

**Département de géomatique appliquée  
Faculté des lettres et sciences humaines  
Université de Sherbrooke**

**Système d'information géographique intérieur 3D pour les services d'urgence**

**Alexandre Gobeil**

**Éssai présenté pour l'obtention du grade de Maître ès sciences géographies (M. Sc.),  
cheminement géodéveloppement durable**

**© Alexandre Gobeil 2017**

## Composition du jury

Système d'information géographique intérieur 3D pour les services d'urgence  
Alexandre Gobeil

Cet essai a été évalué par un jury composé des personnes suivantes :

Pr. Mickaël Germain, directeur de recherche

(Département de géomatique appliquée, Faculté des lettres et sciences humaines)

Marcel Laperle, autre membre du jury

(Département de géomatique appliquée, Faculté des lettres et sciences humaines)

## **Remerciements**

Je tiens à remercier Pr. Mickaël Germain de l'Université de Sherbrooke, André Corriveau et Michael Howard de la Ville de Sherbrooke, ainsi que Marie-Ève Fortin pour l'aide inestimable apportée à la réalisation de cet essai.

## **Sommaire**

Les autorités doivent intervenir le plus rapidement possible dans des situations imprévues et soudaines qui menacent la sécurité des personnes et l'intégrité des biens. L'objectif du projet est de réaliser un modèle de réseau 3D pour l'immeuble Jean-Charles Côté de la Ville de Sherbrooke et de présenter l'information dans un système d'information géographique. Ce modèle est réalisé dans l'objectif d'apporter de l'information géographique critique en cas d'intervention et d'effectuer des analyses d'itinéraires à partir d'un réseau de lignes de déplacement intérieur. Le modèle est entièrement conçu dans le système ESRI, à partir des plans CAO du Centre Jean-Charles Côté qui ont été fournis par la Ville de Sherbrooke. L'essai aborde également l'échange de données et les standards de partage de l'information géographique intérieure entre les systèmes d'information.

## **Table des matières**

Remerciements .....	iii
Sommaire .....	iv
Liste des tableaux .....	vi
Glossaire.....	vii
1. Introduction .....	1
1.1. Mise en contexte.....	1
1.2. Problématique.....	2
1.3. Objectifs.....	2
2. Cadre théorique .....	3
2.1. État des connaissances.....	3
2.1.1. L'information spatiale intérieure.....	3
2.1.2. Les systèmes d'information géographiques intérieurs .....	4
2.1.3. La structure de l'information.....	6
2.1.4. Les modèle de réseau dans les immeubles .....	8
2.2. Revue de littérature.....	9
3. Matériel et méthodes .....	11
3.1. Aperçu des données de base .....	11
3.2. Méthodologie.....	15
3.2.1. Schéma méthodologique .....	15
3.2.2. Conversion des données et géoréférencement .....	16
3.2.3. Création des lignes intérieures de déplacement .....	17
3.2.4. Ajout de l'élévation.....	18
3.2.5. Liens entre les étages.....	19
3.2.6. Construction du réseau .....	20
3.2.7. Calcul des itinéraires .....	21
4. Résultats .....	22
4.1. Application python pour la visualisation et le calcul d'itinéraire. ....	23
4.2. Interprétation et discussion des résultats .....	24
5. Conclusion.....	25
6. Recommandations .....	26
7. Références .....	27
ANNEXE I: Scripts python.....	31

## **Liste des figures**

Figure 1 : extrait CityGML .....	7
Figure 2 : extrait IndoorGML.....	8
Figure 3 : cartographie des données de base .....	13
Figure 4 : plans des étages du Centre Jean-Charles Côté.....	14
Figure 5 : méthodologie de création du modèle 3D et de calcul des itinéraires .....	15
Figure 6 : transfert des CAO vers classe d'entité.....	17
Figure 7 : plan des étages géoréférencés .....	17
Figure 8 : lignes intérieures de déplacement.....	18
Figure 9 : création des classes d'entité 3D .....	19
Figure 10 : pièces et lignes intérieures de déplacement en 3D .....	19
Figure 11 : intégration des escaliers aux lignes de déplacement.....	20
Figure 12 : liens entre les étages connectés au reste du réseau piétonnier .....	20
Figure 13 : résultat d'un calcul d'itinéraire .....	22
Figure 14 : localisation de la génératrice .....	23
Figure 15 : création des itinéraires .....	24
Figure 16 : illustration des temps de déplacement .....	24
Figure 17 : résultat d'un déplacement.....	25

## **Liste des tableaux**

Tableau 1 : Liste des données de base.....	11
Tableau 2 : Contenu du fichier CAO .....	13

## **Glossaire**

2D : Deux dimensions

3D: Trois dimensions

ESRI: *Environmental System Research Institute*

CAO: Conception assistée par ordinateur

GML: *Geography Markup Language*

IFC: *Industry Foundation Class*

LOD: *Level of Detail*

OGC: *Open Geospatial Consortium*

SIG: Système d'information géographique

XML: *Extensible Markup Language*

## **1. Introduction**

### **1.1. Mise en contexte**

Lorsqu'il est question de la sécurité des personnes, chaque seconde compte. Les autorités doivent intervenir le plus rapidement possible dans des situations imprévues et soudaines qui menacent la sécurité des personnes et l'intégrité des biens. Les premiers répondants interviennent dans des opérations de sauvetage ou de traitement médical, des situations de nature criminelle ou environnementale. Ils fournissent les premiers soins pour stabiliser l'état de santé de la personne en détresse et doivent rétablir la situation par une intervention rapide et contrôlée (APSAM, 2016). La rapidité de leur intervention réduit les conséquences possibles pour les victimes. Au Québec, ce sont les pompiers et les ambulanciers qui sont désignés comme premiers répondants. Les policiers et les autorités environnementales interviennent également dans des situations d'urgence (MSP, 2013).

Les moyens et les méthodes d'intervention s'appuient sur la connaissance de la situation qui s'élabore entre le moment de l'appel d'urgence et l'arrivée sur les lieux. La connaissance de la situation (*situational awareness*) est « la perception des éléments de l'environnement, à l'intérieur d'un espace et dans un laps de temps, la compréhension de leur sens et la projection de leur évolution dans le temps » (US Coast Guard, 2016, ch.5, p.1). Cette connaissance de la situation est cruciale pour intervenir sans délai. Pour améliorer leur connaissance de la situation et le temps de réponse, les premiers répondants doivent disposer de toute l'information nécessaire en temps réel. L'information utile et critique en cas d'urgence est transmise à une variété de systèmes et d'intervenants lors d'un événement (Holmberg, 2013).

Plusieurs situations peuvent survenir à l'extérieur comme à l'intérieur des bâtiments, des complexes industriels et des lieux de travail : incendies, déversements de matières dangereuses, opérations de sauvetage et traitement médical, pannes électriques, colis suspects, cambriolages, alertes à la bombe, etc. Plusieurs employeurs sont tenus d'élaborer un plan de mesures d'urgence ayant pour objectifs de protéger les travailleurs et de réduire les dommages possibles en cas de désastre. Ce plan mentionne les procédures à suivre et identifie les dangers auxquels sont exposés ses clients et ses employés (APSAM, 2016).



## **1.2. Problématique**

Les premiers répondants doivent intervenir dans des lieux et des structures urbaines de plus en plus complexes tels que des immeubles ou des ensembles d'immeubles. L'information permettant de se rendre sur les lieux d'une intervention et les éléments critiques à la connaissance d'une situation ne sont pas toujours connus par les premiers répondants. Par conséquent, ils peuvent perdre du temps à s'orienter à l'intérieur des structures et à procéder à la reconnaissance du terrain. Des plans d'intervention complets doivent par ailleurs identifier toutes les sources de risques potentiels et doivent permettre de planifier des interventions rapides. Le développement récent des technologies et des systèmes d'information géographiques (SIG) en 3D permet maintenant de modéliser l'intérieur d'un immeuble, de tracer des itinéraires possibles et de présenter l'information pertinente en cas d'urgence.

## **1.3. Objectifs**

L'objectif du projet est de réaliser un modèle de réseau 3D pour l'immeuble Jean-Charles Côté de la Ville de Sherbrooke à partir de lignes de déplacement intérieures et de présenter l'information dans un SIG. Ce modèle permettra d'effectuer des analyses d'itinéraires en cas d'intervention d'urgence. Le calcul d'itinéraires permet aux premiers répondants de trouver le plus court chemin dans l'immeuble et ainsi réduire le temps de déplacement.

Le modèle permettra plus spécifiquement :

- d'apporter de l'information géographique critique en cas d'intervention;
- de cartographier les temps de déplacement et les distances de parcours dans l'immeuble dans une perspective de prévention et de réduction des risques;
- d'aborder l'échange de données et les standards de partage de l'information géographique intérieure entre les systèmes d'information;
- d'évaluer l'avancement des technologies ESRI dans la conception de réseaux intérieurs et de formuler des recommandations dans l'utilisation de ces systèmes.

## **2. Cadre théorique**

### **2.1. État des connaissances**

#### **2.1.1. L'information spatiale intérieure**

Selon Li (2008, p1), la différence principale entre l'information spatiale intérieure et extérieure est « la contrainte de l'espace » : le confinement et les limites formées par les éléments architecturaux de l'immeuble tels que les murs, les portes, les corridors et les escaliers. La représentation de ces contraintes est l'élément clé de l'information et des normes de modélisation spatiale intérieure (OGC, 2014, p.16).

La limite avec l'extérieur peut cependant être floue et difficilement déterminée de façon précise. L'entrée d'un immeuble et ses environs représentent une transition avec l'extérieur : une connectivité. On parlera plus précisément d'un SIG représentant un espace micro ou scène, par opposition à macro, plutôt qu'intérieur ou extérieur (Winter, 2012 p.31, Li, 2008). Par ailleurs, un SIG peut représenter non seulement un seul immeuble, mais aussi un ensemble ou un complexe d'immeubles et leurs étages. La seconde caractéristique importante différente de l'information spatiale conventionnelle est la présence de différents niveaux, reliés par des ascenseurs et des escaliers. Les liens spatiaux entre les éléments intérieurs sont représentés plus facilement en 3D (Thill, 2011, p.406).

À la « contrainte de l'espace » s'ajoute également l'espace privé. Dans un immeuble, les déplacements et les accès sont généralement contrôlés en fonction de l'occupation de la personne qui se déplace. Ces caractéristiques et contraintes font en sorte de rendre plus difficile la réalisation d'un SIG à l'échelle d'un immeuble (Winter, 2012 p.36, Li, 2008).

