analyse d'une structure d'un bâtiment directement par requête sur la base de données établie.

## Conclusion et perspectives

Bien que le monde réel soit en 3D, sa représentation numérique, ainsi que les supports de visualisation (écrans, papier) sont limités à l'espace 2D. de nos jours ces modèles ne sont pas suffisants pour répondre aux besoins actuels en informations géographiques. L'intégration de la troisième dimension dans le domaine géographique constitue un thème de recherche et d'actualité.

La plupart des modèles 3D, que nous avons étudié en bibliographie sont conçus pour des fins de visualisation dans les systèmes CAO et l'infographie. Ces modèles ne prennent pas en compte les spécificités des SIG tels que l'aspect topologique et le stockage des données dans les Base de données spatiales. Dans ce contexte, le travail que nous avons réalisé est une initiation à ce domaine de recherche et ses applications sous les plates-formes des progiciels SIG. Nous avons contribué par une modélisation de la topologie 3D des structures urbaines et nous nous sommes particulièrement focalisés sur la conception d'un modèle vectoriel topologique 3D comme une extension des modèles topologique 2D sur les structures volumiques telles que le bâti.

Des passerelles des modèles topologiques 2D vers le 3D sont définies pour exploiter les données et les moyens de stockage topologique en 2D. Ce passage n'est possible qu'avec un processus semi-automatique dans lequel l'utilisateur n'intervient qu'au niveau sémantique pour définir les différentes entités et paramètres associés à chaque objet. A cet effet, nous avons exploité le concept de *niveau* comme élément essentiel dans la définition de la position verticale relative de l'objet par rapport au sol et aux autres objets. Un processus automatique a été établi combinant le codage des niveaux et la représentation 2D pour reconstruire les objets selon le modèle topologique 3D. Les relations topologiques 3D relient les objets qui ont des *niveaux pairs identiques* de façon équivalente en 2D et les *différents niveaux pairs* sont reliés verticalement à travers les objets qui ont des *niveaux impairs*.

Pour la mise en œuvre de notre modèle de passerelle topologique 2D à 3D, nous avons utilisé les outils et les moyens topologiques des SGBD-S Open source PostgreSQL/POSTGIS (en 2D) en intégrant les plans de niveaux d'un bâtiment importés sous le format DXF. Dans cette démarche, l'aspect topologique 3D est ajouté aux données dans une base de données topologique 2D. Et pour valider notre modèle nous avons effectué une analyse spatiale 3D directement par requête et visualiser les résultats en format X3D, sur des exemples d'objets volumiques.

Par ailleurs, nous avons été amené à étudier plusieurs outils open source les adapter et d'en tirer l'essentiel pour la réalisation de notre application.

Les perspectives à ce travail consistent en l'intégration des passerelles dans une application SIG en ligne 3D. Cet objectif nécessite l'ajout des nouvelles fonctionnalités pour la visualisation 3D en utilisant la base de données topologique 3D. Au niveau de stockage des données il est utile d'intégrer le concept de niveau dans le module topologique d'un SGBD-S pour permettre l'insertion des objets directement dans les tables topologiques 3D selon leurs valeurs de niveaux sans passer par la création d'un schéma topologique pour chaque valeur de niveau.

L'exploitation de ce modèle avec des données réelles est utile dans plusieurs applications possibles comme le guidage automatique. Ce type d'applications a un intérêt pour la gestion des hauts bâtiments à accès publique ou lors de l'évacuation en cas d'incendies ou autres situations similaires.