Avec l'augmentation de la taille et de la complexité des constructions, les visiteurs peuvent être confrontés à des environnements inconnus et avoir davantage de difficulté à trouver leur chemin (Kim, 2014). Dans le cas d'un incident qui requiert une réponse et une intervention immédiate, trouver rapidement son chemin dans un immeuble s'avère nécessaire pour minimiser la perte de vies humaines ou de propriété (Liu, 2010 in Kim, 2015). Pour améliorer la connaissance de la situation, les premiers répondants doivent disposer de l'information pertinente. Selon les travaux de l'*U.S. Department of Commerce* (Jones, 2005), l'information pertinente en cas d'urgence peut être classée en deux catégories : statique ou dynamique. L'information statique est

disponible préalablement à l'occurrence d'un événement. L'information dynamique est générée par les systèmes lors d'un événement.

**Information statique :**

- la localisation des points d'accès, des ascenseurs ou des escaliers;
- les obstructions à l'intervention dans l'immeuble ou dans les environs;
- la localisation des systèmes techniques : génératrice, sorties de canalisation d'incendie (*standpipe*), caméras, etc.
- les dangers potentiels à l'intérieur ou à l'extérieur (matières dangereuses).

**Information dynamique :**

- Où l'incident est-il situé dans l'immeuble?
- Quelle route d'évacuation est disponible et non obstruée?
- Quel escalier est le plus approprié pour évacuer ou intervenir?
- Quelle-est la localisation des contaminants (matières dangereuses)?
- Mesures directes des systèmes techniques de l'immeuble : alarmes d'incendie, relevés des détecteurs de chaleur et de fumée (Jones, 2005, p. 3-4).

2.1.2. Les systèmes d'information géographiques intérieurs

La demande pour l'information spatiale vient d'activités humaines qui requièrent un processus décisionnel spatio-temporel. Dans le cas de situations de la vie courante comme dans des situations exceptionnelles d'urgence, c'est la possibilité de gagner du temps, de s'orienter et de trouver son chemin dans des environnements qui ne sont pas familiers qui donne la valeur à l'information transmise par un SIG (Winter, 2012).

Les SIG, généralement en 2D, intègrent de plus en plus la troisième dimension (Kim, 2014). Cependant, avec l'expansion de leurs frontières, les SIG ne se limitent plus à l'espace extérieur : ils englobent maintenant l'espace intérieur, des domaines traditionnellement réservés aux systèmes de conception assistée par ordinateur (CAO) (Li, 2008). L'idée de réunir l'espace intérieur et extérieur dans un même système d'information apporte des défis particuliers, car il existe une importante différence d'échelle et de concepts entre les deux systèmes (Winter, 2012). Les données de CAO sont constituées de géométries, de textes et de symboles organisés en couches ou niveaux. Les données de SIG sont structurées selon leur géométrie et sont

organisées en classes d'entités géométriques géoréférencées (ESRI, 2015). Les logiciels de CAO sont généralement utilisés pour concevoir et représenter les espaces intérieurs qui n'existent pas encore. Les logiciels de SIG ont été développés pour l'espace extérieur principalement (Isikdag, 2008).

Dans le cadre de la gestion des urgences, les SIG intérieurs sont utiles sous deux aspects : avant l'intervention pour la prévention des risques et durant l'intervention, en fournissant des informations critiques et des itinéraires intérieurs (Isikdag, 2008). Durant l'intervention d'urgence, les premiers répondants doivent améliorer leur connaissance de la situation (*situational awareness*) pour gagner du temps. Un SIG devrait être intégré intérieur/extérieur (*seamless*), présenter l'information en 3D et apporter une information complète au niveau géométrique et sémantique. En plus des itinéraires entre les routes de transport automobile et l'intérieur du bâtiment, le modèle devrait offrir de l'information pertinente sur l'immeuble et ses environs et offrir la possibilité de requêtes spatiales (Tashakkori, 2015).

Les SIG développés dans le système ESRI représentent les immeubles comme un empilement de plans d'étages, séparés par une élévation constante. Les lignes de déplacement schématisées relient les pièces aux portes de sortie et aux corridors. Les étages sont reliés entre eux par les escaliers et les ascenseurs. Le modèle dans l'environnement ESRI est en 2,5 D : il ne peut y avoir deux élévations définies pour un même point. Ce qui signifie qu'il n'y a pas de topologie en 3D définie pour les objets (Thill, 2011). C'est un modèle de stockage d'entités simples.

Généralement, les logiciels de SIG n'ont pas été conçus pour représenter des environnements bâtis à petite échelle, avec un niveau de détail élevé. Ce n'est que récemment que la troisième dimension a été intégrée à ces systèmes. Les logiciels de SIG ont davantage été conçus pour représenter de grandes surfaces avec un niveau d'abstraction plus élevé que dans les logiciels de CAO. Ceux-ci sont généralement utilisés pour la conception des bâtiments et pour représenter des niveaux de détails élevés. Le niveau de détails (*Level of detail* ou LOD) est une des caractéristiques importantes des modèles de bâtiments en 3D. Les niveaux de détails sont classés de la manière suivante (Hagedorn, 2009):

- LOD-1 : enveloppe extérieure du bâtiment et points d'accès;
- LOD-2 : les différents espaces et les chemins intérieurs, plans en 2D;

- LOD-3 : hauteur des espaces, formes des portes et des fenêtres, plus haut niveau de réalisme;
- LOD-4 : montre tous les détails intérieurs et extérieurs, niveau de réalisme supérieur.

La structure des données dans l'environnement ESRI permet l'intégration de la coordonnée Z (la troisième dimension), mais le logiciel ArcGIS ne permet pas la visualisation de la troisième dimension. Il faut utiliser d'autres plates-formes logicielles conçues spécifiquement à cette fin comme, par exemple, ArcScene ou ArcGIS pro de ESRI.

Les CAO et les SIG possèdent leurs propres normes de structure de données, souvent conçues pour des systèmes propriétaires tels qu'ESRI et Autodesk. Plusieurs SIG du domaine scientifique sont conçus à partir des données CAO de gestion des bâtiments. La difficulté réside dans la réorganisation de l'information sur les composantes et les surfaces du bâtiment lors du transfert dans la base de données du SIG. Représenter un immeuble dans un modèle de données géospatiales est également une difficulté (Isikdag, 2008). Pour passer d'un système à l'autre, il faut disposer d'un modèle de données interopérable.

### 2.1.3. La structure de l'information

Selon Li, un modèle de données spécifique au SIG est nécessaire pour décrire adéquatement les contraintes et les éléments architecturaux de l'environnement bâti (Li, 2008). Un modèle de données appelé IFC (*Industry foundation class*) permet de représenter les éléments d'information spatiale dans un immeuble. Ce modèle, qui utilise le XML comme structure de données, permet de décrire les éléments architecturaux en détail, tels que les poutres, les tuyaux, les rampes, etc. (Kim, 2014 p.248). Lee propose le développement d'un modèle « cellulaire » entièrement topologique nommé IndoorGML, dans lequel chaque espace, comme les pièces et les corridors, sont des cellules définies par rapport aux autres cellules.

Deux formats géospatiaux dérivés du *Geography Markup Language* (GML) ont été développés par l'*Open Geospatial Consortium* (OGC) pour permettre l'interopérabilité des SIG intérieurs : CityGML (LOD-4) et son complément IndoorGML. Le modèle de données CityGML a été développé pour modéliser des villes virtuelles en 3D, et permet un très haut niveau de détail dans l'affichage. C'est un schéma d'application basé sur GML 3.2.1, qui permet de définir les éléments structuraux d'un immeuble : planchers, murs, portes, etc. IndoorGML permet de représenter les espaces comme des cellules (Kim, 2014). Il a principalement été développé pour

permettre la navigation intérieure. Ainsi, un espace dispose d'un identifiant unique, et se définit par ses propriétés ainsi que par ses relations avec les cellules adjacentes. Les relations entre les cellules permettent la navigation intérieure. IndoorGML peut aussi inclure des références aux descriptions IFC ou CityGML des objets, permettant ainsi de construire des modèles d'immeubles complets qui disposent d'un niveau de détail élevé.

Le format XML est un langage lisible par un ordinateur, composé d'un ensemble de termes et de valeurs, défini par un schéma et séparé par des balises (`<balise></balise>`). CityGML est un modèle de données XML composé d'un module principal et d'extensions thématiques, qui possèdent chacune leur propre schéma XML. L'exemple suivant (Figure 1) donne la définition, en CityGML, d'une façade d'un immeuble d'un étage. Par exemple, la balise `<posList>` donne la position des 4 coins de la façade (OGC, 2012).

```

<cityObjectMember>
  <bldg:Building gml:id="GML_7b1a5a6f-ddad-4c3d-a507-3eb9ee0a8e68">
    <gml:name>Example Building LOD1 </gml:name>
    <bldg:function
      codeSpace="http://www.sig3d.org/codelists/standard/building/2.0/_AbstractBuilding_function.xml">1000</bldg:function>
    <bldg:yearOfConstruction>1985</bldg:yearOfConstruction>
    <bldg:roofType
      codeSpace="http://www.sig3d.org/codelists/standard/building/2.0/_AbstractBuilding_roofType.xml">1030</bldg:roofType>
    <bldg:measuredHeight uom="#m">5.0</bldg:measuredHeight>
    <bldg:storeysAboveGround>1</bldg:storeysAboveGround>
    <bldg:storeyHeightsAboveGround uom="#m">3.0</bldg:storeyHeightsAboveGround>
    <bldg:lod1Solid>
      <gml:Solid>
        <gml:exterior>
          <gml:CompositeSurface >
            <!-- Face Side 1 -->
            <gml:surfaceMember>
              <gml:Polygon>
                <gml:exterior>
                  <gml:LinearRing>
                    <gml:posList>458875.0 5438350.0 112.0 458885.0 5438350.0 112.0 458885.0 5438350.0 116.0 458875.0
                    5438350.0 116.0 458875.0 5438350.0 112.0 </gml:posList>
                  </gml:LinearRing>
                </gml:exterior>
              </gml:Polygon>
            </gml:surfaceMember>
          </gml:CompositeSurface >
        </gml:exterior>
      </gml:Solid>
    </bldg:lod1Solid>
  </bldg:Building >
</cityObjectMember>

```

**Figure 1 : extrait CityGML**

IndoorGML sert davantage à décrire l'espace intérieur que les éléments architecturaux. Dans IndoorGML, il existe deux espaces : l'un qui décrit des objets solides en 3D (*primal space*), et

l'autre qui représente les relations d'adjacence entre les solides du premier espace (*dual space*) (Lee, 2014, p.19). L'exemple suivant<sup>1</sup> (figure 2) montre un élément solide du premier espace.

```
▼<IndoorFeatures xmlns:gml="http://www.opengis.net/gml/3.2" xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.opengis.net/indoorgml/1.0/core indoorgmlcore.xsd" gml:id="IFs">
  ▼<primalSpaceFeatures>
    ▼<PrimalSpaceFeatures gml:id="PS1">
      ▼<cellSpaceMember>
        ▼<CellSpace gml:id="C1">
          <gml:name>002</gml:name>
          ▼<Geometry3D>
            ▼<gml:Solid gml:id="s11">
              ▼<gml:exterior>
                ▼<gml:Shell>
                  ▼<gml:surfaceMember>
                    ▼<gml:Polygon gml:id="poly002_10">
                      ▼<gml:exterior>
                        ▼<gml:LinearRing>
                          <gml:pos>445538.386149777 5444904.86681726 -2.52</gml:pos>
                          <gml:pos>445538.289557195 5444904.84093536 -2.52</gml:pos>
                          <gml:pos>445538.302498147 5444904.79263906 -2.52</gml:pos>
                          <gml:pos>445537.447653791 5444904.56358421 -2.52</gml:pos>
                          <gml:pos>445537.434712838 5444904.6118805 -2.52</gml:pos>
                          <gml:pos>445535.47871304 5444904.08777193 -2.52</gml:pos>
                          <gml:pos>445534.408496289 5444908.08187523 -2.52</gml:pos>
                          <gml:pos>445537.315933026 5444908.86092055 -2.52</gml:pos>
                          <gml:pos>445538.386149777 5444904.86681726 -2.52</gml:pos>
```

**Figure 2 : extrait IndoorGML**

#### 2.1.4. Les modèle de réseau dans les immeubles

La destination d'un déplacement peut très bien se trouver sur un niveau supérieur d'un immeuble, lui-même constitué d'un réseau complexe de corridors, d'escaliers, d'ascenseurs et autres passages piétonniers. Un système d'information en 2D nous informera sur le chemin à prendre vers l'entrée de l'immeuble, mais il faut également trouver le plus court chemin à l'intérieur et être relié au réseau routier extérieur (Thill, 2011).

Dans l'environnement ESRI, un réseau linéaire est constitué de deux éléments fondamentaux, les tronçons et les jonctions (*edges and junctions*). Les jeux de données de réseau sont utilisés pour modéliser les mouvements des personnes dans les rues, le débit des pipelines ou d'énergie dans les lignes de transmission. Les tronçons se joignent aux jonctions, et la logique de

---

<sup>1</sup> [http://indoorgml.net/data/1.0/FJK-Haus\\_IndoorGML\\_withEXR.gml](http://indoorgml.net/data/1.0/FJK-Haus_IndoorGML_withEXR.gml)

connectivité entre les tronçons permet de calculer différents éléments à l'aide du logiciel, tel que le plus court chemin en termes de temps ou de distance (ESRI, 2016).

Dans un environnement micro, le réseau représente les déplacements possibles par les humains (réseau piétonnier). Il est extrait des entités géométriques du bâtiment. Les lignes de déplacement qui connectent les espaces, les corridors et les entrées de chaque étage sont d'abord créées en 2D. Puis, la même chose est faite pour les ascenseurs, les escaliers et les autres liens qui connectent les étages entre eux. Les points de départ et de fin de ces lignes de connexion sont reliés à chaque étage. L'élévation des lignes de déplacement est ensuite ajoutée. À partir des lignes de déplacement, un jeu de données de réseau est généré, et l'analyse des déplacements est alors rendue possible avec les logiciels ESRI.

## **2.2 Revue de littérature**

Isikdag *et al.* (2008) ont développé un modèle 3D pour la gestion des urgences incendies à partir de données au modèle IFC, entièrement intégré à l'environnement ESRI. La recherche porte davantage sur les problèmes reliés au transfert des données de l'environnement CAO à un environnement géospatial. Les auteurs notent qu'un SIG complet pour la gestion des urgences requiert une quantité importante de données sémantiques et géométriques, et que ces données n'étaient pas aisément transférables d'un environnement à l'autre.

Selon Goetz (2012), les modèles de réseau intérieur ne sont pas bien visualisés sur le Web. Il a développé une application de réseau 3D basée sur XML3D, une extension de HTML5 qui permet de visualiser l'intérieur d'un immeuble en 3D, dans des navigateurs Web comme Google Chrome et Firefox. Les données de base du projet proviennent d'*OpenStreetMap*, une plateforme d'échange communautaire.

L'approche adoptée par Thill *et al.* (2011) vise le développement d'un modèle de réseau 3D de déplacements possibles à l'intérieur et à l'extérieur d'un complexe de 146 immeubles qui forment le *North Campus of the State University of New-York*. Le modèle, appelé 3DCityNet, a été entièrement développé dans le système ESRI. Il a pour objectifs de planifier les déplacements, mesurer l'accessibilité et planifier la localisation d'équipements. Il permet de mesurer des temps de parcours, simuler des temps de sortie de l'immeuble et servir d'aide à la planification en cas d'urgence.



Atila *et al.* (2012) s'intéressent à l'évacuation en cas d'urgence (accidents, attaques terroristes). Selon eux, un des aspects les plus importants dans ce domaine de recherche est de trouver le plus court chemin vers l'extérieur dans un immeuble complexe. Le système idéal pour ce faire devrait comprendre un moteur de calcul d'itinéraires d'évacuation et une interface graphique pour présenter les résultats. Ils ont développé un système qui s'appuie sur le modèle de réseau de la Géo-SGDB Oracle spatial et une interface 3D en Java (module OpenGL). Les calculs d'itinéraires sont effectués par Oracle et présentés par l'interface Java. Les autres éléments graphiques de l'immeuble (murs, structure, etc.) proviennent d'un modèle CityGML.

Récemment, ce sont Tashakkori *et al.* (2015) qui ont développé le modèle de SIG intérieur le plus complet. À partir de données au modèle IFC extraites de plans CAO détaillés du campus de l'Université de Melbourne, ils ont élaboré une plateforme 3D dans l'environnement ESRI (*ArcScene*) qui permet la visualisation, la navigation et l'analyse spatiale avec un niveau de détail supérieur à LOD-4. Le modèle démontre que les SIG intérieurs sont efficaces en situation d'urgence et permettent :

- d'améliorer la perception des premiers répondants de l'espace intérieur en proposant une visualisation en 3D et ainsi gagner du temps lors d'une intervention;
- d'apporter de l'information critique détaillée sur le plan géométrique et sémantique;
- de faciliter la navigation intérieure et extérieure;
- d'intégrer l'analyse spatiale intérieur/extérieur;
- de réduire le temps de réponse en améliorant la connaissance de la situation et en proposant des itinéraires optimaux (Tashakkori, 2015, p.181).

Kim et Wilson (2015) utilisent sensiblement la même approche en modélisant un immeuble de l'université Southern California : des données CAO sont transférées dans le système ESRI, afin de créer une application de réseau intérieur. Le modèle n'a pas pour objectif de servir en situation d'urgence, mais plutôt d'expérimenter la visualisation d'itinéraires intérieurs en 3D dans l'environnement *CityEngine*, un logiciel de modélisation de villes virtuelles d'ESRI avec un niveau de détail LOD-2,5. L'avantage de ce type de modélisation est de permettre une visualisation dans des navigateurs Web qui supportent le format WebGL.

Selon Khan *et al.* (2015), la plupart des applications de réseau actuelles se basent sur une architecture client/serveur pour stocker l'information, calculer des itinéraires et retourner










l'information au client. Cette architecture ne permet pas à l'utilisateur de modifier et ajuster le réseau selon le contexte (par exemple en ajoutant des contraintes supplémentaires). Les auteurs proposent une architecture différente, appelée trois tiers, basée sur le modèle IndoorGML de l'OGC, le traitement délocalisé (*cloud-based computing*), et un client web 3D. Ils ont développé un modèle 3D du campus de l'Université technique de Munich (TUM), permettant de décrire l'information géométrique, sémantique et topologique de l'immeuble. Les auteurs ont également développé une interface simple d'utilisation, basée sur la librairie JavaScript ExtJS et le *plug-in* Google Earth, qui permet la visualisation graphique, l'édition des informations de base et la modification des contraintes du réseau.

### 3. Matériel et méthodes

#### 3.1. Aperçu des données de base

La ville de Sherbrooke a d'abord rendu disponible des couches d'information géographiques vectorielles de la région immédiate du bâtiment : les contours des bâtiments, les segments de rues et de voies ferrées, ainsi qu'une orthophotographie en format jpeg d'une résolution spatiale de 20 cm. Les données vectorielles sont au format geodatabase fichier ESRI. Le tableau suivant (Tableau 1) donne le détail des fichiers géographiques vectoriels et matriciels disponibles. La carte de la page 13 (figure 3) montre la zone d'étude, les données d'empreinte des immeubles et l'orthophotographie utilisée pour construire le modèle.

**Tableau 1 : Liste des données de base**

Type	Données	Description
Orthophotographie	  Ortho2013.jpg  Band_1  Band_2  Band_3 -	Orthophotographie d'une résolution spatiale de 20 cm – 2013 - Format d'image jpg.
Plans	  dn1-janvier2016.dwg   dn2-janvier2016.dwg	Plans de localisation des extincteurs du Centre Jean-Charles Côté – janvier 2016 - Format CAO (DWG).
Couches géographiques vectorielles	Bâtiment	Empreinte des immeubles – polygones (Entités de geodatabase fichier ESRI)
	Segment de rue	Rues du territoire – lignes (entité de geodatabase fichier ESRI)

	Voie ferrée	Voies ferrées du territoire – lignes (entité de geodatabase fichier ESRI)
--	-------------	---

Le modèle de données est conçu principalement à partir des plans de localisation des extincteurs du Centre Jean-Charles Côté, au format CAO, qui ont été fournis par la Ville de Sherbrooke. Ces plans contiennent la disposition et les dimensions des pièces des deux étages de l'immeuble, des corridors, des escaliers et de l'ascenseur ainsi que des éléments architecturaux tels que les murs, les portes et les fenêtres. L'immeuble est le centre Jean-Charles Côté de la Ville de Sherbrooke, qui est un immeuble administratif de deux étages dont la dimension est d'environ 120 X 22 mètres. Il est situé au 555 rue des Grands Fourches Sud à Sherbrooke. Le plancher du deuxième étage est à une hauteur de 3,861 mètres. L'intérêt de cet immeuble pour la réalisation d'un modèle de réseau est que la configuration intérieure est relativement complexe, suffisamment pour ralentir une intervention d'urgence.

Les données sur la disposition des éléments architecturaux proviennent de plans de conception assistés par ordinateur (CAO) au format DWG. Le format DWG est un format vectoriel qui n'est pas géoréférencé, et qui sert généralement à la conception et la gestion des immeubles. La figure 4 de la page 15 montre l'ensemble des plans des deux étages et illustre le niveau de détail de l'information contenu dans les fichiers CAO.