## Références Bibliographiques

- [1] A. **Abdul-Rahman**, and J E. Drummond (2000). The Implementation of Object-Oriented TIN-BASED Subsystems For GIS, International Archives of Photogrammetry and Remote Sensing. Vol. XXXIII, Part B4. Amsterdam 2000.
- [2] A. **Abdul-Rahman** and M. Pilouk (2008). Book, Spatial Data Modelling for 3D GIS Springer Berlin Heidelberg New York, 2008.
- [3] **AFNOR**, (1992). Norme EDIGÉO (Echanges de Données Informatisées dans le domaine de l'information Géographique), normalisation française, French standard, no : Z 13-150(ISSN 0335-3931), Aug. 1992.
- [4] P. **Algrain** , J. Denègre and F. Salgé (1991) La normalisation des formats d'échange de données géographiques en France, in 15th ACI Conference, Bournemouth.
- [5] Y. **Bédard**, J. Pouliot, S.Larrivée, P.Normand, et P. Frenette, (2001). Optimisation de la modélisation et du traitement graphique des données 3D urbaines, Rapport de recherche, Centre de recherche en géomatique, Université Laval, Ste-Foy.
- [6] A. **Belaroussi**, (2014). Mémoire de magister, Analyse de réseaux routiers et détermination de chemins optimaux dans un SIG, FEI USTHB 2014.
- [7] **Bentley** (2002) Documentation de MicroStation SE, Documentation en ligne.
- [8] B. **Bessaa**, A. Belhadj-aissa, M. Belhadj-aissa, (2010)."Modèle Topologique 3d pour le réseau routier". XIIème Journées Scientifiques du Réseau Télédétection de l'A.U.F.- 2010 du 23 au 25 novembre 2010 à Monastir Tunisie. [www.js-auf-tun2010.org](http://www.js-auf-tun2010.org).
- [9] R. **Billen**, F. Laplanche, S. Zlatanova, L. Emgard, (2008) , Vers la création d'un méta-modèle générique de l'information spatiale 3D urbaine 2008, Revue XYZ • N° 114 – 1er trimestre 2008.
- [10] A. **Brisebois**, J. Pouliot, B. Lachance, et O.Rabeau, (2003) Three-dimensional spatial analysis : myth or reality, Conférence annuelle du réseau GEOIDE, Victoria,Canada.
- [11] E. **Carlson**, (1987), Three-dimensional conceptual modelling of subsurface structures, In : Technical Papers of ASPRS/ACSM Annual Convention, Baltimore, Vol. 4, pp. 188-200

- [12] C. **Chaillon**,(1992), Architectures des systèmes pour la synthèse d'images, Dunod 1992
- [13] E. **Clementini**, R. Laurini, (2008). Un cadre conceptuel pour modéliser les relations spatiales. Revue des Nouvelles Technologies de l'Information RNTI-E-14():1-17, ISSN 978.2.85428.86, Novembre 2008.
- [14] E. **Clementini** and R. Billen (2006), « Modeling and Computing Ternary Projective Relations between Regions », , IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 18, NO. 6, JUNE 2006.
- [15] **COLLADA** (2008). Digital Asset Schema Release 1.5.0 Specification. April 2008. Editors: Mark Barnes and Ellen Levy Finch, Sony Computer Entertainment Inc.
- [16] V. **Coors**, (2003), 3D-GIS in networking environments, Computers, Environment and Urban Systems 27 (2003) p 345–357
- [17] B. **de Cambray** et T-S. Yeh,(1994) A multidimensional (2D, 2.5D, and 3D) Geographical Data Model, International Conference on Management of Data (COMAD'94), Bangalore, Inde,1994, Tata Mc Graw-Hill ed., p. 317-336.Publié aussi en Rapport PRISM n° 48.
- [18] J. **Dangermond** (2007), Concepts et spécificités des SIG 2007.
- [19] A. **De La Losa**,(2000), Modélisation de la troisième Dimension dans les base de données géographique, Thèse de doctorat, UNIVERSITÉ DE MARNE-LA-VALLÉE, 2000
- [20] B. **Delaunay**, (1934) Sur la sphère vide. Bulletin of the Academy of Sciences of the USSR. 1934,Classe des Sciences Mathématiques et Naturelles, 8, pp. 793-800
- [21] T. **Delpy**, B. Brillault, K. Zeitouni (1993) A Topological Database to Simulate 3DRobotics Navigation, in Third International Conference on Computers in Urban Planning and Urban Management, Atlanta, USA,1993.
- [22] T. **Delpy**, K. Zeitouni (1994) A graph model to describe topological relationships in a 3D world, in Third International Conference on Automation, Robotics and Computer Vision, ICARCV'94, Singapore.
- [23] K. **Devlin**, (1994) Mathematics: The sciences of patterns. Scientific American Library, New York, 215 pp.