Lorsque le fichier CAO est interprété par le logiciel de SIG, il est converti en 5 types d'entités. La couche de polygones (*Polygon*) comprend les pièces, les murs et les corridors de l'immeuble. Les portes sont illustrées dans la couche de lignes (*polyline*). La couche d'annotation permet d'obtenir des informations sur un élément en particulier, comme la description de la pièce. Les entités « Multipatch » et de points n'est pas utilisée dans ce travail. Le fichier CAO ne contient pas d'information sur l'élévation des éléments. Le tableau de la page suivante (Tableau 2) montre le détail du contenu des fichiers CAO.

**Tableau 2 : Contenu du fichier CAO**

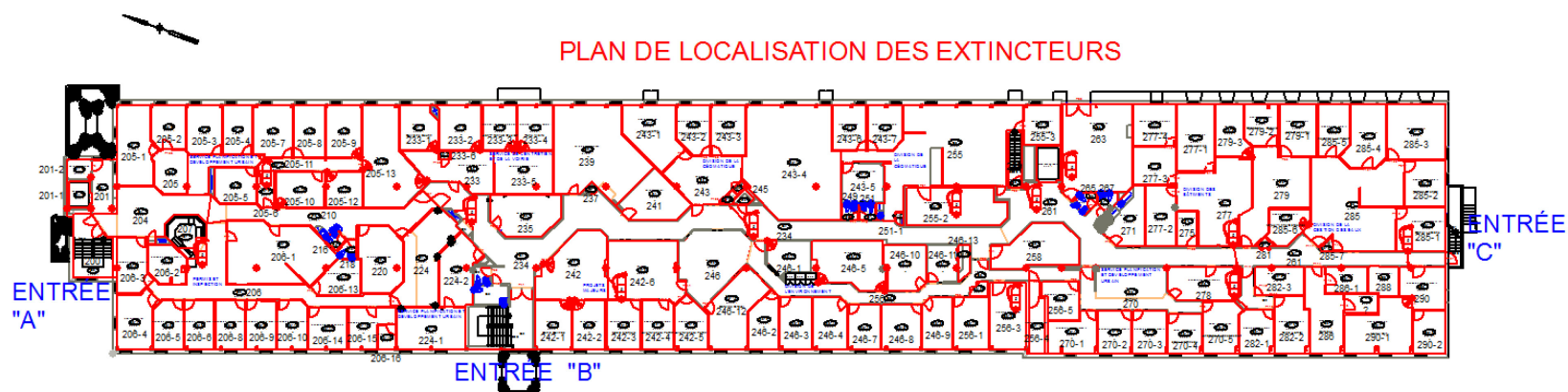
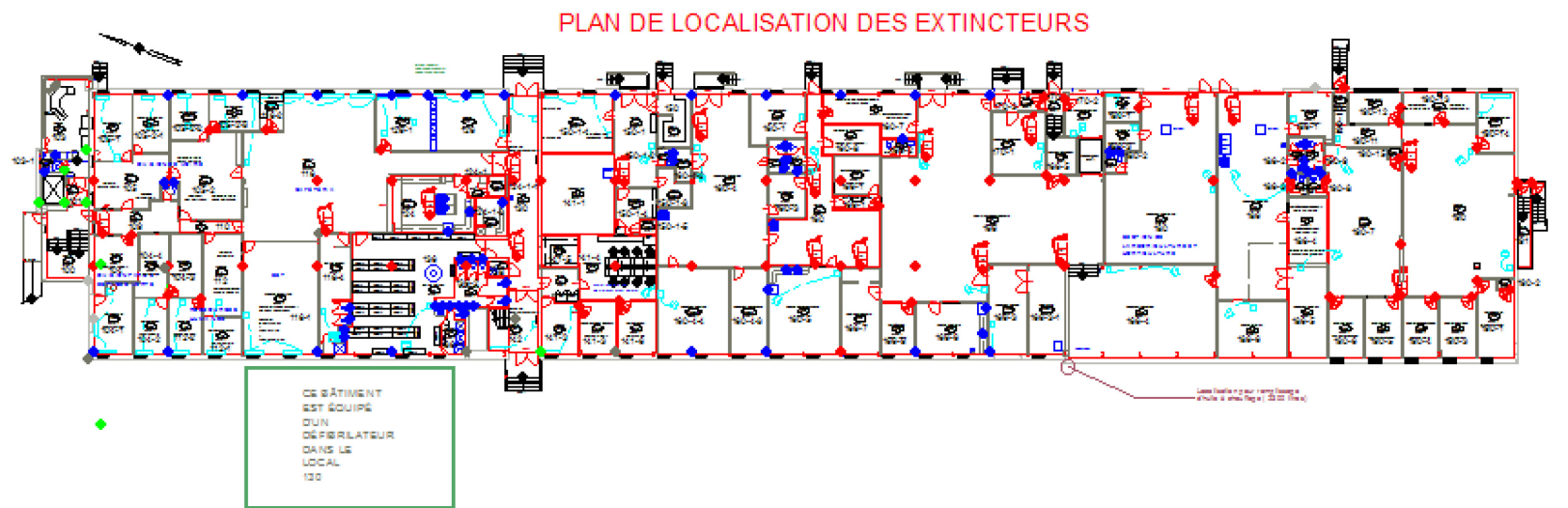
Entité	Description	Exemple
Annotations	Texte et étiquettes	Étiquette #de pièce 180-2
« Multipatch »	Entité de surface qui peut contenir des éléments 3D.	Surface d'une pièce ou de l'épaisseur d'un mur.
Point	Entité de point	Positionnement de différents éléments comme les extincteurs, éviers, etc.
Polyligne	Ensemble de lignes non-continues	Marches d'escaliers, dégagement d'ouverture des portes.
Polygone	Entité de polygone (surface)	Surface d'une pièce ou de l'épaisseur d'un mur.

**Sherbrooke - Secteur du centre Jean-Charles Côté**



Projection: NAD83MTM-zone 7

**Figure 3 : cartographie des données de base**



**Figure 4 : plans des étages du Centre Jean-Charles Côté**

## 3.2. Méthodologie

### 3.2.1. Schéma méthodologique

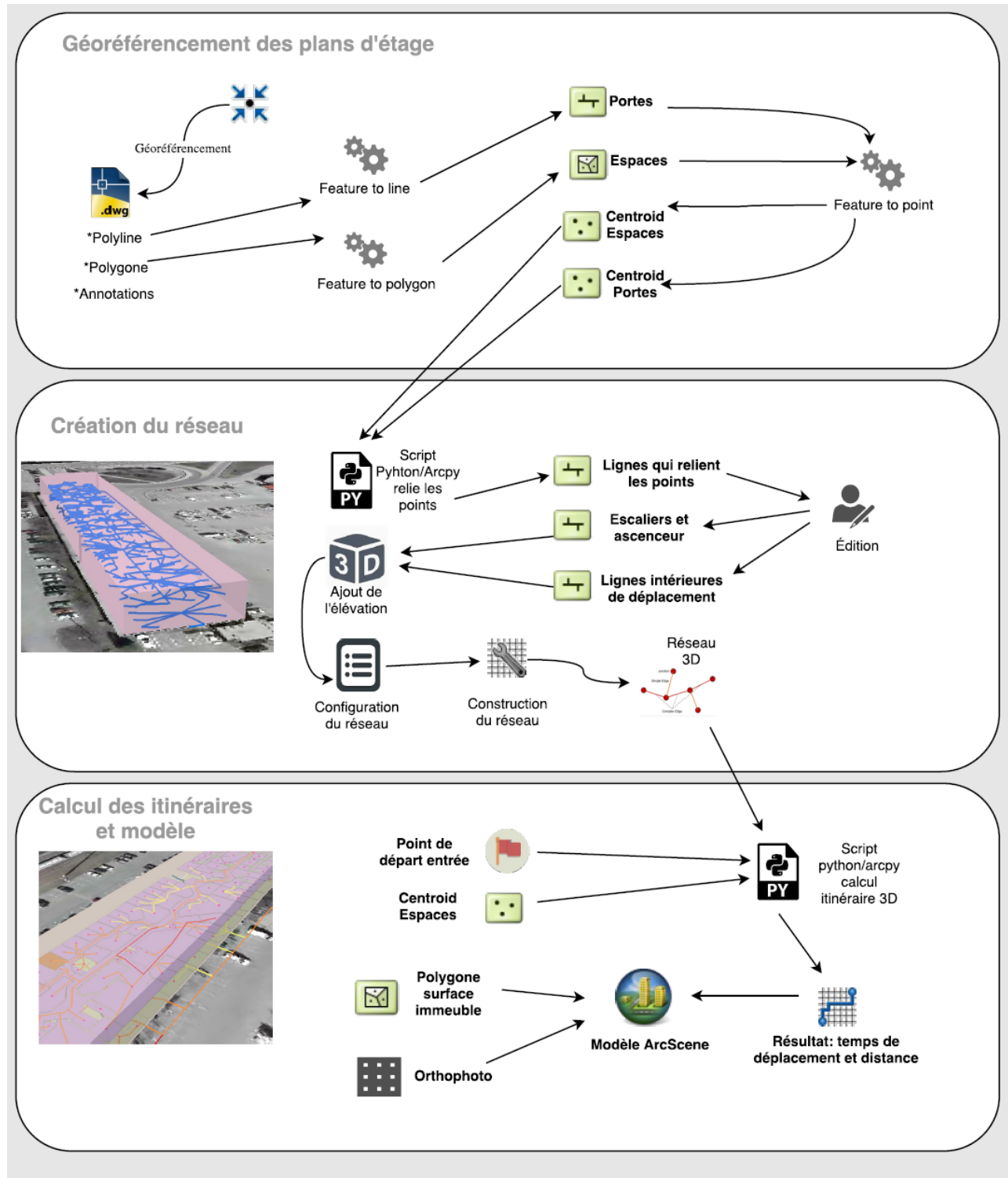


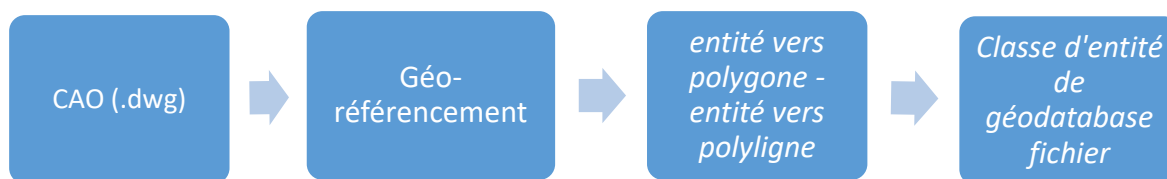
Figure 5 : méthodologie de création du modèle 3D et de calcul des itinéraires

Les données sont entièrement transférées vers ArcGIS. Le format de stockage de l'information est un fichier géodatabase. La création du réseau est faite avec l'extension *Network Analyst* d'ArcGIS et des scripts python de géotraitement. Les résultats en 3D sont présentés dans le logiciel ArcScene, lequel permet l'affichage de la troisième dimension. Le modèle de données est conçu pour visualiser la cartographie intérieure de l'immeuble et pour apporter aux utilisateurs de l'information géographique pertinente en situation d'urgence. Il permet aussi de visualiser des lignes intérieures de déplacement qui servent au calcul d'itinéraires.

### 3.2.2. Conversion des données et géoréférencement

Les plans de conception et les systèmes de gestion des immeubles ne tiennent pas compte de l'environnement géographique, et ne sont pas utilisés pour positionner l'immeuble par rapport aux autres éléments extérieurs. Plus précisément, le géoréférencement permet d'aligner le plan d'étage sur des repères de référence, à l'aide de points de contrôle. Les deux étages ont été géoréférencés par rapport à deux coins opposés du polygone de l'immeuble fourni par la Ville de Sherbrooke. Le logiciel ne permet pas de géoréférencer un CAO avec plus de deux points de contrôle. La méthode qui a été retenue est de géoréférencer les plans par rapport aux contours du bâtiment fournis par la Ville de Sherbrooke. Le contour de l'immeuble provient de relevés d'arpentage, et est une meilleure source de géoréférencement qu'une image orthorectifiée, dont la résolution ne permet pas de bien positionner le plan.

Ensuite, les corridors et les pièces ont été sélectionnés et convertis en classes d'entités Arcgis (*feature class*) et intégrés à une géodatabase. L'utilisation d'une géodatabase fichier permettra par la suite de créer des jeux de données de réseau. Un nouveau jeu de classe d'entité, avec la projection appropriée, a été créé pour contenir les données (NAD83 - MTM zone 7 (CSRS) – EPSG 2949). La conversion s'effectue avec la fonction de géotraitement « *entité vers polygone* », en utilisant l'option « *label features* » qui permet de transférer la couche d'annotation aux polygones. On conserve ainsi le nom et le numéro des pièces de l'immeuble, ce qui permet de les intégrer au modèle par la suite. La même opération est effectuée sur les portes avec la fonction de géotraitement « *entité vers ligne* ». Cette couche sert ensuite à créer les lignes intérieures de déplacement, nécessaires à la création du réseau. La figure 6 de la page suivante illustre le processus de conversion vers une classe d'entité de geodatabase fichier et la figure 7 montre le résultat obtenu.



**Figure 6 : transfert des CAO vers classe d'entité**



**Figure 7 : plan des étages géoréférencés**

### 3.2.3. Création des lignes intérieures de déplacement

Pour générer des itinéraires intérieurs et mesurer le chemin le plus court dans l'immeuble, il faut d'abord tracer des chemins piétonniers. Ces lignes de déplacement sont simplifiées par rapport aux déplacements réels dans un immeuble. Il n'y a pas de méthode automatique de génération de ces lignes, et une grande partie du travail consiste à joindre manuellement des parties de tracés générés par programmation. Les chemins sont des lignes tracées en partant du centre des pièces vers la porte, et reliées à une ligne de centre des corridors. Les lignes de centre des corridors mènent vers l'extérieur de l'immeuble, et s'arrêtent aux ascenseurs et aux escaliers, qui seront tracés ensuite séparément. La manière de tracer les chemins est semi-automatique, par une combinaison de programmation et d'édition manuelle dans ArcGIS. Il faut d'abord créer les points de centre des pièces (centroïde), puis les points de centre des portes. On relie ensuite chaque centroïde avec le point de porte le plus proche à l'aide d'un script python. Par la suite, les points des portes sont reliés à des lignes de centre des corridors manuellement dans ArcGIS. L'image de la page suivante (Figure 8) montre les plans d'étage, ainsi que les lignes de déplacement. Il est à noter que peu d'études présentent une génération



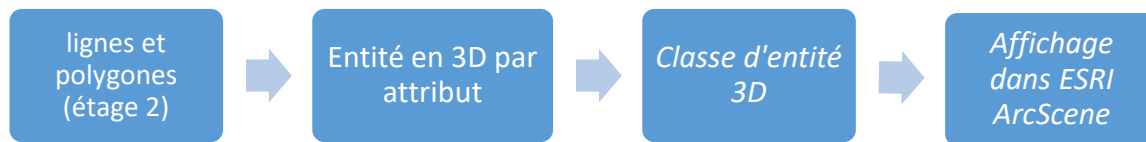
automatique de réseau à partir de plans d'immeubles. Cela implique généralement un traitement manuel, car il est difficile de gérer toutes les exceptions géométriques par programmation.



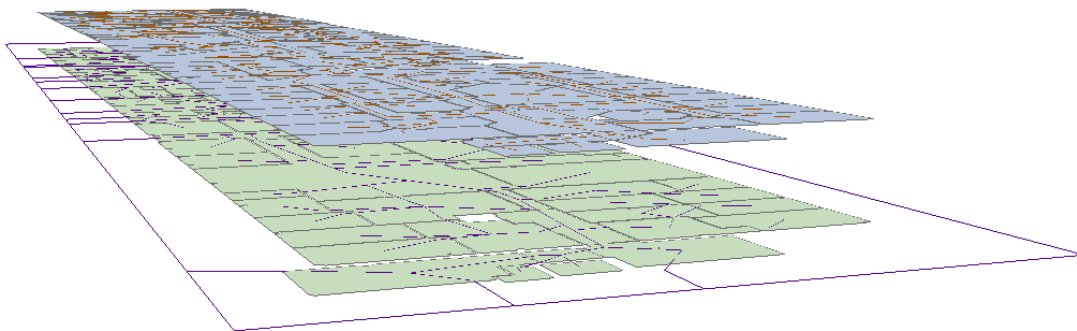
**Figure 8 : lignes intérieures de déplacement**

#### 3.2.4. Ajout de l'élévation

L'étape suivante consiste à ajouter l'élévation du deuxième étage aux pièces et aux lignes de déplacement (l'élévation étant la hauteur à partir du sol, le sol étant considéré au niveau 0). Les géométries de base dans l'environnement ESRI contiennent minimalement les coordonnées de références en 2D (coordonnées X, Y). On peut également transformer les géométries de base pour qu'y soient intégrées les dimensions Z et M : Z est l'élévation et M la longueur (*entité en 3D par attribut*). L'ajout de l'élévation permet aux logiciels d'affichage en 3D comme *Esri ArcScene* de faire afficher les géométries dans l'espace 3D. La hauteur du 2e étage est de 3,861 mètres. La figure de la page suivante (Figure 9) illustre le processus d'ajout de la troisième dimension et la figure suivante (Figure 10) montre le résultat obtenu.



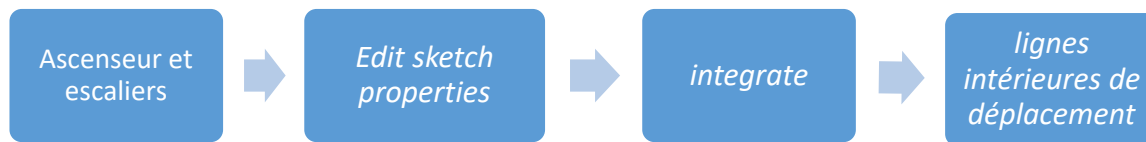
**Figure 9 : création des classes d'entité 3D**



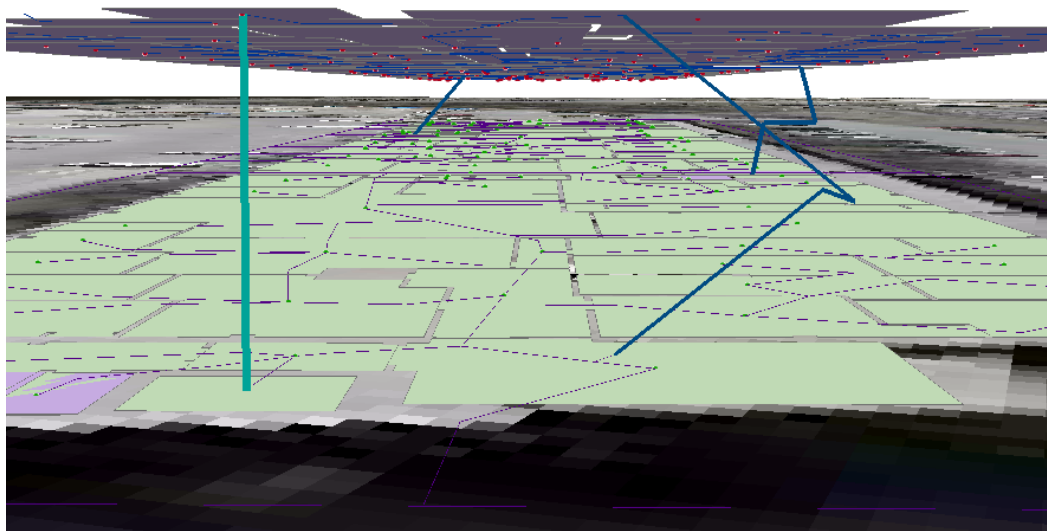
**Figure 10 : pièces et lignes intérieures de déplacement en 3D**

### 3.2.5. Liens entre les étages

Il y a deux types de liens entre les étages dans l'immeuble à l'étude : les escaliers intérieurs et l'ascenseur. Les liens entre les étages sont des lignes dont les extrémités n'ont pas la même élévation. L'élévation des points est déterminée dans ArcScene à l'aide des outils d'édition 3D (*edit sketch properties*). Ces outils permettent de modifier l'élévation de chacun des vertex des lignes séparément. Le logiciel calcule ensuite la ligne entre les vertex. Les liens entre les étages doivent ensuite être connectés correctement au réseau piétonnier avec la fonction « *Integrate* », et en spécifiant un niveau de tolérance approprié (0,1 mètre). Cette fonction permet de relier les extrémités aux lignes les plus proches. Les escaliers et l'ascenseur doivent nécessairement être connectés au reste des lignes de déplacement pour permettre la construction des données de réseau par la suite. La figure de la page suivante (Figure 11) montre le processus d'intégration des liens entre les étages aux lignes intérieures de déplacement et la figure suivante (Figure 12) montre le résultat obtenu.