- [24] M. J. **Egenhofer** and D. M. Mark, (1995), "Naïve Geography," in Spatial Information Theory : A Theoretical Basis for GIS - International Conference, COSIT'95. vol. LNCS 988, A.U. Frank and W. Kuhn, Eds. Berlin: Springer-Verlag, 1995, pp. 1-15.
- [25] M. J. **Egenhofer**, M., R.HERRING, J (1990). *Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases*. NCGIA Technical Report, N°SBR-9204141
- [26] R. **Elmasri**, S. B. Navathe, (1989) Fundamentals of Database Systems, The Benjamin/Cummings Publishing Company.
- [27]: S. **Flick** (1999), Konzeption eines adaptiven frameworks für 3D-GIS. PhD thesis. Fraunhofer IRB Verlag, Stuttgart. (1999) ,(in German)
- [28]: S. R. **Galati** (2006), book, Geographic Information Systems Demystified, 2006, British Library Cataloguing in Publication Data.
- [29] M. **Goodchild** and K. Kemp (1990), "NCGIA Core Curriculum in GIS", National center for geographic information and analysis, University of California, Santa Barbara,1990, <http://www.geog.ubc.ca/courses/klink/gis.notes/ncgia/toc.html>
- [30] G. **Gröger** et L. Plümer (2005), How to Get 3-D for the Price of 2-D- Topology and Consistency of 3-D Urban GIS, *GeoInformatica* 9:2, 139–158, 2005 Springer Science + Business Media, Inc. Manufactured in The Netherlands.
- [31] G. **Gröger** & L. Plümer (2009), How to achieve consistency for 3D city models, *Geoinformatica* DOI 10.1007/s10707-009-0091-6, # Springer Science + Business Media, LLC 2009.
- [32] B. **Herve** (2005), Etat de l'art des systèmes d'information géographique.
- [33] K. **Hyeyoung**, K. Geunhan, J. Chulmin and Y. Hyunjin (2009), Comparing DBMS-based approaches for representing 3D building objects, 2009 Urban Remote Sensing Joint Event
- [34] H. **Kennedy**, (2001) The ESRI Press dictionary of GIS terminology, Redlands, California, ESRI Press, p. 200.
- [35] F. **Klein**, (1893), "Vergleichen de Betrachtungen über neuere geometrische Forschungen," *Bulletin of the New York Mathematical Society*, vol. 2, pp. 215-249, 1893.
- [36] H. **Kluijver**, and J.E. Stoter, (2003), Noise mapping and GIS: optimising quality and efficiency of noise effect studies, In: *Computers, Environment and Urban Systems (CEUS)*, 2003, Volume 27, no. 1, January 2003, pp.85-102 ISSN: 0198-9715

- [37] T. H. **Kolbe** (2009), Representing and Exchanging 3D City Models with CityGML, 3D Geo-Information Sciences, 2009, pp 15-31, Editors: Jiyeong Lee, Sisi Zlatanova, ISBN: 978-3-540-87394-5 (Print), Springer Berlin Heidelberg, ISSN : 1863-2246
- [38] C. **Koussa** (2009), M. Koehl, A simplified geometric and topological modeling of 3Dbuildings enriched by semantic data: combination of surface-based and solid-based representations, GIS OSTRAVA 2009.
- [39] C. **Koussa** (2011), Implantation d'un système d'information géographique 3D sur Internet pour la gestion des modèles urbains, thèse de doctorat Université de Strasbourg, 2011.
- [40] B. **Lachance** (2005), développement d'une structure Topologique de Données 3D pour l'analyse de modèles géologiques, thèse de maître ès sciences, Université Laval 2005
- [41] T. **Larue**, D. Pastre et M. Chapuis (1992), Schéma d'une base de données géographiques pour véhicules dirigés automatiquement, actes du 15ème Symposium Européen des Systèmes d'Information Urbains, 1992, vol. 2, p. 371-382.
- [42] B. B. **Luis** (2003), Apports des systèmes d'information géographique et l'évaluation de la qualité des eaux côtières. Une contribution à la gestion intégrée de la zone côtière Colombienne» Thèse pour l'obtention du grade de Docteur en Sciences techniques 2003, Ecole Polytechnique Fédérale de Lausanne.
- [43] D. M. **Mark**, JA. Cebrian, (1986), Octrees: a useful data-structure for the processing of topographic and sub-surface data. Technical Papers of ACSM-ASPRS Annual Convention. Vol. 1 (Cartography and Education).
- [44] D. **Meagher**, (1982), Geometric modelling using octree encoding. Computer Graphics and Image Processing. Vol. 19, pp. 129-147
- [45] M. **Molenaar**, (1989) Single valued vector maps - a concept in GIS, Geo-Information-Systeme. Vol. 2, No. 1, pp. 18-27
- [46] M. **Molenaar**, (1990). A formal data structure for 3D vector maps. Proceedings of EGIS'90, Amsterdam, The Netherlands, pp.780-781
- [47] M. **Molenaar**, (1991) Formal data structures, object dynamics and consistency rules. Digital Photogrammetric Systems. Herbert Wichmann Verlag GmbH, Karlsruhe, pp. 262-273.

- [48] M. **Molenaar**, (1992), A topology for 3D vector maps, ITC Journal 1992-1, pp. 25-33.
- [49] **OGC**, (1994), Open Geospatial Consortium. OpenGIS® and OGC® are registered trademarks of OGC© 1994–2014,site officiel : <http://www.opengeospatial.org/standards/citygml>
- [50] M. **Pilouk**, (1996) Integrated modelling for 3D GIS, Thèse de doctorat, International Institute for Geo-Information Science and Earth Observation (ITC), p. 200.
- [51] **PostGIS** 2.0.4 Manual (2014), POSTGIS officiel website : <http://www.postgis.org>, 2014.
- [52] D. **Pullar**, M.J.Egenhofer, (1988) Towards formal definition of topological relations among spatial objects, Third International Symposium on Spatial Data Handling, Sydney, Australia, p. 225-242.
- [53] **Q. Weng** (2010), book, Remote sensing and GIS Integration (Theories, Methods And Applications) .The McGrawhill companies ,2010.
- [54] F. **Ramos** (2003), Modélisation et validation d'un système d'information géographique 3D opérationnel, thèse de doctorat de l'université de Marne-La-Valée 2003.
- [55] J. **Raper**, B. Kelk, (1991) Three-dimensional GIS, In: Geographical information systems: principles and applications. D J Maguire, M Goodchild andDW. Rhind (eds.) Longman Geoinformation, pp. 299-317.
- [56] D. **Schneuwly**, R. Caloz (2010), Les modèles numériques, Geographic Information Technology Training Alliance (GITTA), <http://www.gitta.info-2010>.
- [57] B. **Schön**, D. F. Laefer, S. W. Morrish and M. Bertolotto (2009), Three-Dimensional Spatial Information Systems: State of the Art Review, Recent Patents on Computer Science, 2009, Vol. 2, No. 1
- [58] J. **Stoter** and S. Zlatanova (2003), 3D GIS, where are we standing?, ISPRS Joint Workshop on 'Spatial, Temporal and multi-dimensional data modelling and analysis', Québec, October, 2003
- [59] T. **Ubeda** (1997), Contrôle de la qualité spatiale des bases de données géographiques : cohérence topologique et corrections d'erreurs, thèse soutenue le 1er décembre 1997, INSA Lyon.