**Figure 11 : intégration des escaliers aux lignes de déplacement**



**Figure 12 : liens entre les étages connectés au reste du réseau piétonnier**

### 3.2.6. Construction du réseau

Pour effectuer les calculs de temps de déplacement et de plus court chemin dans l'immeuble, on doit disposer d'un jeu de données de réseau créé à partir des lignes de déplacement. Seul le jeu de données de réseau permet d'effectuer ces calculs. Les segments du réseau sur les deux étages sont reliés par un ascenseur et deux escaliers. À la différence des entités de ligne, le réseau contient l'information sur la connectivité entre les segments qui le composent. Chaque segment du réseau contient un attribut de coût, en minutes ou en mètres, et l'accumulation de ce coût permet de mesurer le plus court chemin en termes de temps de déplacement ou de distance, d'un point à l'autre dans l'immeuble. La capacité du logiciel à intégrer des segments avec des élévations différentes pour chaque extrémité permet de mesurer le coût d'un étage à l'autre.

Le réseau est construit à partir des classes d'entités de lignes des étages, des escaliers et de l'ascenseur. Préalablement, chaque classe d'entité doit contenir une colonne de longueur nommée « *length* » et une colonne temps nommée « *minutes* ». La colonne de longueur est calculée automatiquement par ArcGIS en mètres, à l'exception de la longueur des segments de transitions entre les étages, dont la longueur doit être calculée en 3D (note : l'objet « *shape@* » du curseur (*UpdateCursor*) du module *arcpy data access (arcpy.da)* permet d'accéder à l'attribut de géométrie « *length3D* ». On peut donc mesurer la longueur des entités en 3D et la transférer vers l'attribut « *length* »).

La vitesse de déplacement sur chaque segment du réseau, ainsi que des pénalités de temps pour les escaliers sont intégrées aux paramètres. Une vitesse moyenne de déplacement de 5 km/h a été choisie pour le réseau aux étages et dans les escaliers, une vitesse de 2,5 km/h (pénalité de 2x le temps de déplacement). Un temps constant de déplacement de 30 secondes est donné à l'ascenseur.

La colonne de temps doit être calculée en tenant compte de la vitesse de marche dans l'immeuble. Un temps moyen de 5 km/h a été utilisé.<sup>2</sup> La colonne de temps en minutes est obtenue en divisant la longueur des segments par 83,33 m./min. (la mesure du déplacement en mètre pour une minute). Un temps deux fois plus long est calculé pour les escaliers ( $(\text{longueur du segment}/83,33)*2$ ). L'ascenseur est également une ligne dont les extrémités sont à des élévations différentes, et qui sont connectées à chaque étage du réseau.

Pour obtenir un réseau en 3D, chaque classe d'entité utilisée en entrée pour la création du réseau doit disposer de la coordonnée Z, même s'il s'agit de zéro. ArcGIS utilise ensuite cette coordonnée pour la création du réseau. La connectivité (la manière de connecter les segments du réseau) est spécifiée de manière à ce que chacune des lignes soit connectée, et non seulement les extrémités des lignes (*any vertex*). Les attributs du réseau permettent par la suite de calculer le temps ou la distance des itinéraires, et doivent être spécifiés en attributs de coût.

### 3.2.7. Calcul des itinéraires

Une fois le réseau construit, l'extension ArcGIS d'analyse de réseau (*Network Analyst*) permet de calculer l'itinéraire le plus court entre deux points. Il s'agit d'abord de spécifier le point d'origine et de destination, puis d'effectuer le calcul. Le résultat est stocké par le logiciel dans une couche de résultat en dehors de la géodatabase. Il n'est pas possible de visualiser les données de réseau en 3D, bien que l'élévation ait été spécifiée dans les couches de base ayant servi à le créer. On ne peut qu'utiliser ces données en programmation sous-jacente, pour ensuite visualiser des résultats dans des

---

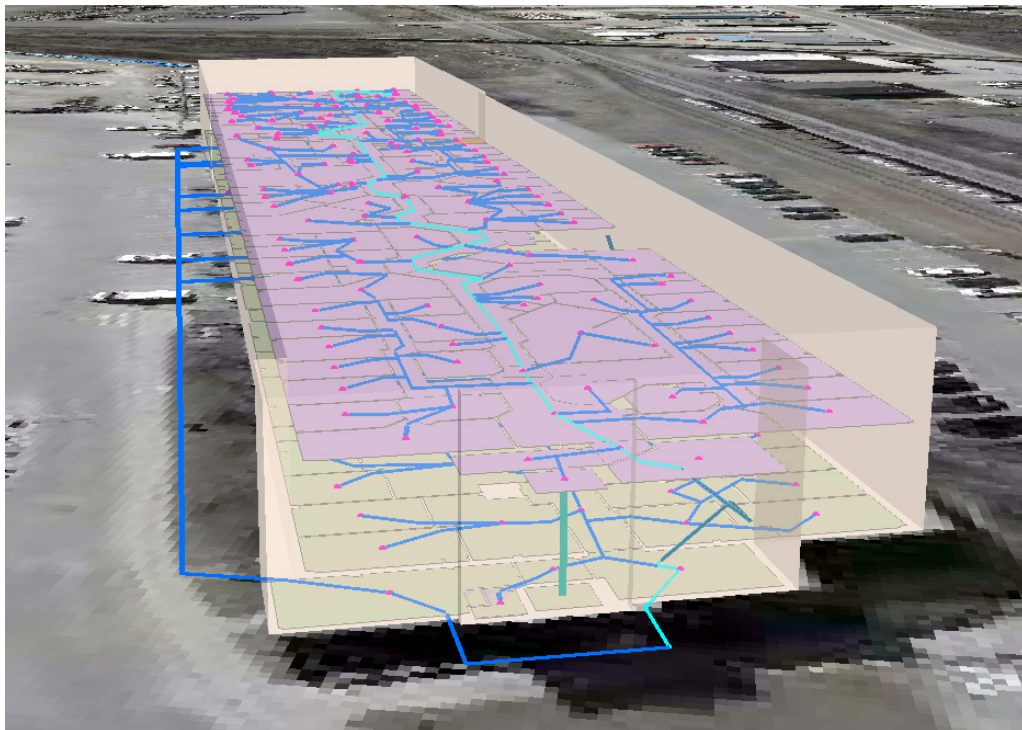
<sup>2</sup> Source : *The pace of life in 31 countries*, NORENZAYAN, LEVINE, 1999. Moyenne de tous les pays = 4,9 km/h.

logiciels qui permettent d'afficher des données géomatiques en 3D. Étant donné que les points de destination en 3D dans l'immeuble ne peuvent être créés de façon manuelle, on doit procéder par programmation pour générer des itinéraires. La librairie Arcpy et l'extension *Network Analyst* (*arcpy.na*) permettent d'utiliser le jeu de données de réseau et d'extraire des itinéraires sous forme d'entité de ligne. Un temps de déplacement et l'itinéraire le plus court sont obtenus à partir du modèle en donnant des coordonnées de latitude, de longitude et d'élévation de l'origine et de la destination. Les points d'origine et de destination sont des couches en entrée et doivent avoir des coordonnées X, Y et Z.

Le plus court chemin entre deux points dans l'immeuble peut être mesuré en termes de temps ou de distance. Cela signifie que l'itinéraire calculé minimise soit le temps de déplacement ou la distance entre le point d'origine et la destination.

#### 4. Résultats

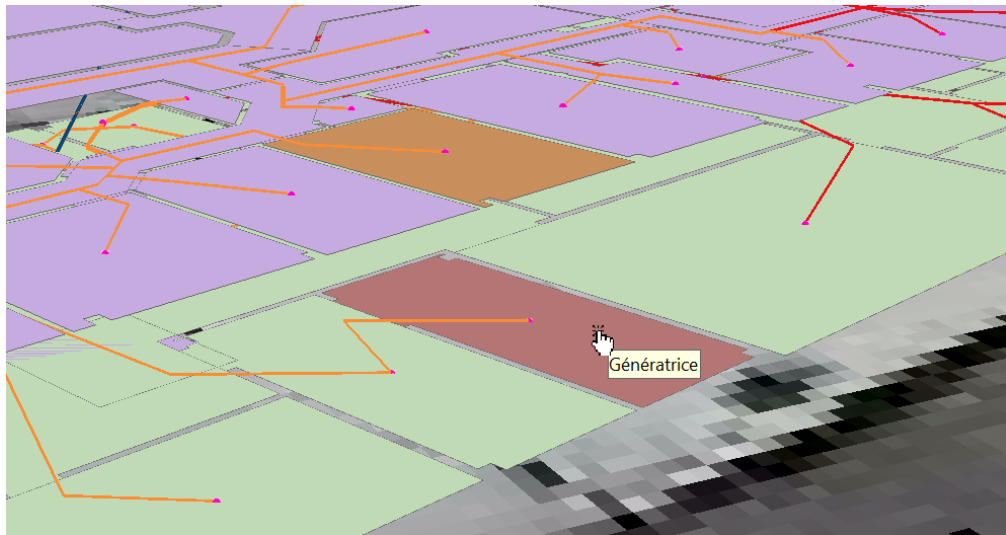
Un essai a été réalisé avec un point d'origine situé à l'entrée de l'immeuble. L'ensemble des itinéraires possibles qui minimisent le temps de déplacement à partir du point d'origine vers les centroïdes de chacune des pièces a été calculé avec le module python Arcpy. L'image suivante montre un itinéraire dont le temps de déplacement est de 1,88 minute et la longueur de 151,39 mètres.



**Figure 13 : résultat d'un calcul d'itinéraire**

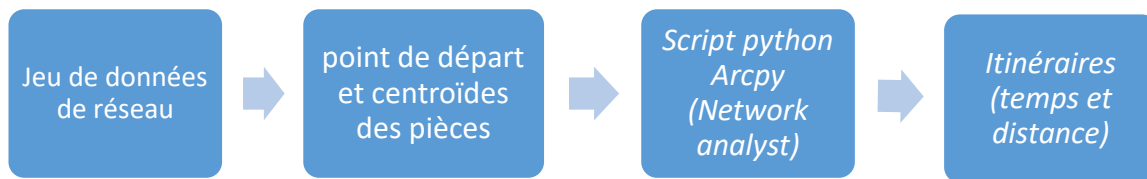
#### 4.1. Application python pour la visualisation et le calcul d'itinéraire.

Dans le contexte de la gestion d'urgences, le modèle permet d'abord de visualiser les points d'entrée et les portes intérieures ainsi que l'environnement extérieur. À l'intérieur, il est aussi possible de visualiser les plans d'étages, la localisation de l'ascenseur et des escaliers. Il est également possible de consulter et de faire des requêtes à partir des informations disponibles sur les espaces de l'immeuble. On peut notamment identifier la localisation de la génératrice au premier étage, et de la chambre électrique au deuxième étage, tel qu'illustré par la figure suivante (Figure 14).