## Références Bibliographiques :

---

- [60] P. **Van Oosterom**, W. Vertegaal, M. Van Hekken (1994), Integrated 3D modelling within a GIS, in Advanced geographic data modelling, Martien Molenaar Sylvia de Hoop, 1994, p.80-95.
- [61] **Wees**, J.D. van, R.W. Versseput, H.J.Simmelink, R.R.L. Allard and H.J.M. Pagnier, (2002), Shared Earth system models for the dutch subsurface, Netherlands Institute of Applied Geoscience TNO-National Geological Survey, Geo-informatiedag, 2002, February, 2002, Ede, the Netherlands.
- [62]: K. **Zeitouni**, B. de Cambray (1995), Topological Modelling For 3D GIS, Fourth International Conference on Computers in Urban Planning and Urban Management, Melbourne, Australie / Australia, Juillet 1995 / July 1995.
- [63] S. **Zlatanova**, (2000). 3D GIS for Urban Development. PhD Thesis, ITC, The Netherlands, 222 p.
- [64] S. **Zlatanova**, A. Abdul-Rahman, and W. Shi (2002). Topology for 3D spatial objects. International Symposium and Exhibition on Geoinformation 2002, 22-24 October, Kuala Lumpur, Malaysia. Kluwer Academic Publishers. ISSN:1384-6175.
- [65] S. **Zlatanovaa**, A. Abdul Rahman b, M. Pilouk (2002), 3D GIS: Current Status and Perspectives Symposium on Geospatial Theory, Processing and Applications, Ottawa 2002.

### Résumé

Bien que le monde réel soit en 3D, sa représentation numérique, ainsi que les supports de visualisation (écrans, papier) sont limitées à l'espace 2D. A nos jours ces modèles ne sont pas suffisants pour répondre aux besoins actuels en informations géographiques. Les besoins en 3D ne sont apparus que dernièrement avec le développement technologique et informatique. Alors, l'intégration de la troisième dimension dans le domaine géographique constitue un thème de recherche et d'actualité.

La plupart des modèles 3D sont conçus pour des fins de visualisation dans les systèmes CAO et l'infographie. Ces modèles ne prennent pas en compte les spécificités des SIG tels que l'aspect topologique et le stockage des données dans les Base de données spatiales. Dans ce contexte se place notre travail qui s'intéresse à la modélisation topologique 3D pour les structures du milieu urbain.

Dans ce travail nous nous sommes concentrés sur la conception d'un modèle vectoriel topologique 3D comme une extension des modèles topologique 2D. Des passerelles des modèles topologiques 2D vers le 3D sont définies pour exploiter les données et les moyens de stockage en 2D dans les modèles topologique 3D. Le passage du 2D vers le 3D n'est possible qu'avec un processus semi-automatique dans lequel l'utilisateur intervient au niveau sémantique pour définir les différentes entités et quelques paramètres à chaque objet. Parmi ces paramètres, nous utilisons le concept de *niveau* qui sert à définir la position verticale relative de l'objet par rapport au sol et aux autres objets. Lors du traitement de passage du 2D vers la 3D, le processus automatique utilise ces paramètres avec la représentation 2D pour reconstruits les objets selon le modèle topologique 3D. Les relations topologiques 3D relient les objets qui ont des *niveaux* pairs identiques de façon équivalente en 2D et les différents *niveaux* pairs sont reliés verticalement à travers les objets qui ont des *niveaux* impairs.

Ainsi, nous avons établi un modèle conceptuel dans lequel le concept de *niveau* utilisé permet d'étendre un modèle 2D pour représenter les objets dans un modèle topologique 3D de façon simple.

Nous avons effectué la mise en œuvre du modèle d'une passerelle topologique du 2D vers le 3D avec les outils et les moyens des SGBD-S Open source PostgreSQL/POSTGIS (en 2D) en utilisant des plans de niveaux d'un bâtiment importés sous le format DXF. La base de données a été établie suivant le processus décrit ci-dessus. Dans cette démarche l'aspect topologique 3D est ajouté aux données dans une base de données 2D.

**Mots clés :** SIG, Modèle vectoriels, Modèle topologique, topologie 2D, topologie 3D.