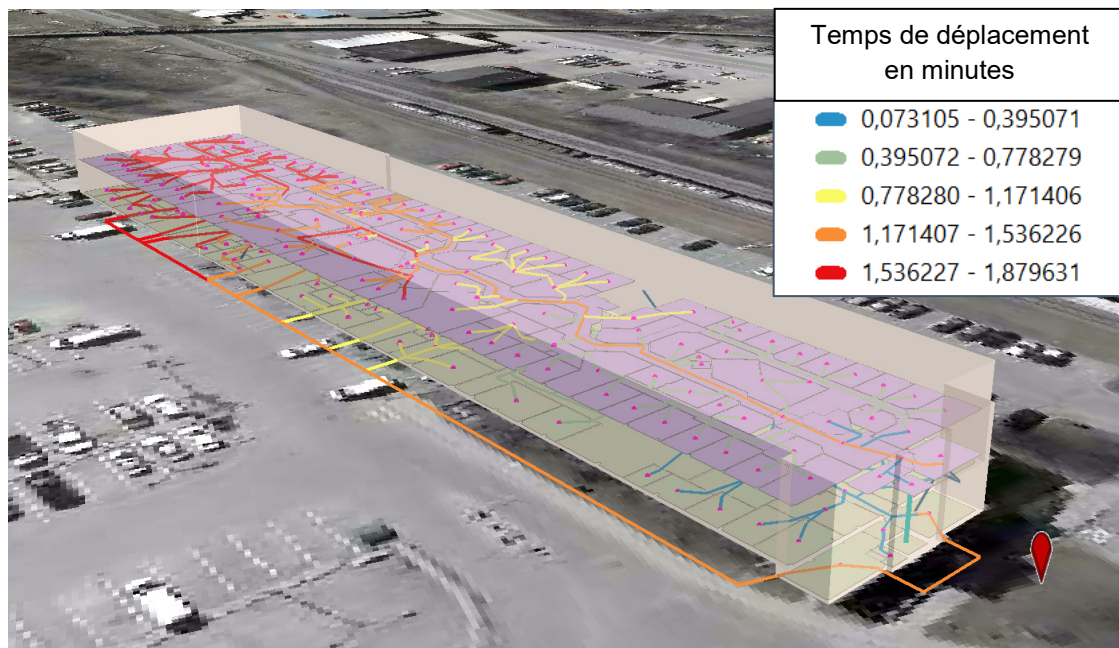


**Figure 14 : localisation de la génératrice**

L'application permet de trouver l'itinéraire le plus court en termes de temps ou de distance à travers le réseau de lignes de déplacement en 3D. Une couche d'itinéraires précalculés à partir de l'entrée principale vers chacun des centroïdes des pièces a été produite à l'aide du modèle sans indiquer de restrictions. Le temps de parcours moyen à partir de l'entrée principale est de 1,07 minute (64,2 secondes). La figure de la page suivante (Figure 16) montre les étapes méthodologiques pour le calcul des itinéraires et la figure suivante (Figure 17) montre le résultat de l'ensemble des calculs ainsi que le point de départ, soit la balise 3D située à l'entrée en bas à droite. C'est une illustration des temps de déplacement à travers l'immeuble.



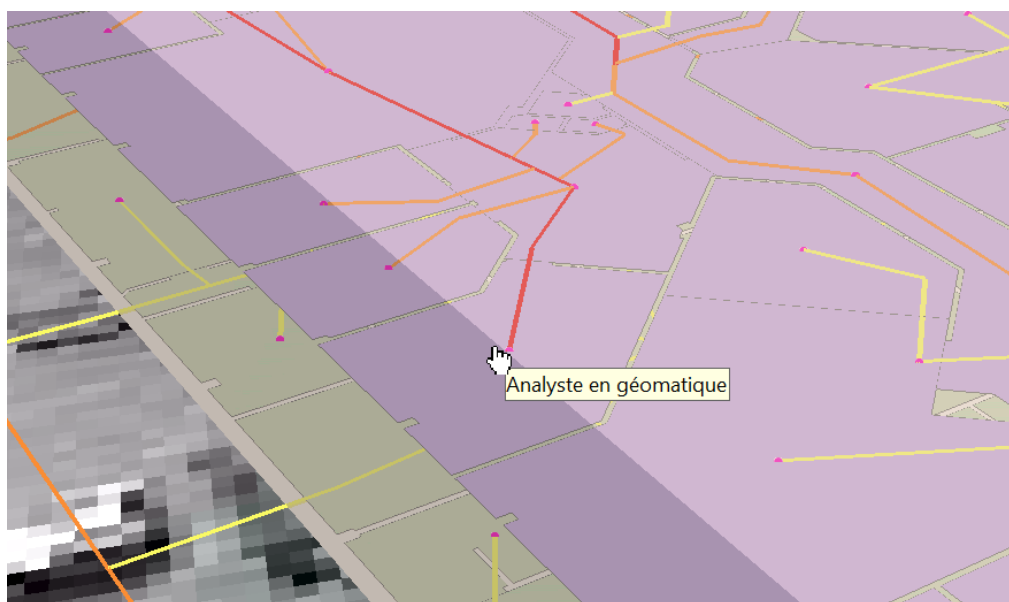
**Figure 15 : création des itinéraires**



**Figure 16 : illustration des temps de déplacement**

#### 4.2. Interprétation et discussion des résultats

Le modèle permet d'identifier des zones plus difficiles d'accès en cas d'urgence en raison de la complexité du parcours dans l'immeuble. On peut constater que l'accès au bureau de l'analyste en géomatique est davantage difficile d'accès que les espaces adjacents par rapport à la porte d'entrée, même s'il est situé au centre de l'immeuble (1,55 minute), comme illustré par la figure de la page suivante (Figure 17).



**Figure 17 : résultat d'un déplacement**

Le parcours le plus long permettant d'atteindre l'extrémité de l'immeuble au deuxième étage est de 1,88 minute, pour une longueur de 151 mètres, à une vitesse de 4,81 km/h. Le ralentissement par rapport à la vitesse moyenne de 5 km/h qui a été appliqué aux segments de lignes de déplacement sur les étages est causé par le passage dans les escaliers. Par ailleurs, on peut noter qu'aucun itinéraire calculé avec ces paramètres ne passe par l'ascenseur, car la pénalité de temps appliquée au modèle est trop élevée et ne permet pas de minimiser le temps de parcours. Lorsque le même essai est réalisé en minimisant la distance, le modèle calcule des itinéraires qui passent par l'ascenseur. On peut noter également qu'aucune barrière n'ayant été appliquée, certains itinéraires débutent à la porte d'entrée, passent par l'extérieur, puis reviennent vers l'intérieur. Il serait également possible d'ajouter des pénalités de temps pour le passage des portes, et des contraintes de réseau pour les portes à l'extérieur. Des résultats différents seraient alors obtenus.

## **5. Conclusion**

La connaissance de la situation et du terrain est critique pour les premiers intervenants en cas d'urgence et un SIG en 3D à l'échelle d'un immeuble peut contribuer à réduire le temps d'intervention. Il est également possible pour une ville de développer un modèle à l'échelle micro permettant le calcul d'itinéraire en 3D dans un environnement ESRI, et ainsi identifier les risques lors de l'élaboration de plans de mesures d'urgence. Le travail réalisé démontre également qu'il est possible d'élaborer de tels modèles à partir de données CAO.



À la lumière des travaux réalisés, on peut noter certaines limites au développement de ce type de modèle. L'obstacle principal est la difficulté du transfert des données CAO au SIG. Le format de données CAO n'est pas parfaitement interprété par ArcGIS, notamment les annotations. Le résultat du transfert est imparfait, et plusieurs informations n'ont pu être transférées automatiquement d'un format à l'autre. Un important travail d'édition a été nécessaire pour ajuster les données de base à la création du modèle. Bien que certaines fonctions existent pour automatiser le travail, la création des lignes de centre et la création du réseau a également nécessité un travail d'édition et d'ajustement. La littérature scientifique ne fournit pas d'exemple de réseau généré automatiquement à partir de données CAO, et du travail d'édition est généralement nécessaire. Un modèle d'architecture ESRI est basé sur un modèle d'entité simple en 2,5D, et ne permet pas d'exploiter toutes les requêtes spatiales en 3D.

## **6. Recommandations**

Étant donné la difficulté de transférer les informations d'un environnement CAO à SIG, il est recommandé de bien évaluer le temps requis pour mettre en œuvre une solution 3D dans le cadre de services d'urgence. Cela pourrait être justifié pour un immeuble très complexe, avec une fréquence d'intervention élevée ou avec un haut degré d'importance stratégique. Par ailleurs, l'intégration d'un réseau en 3D intérieur avec les outils ESRI pourrait être intéressante pour une ville ou une organisation qui dispose déjà d'un système de réseau d'urgence basé sur *Network Analyst* d'ESRI.

Il pourrait être avantageux d'intégrer des plans d'étages en 2D de certains immeubles dans un système de cartographie existant, avec la localisation d'éléments critiques comme les systèmes électriques, les sorties d'urgence et la localisation de produits dangereux. Il est à noter également que des données à l'échelle micro demandent une fréquence de mise à jour élevée, en raison de changements rapides à la configuration des immeubles. De plus, l'utilisation encore fréquente de systèmes hors-réseau (« *offline* ») et à faible débit de données limite le développement d'architectures de visualisation Web connectées.

## 7. Références

Association paritaire pour la santé et la sécurité du travail (APSAM), *Plan de mesures d'urgence (PMU), une obligation?* [Présentation powerpoint]. Tiré de <http://www.aqairs.ca/uploads/Session%20de%20perfect/A4-B4-APSAM-Planmesuresurgence-obligation.pdf>.

Atila, U., Karas, I.R. et Abdul Rahman, A. (2012). "A 3D-GIS Implementation for Realizing 3D Network Analysis and Routing Simulation for Evacuation Purpose". Dans Pouliot, J., et al. (édit.), *Progress and new trends in geoinformation science, Lecture notes in geoinformation and cartography* (p. 249-260), Springer Berlin Heidelberg.

Becker, T., Nagel, C. et Kolbe, T. H. (2009). "A Multilayered Space-Event Model for Navigation in Indoor Spaces". Dans Lee, J., Zlatanova, S. (édit.), *Lecture Notes in Geoinformation and Cartography*, ch.5.

Chen, L-C, Chia-Hao, W., Tzu-Sheng, S. et Chien-Cheng, C. (2014), "The application of geometric network models and building information models in geospatial environments for firefighting simulations", *Computer Environment and Urban Systems*, No. 45, 2014.

Döllner, J, Baumann, K. et Buchholz, H. (2006). "Virtual 3D City Models as Foundation of Complex Urban Information Spaces", *CORP 2006 & Geomultimedia06*, Vienna, ISBN: 3-9502139-0-2.

Environmental Systems Research Institute (ESRI), "Arcobject Online - The ArcGIS network model", Rapport technique.[En ligne]. Disponible : <http://edndoc.esri.com/ArcObjectsOnline/TechnicalDocuments/Network/ArcGISNetworkModel/ArcGISNetwork.htm>.

Environmental Systems Research Institute (ESRI), Williams, K. et Reinhart, J. *CAD: Introduction to using CAD data in Arcgis* [Présentation powerpoint], ESRI user Conference paper sessions, 2015. Tiré de [proceedings.esri.com/library/userconf/proc15/tech-workshops/tw\\_361-127.pdf](http://proceedings.esri.com/library/userconf/proc15/tech-workshops/tw_361-127.pdf).

Goetz, M. (2012). "Using Crowdsourced Indoor Geodata for the Creation of a Three-Dimensional Indoor Routing Web Application", *Future Internet*, Vol. 4, p. 575-591.

Hagedorn, B., Trapp, M., Glander, T. et Dollner, J. (2009). "An Indoor Level-of-Detail Model for Route Visualization", *Tenth International Conference on Mobile Data Management: Systems, Services and Middleware* (IEEE).

Han, L., Zhang, T. et Wang, Z. (2014). “The Design and Development of Indoor 3d routing system”, *Journal of software*, Vol. 9, No. 5.

Herman, L., Kynova, A., Russnak, J. et Reznik, T. (2015). “Comparison of Standard- and Proprietary-Based Approaches to Detailed 3D City Mapping”. Dans Brus, J., et al., *Modern Trends in Cartography: Lecture Notes in Geoinformation and Cartography*, Springer International Publishing Switzerland.

Holmberg, D.G., Raymond, M. A. et Averill J. “Delivering Building intelligence to first responders”, Rapport technique, Gaithersburg, MD, 2013. [En ligne]. Disponible : <http://nvlpubs.nist.gov/nistpubs/TechnicalNotes/NIST.TN.1648.pdf>.

Isikdag, U., Zlatanova, S. et Underwood, J. (2013). “A BIM-Oriented Model for supporting indoor navigation requirements”, *Computers, Environment and Urban Systems*, No. 41, p.112–123.

Isikdag, U., Underwood, J. et Aouad, G. (2008). “An investigation into the applicability of building information models in geospatial environment in support of site selection and fire response management processes”, *Advanced engineering informatics*, Vol. 22, No. 4.

Jones, W.W., Holmberg, D.G., Davis, W.D., Evans, D.D., Bushby, S.T. et Reed, K.A. “Workshop to Define Information Needed by Emergency Responders during Building Emergencies”, Rapport technique, *U.S. Department of Commerce*, NISTIR 7193, 2005.

Jun, C., Park, I., Ryu, K-W. et Koh, J.-H. (2006), “3D GIS Data Model for Indoor Route Finding”, *International Journal of Urban Sciences*, Vol.10, No.2, p.87-95.

Khan, A., Zhihang, Y.A., et Kolbe, T.H. (2015). “Context Aware Indoor Route Planning Using Semantic 3D Building Models with Cloud Computing”, *Lecture Notes in Geoinformation and Cartography*, p.175-192.

Kim, H., Jun, C. et Yi, H. (2009). “A SDBMS-based 2D-3D Hybrid Model for Indoor Routing”, *Tenth International Conference on Mobile Data Management: Systems, Services and Middleware*, IEEE.

Kim, J.-K., Yoo, S.-J. et Li, k.J. (2014). “Integrating IndoorGML and CityGML for Indoor Space”, *W2GIS 2014*, Springer-Verlag Berlin, Heidelberg 2014, p. 184–196.

Kim, K. et Wilson, J. P. (2015), “Planning and visualising 3D routes for indoor and outdoor spaces using CityEngine”, *Journal of Spatial Science*, Vol. 60, No. 1, p.179–193.

Koehl, M., Meyer, E., Coussa, C. et Lott, C., “SIG 3D ET 3D dans les SIG : Application aux modèles patrimoniaux”. [manuscrit auteur]. publié dans GéoÉvénement 2008, France, 2008.

Kolbe, T. H. (2009). “Representing and Exchanging 3D City Models with CityGML”. Dans Lee, J., Zlatanova, S. *Lecture Notes in Geoinformation and Cartography* (chapitre 2), ISBN: 978-3-540-87394-5.

Li, K.-J. (2008), “Indoor space: a new notion of space”. Dans Bertolotto, M., Ray, C., Li, X., *Lecture Notes in Computer Science* (p.1-3), Springer-Verlag, Berlin, vol. 5373.

Li, K.-J. “What is IndoorGML?”. Rapport technique. [En ligne]. Disponible : <http://indoorgml.net/>.

Li, K.-J. et Lee, J. “Indoor Spatial Awareness Initiative and Standard for Indoor Spatial Data”, Proceedings of IROS 2010 Workshop on Standardization for Service Robot. [En ligne]. Disponible: <http://isel.cs.pusan.ac.kr/paper/pdf/iros-2010-indoor-standard.pdf>.

Liu, J., Lyons, K., Subramanian, K. et Ribarsky, W. “Semi-Automated Processing and Routing Within Indoor Structures for Emergency Response Applications”. [En ligne] Disponible: <http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=0CEF1CA5EECF83F771F1733BA8F069D2?doi=10.1.1.225.114&rep=rep1&type=pdf>.

Ministère de la Sécurité publique (MSP), “La sécurité incendie au Québec — 2012”, Gouvernement du Québec, ISBN : 978-2-550-69015-3, 2013.

Open Geospatial Consortium (2012). Gröger, G., Kolbe, T.H., Nagel, C. et Häfele, K-H. “*OGC City Geography Markup Language (CityGML) Encoding Standard*”, Version: 2.0.0, Project document: OGC 12-019.

Open Geospatial Consortium (2014), Lee, J., Li, K.-J., Zlatanova, Kolbe, T. H., Nagel, C. et Becker, T., “*OGC® IndoorGML*”, Version 1.0, Project document: OGC 14-005r3005r3.

Petrenko, A. (2013). “Generation of an indoor navigation network for the University of Saskatchewan”. (Mémoire de maîtrise, Department of Geography and Planning, University of Saskatchewan). Tiré de <https://ecommons.usask.ca/bitstream/handle/10388/ETD-2014-07-1111/PETRENKO-THESIS.pdf?sequence=4>.

Pouliot, J., Daniel, S., Hubert et F., Zamyadi, A. (2013). “Progress and new trends in 3D geoinformation science”, *Lecture notes in geoinformation and cartography*, Springer-Verlag Berlin Heidelberg.

Sun, J. et Li, X. (2011). "Indoor Evacuation Routes Planning with a Grid Graph-based Model", *19th International Conference on Geoinformatics (IEEE)*.

Tashakkori, H., Rajabifard, A. et Kalantari, M. (2015). "A new 3D indoor/outdoor spatial model for indoor emergency response facilitation", *Building and Environment*, No. 89, p.170-182.

Thill, J.-C., Dao, T.-H.-D. et Zhou, Y. (2011). "Traveling in the three-dimensional city: applications in route planning, accessibility assessment, location analysis and beyond", *Journal of Transport Geography*, No.19, p.405-421.

US Coast Guard, "Team Coordination Training Student Guide". [En ligne]. Disponible: <https://www.uscg.mil/auxiliary/training/tct/chap5.pdf>.

Winter, S. (2012). "Indoor spatial information", *International Journal of 3-D Information Modeling*, Vol. 1, No. 1, p.25-42.

Yang, L., Prasanna, R. et King, M. (2009). "On-Site Information Systems Design for Emergency First Responders", *Journal of information technology and applications*, Vol. 10, No.1.

## ANNEXE I: Scripts python

### Programme de génération des lignes intérieures de déplacement

```
# #!/usr/bin/python
# -*- coding:UTF-8 -*-

"""
Objectif du programme: générer des lignes entre les points de portes et centroid
de pièces de l'immeuble (point le plus proche)
"""

__author__ = "Alexandre Gobeil"
__date__ = ": 2016/07/28 $"

# /*****/

import arcpy
import os
import math
from errorHandling import ErrorHandling

#On peut choisir d'exécuter le programme dans un model builder ou dans
simplement dans python.

point1 = arcpy.GetParameterAsText(0)
#point1
=r'C:\Users\Alexandre\Documents\Essai\JCC_Model\JCC_Model.gdb\etages\portes_poin
ts_E1'

point2 = arcpy.GetParameterAsText(1)
#point2
=r'C:\Users\Alexandre\Documents\Essai\JCC_Model\JCC_Model.gdb\etages\SJ2'

#Ici c'est un path
ws = arcpy.GetParameterAsText(2)

#Ici c'est du texte
name = arcpy.GetParameterAsText(3)

#Ou utiliser cela à l'extérieur de model builder
#outputlines1 =
r'C:\Users\Alexandre\Documents\Essai\JCC_Model\JCC_Model.gdb\etages\outputlines1
'

def dist(p1, p2):
    '''Cette fonction calcule la distance entre deux points'''
    '''C'est deux deux tuples en entrée'''
    return math.sqrt((p2[0] - p1[0]) ** 2 + (p2[1] - p1[1]) ** 2)
```

```

def main():

    try:
        outputlines1 = os.path.join(ws, name)

        arcpy.CreateFeatureclass_management(os.path.dirname(outputlines1),
                                           os.path.basename(outputlines1),
                                           "POLYLINE")

        #Le premier curseur permet de faire le tour des points de portes
        for row in arcpy.da.SearchCursor(point1, ["SHAPE@XY"]):
            depart = (row[0][0], row[0][1])
            distance = 1000000

            #ici on trouve le point le plus proche (le centre de pièce le plus
            proche)
            for elements in arcpy.da.SearchCursor(point2, ["SHAPE@XY"]):
                arr = (elements[0][0], elements[0][1])
                d1 = dist(depart, arr)
                if d1 < distance:
                    distance = d1
                    closest = arr

            cursor = arcpy.da.InsertCursor(outputlines1, ["SHAPE@"])

            #Ici on crée une nouvelle entité de ligne

            array = arcpy.Array([arcpy.Point(depart[0], depart[1]),
                                arcpy.Point(closest[0], closest[1])])

            polyline = arcpy.Polyline(array)

            cursor.insertRow([polyline])

    except:

        e = ErrorHandler()
        e.printStackTrace()
        e.writeToFile(r"C:\Windows\Temp\logfile.txt")

if __name__ == '__main__':
    main()
    print "test"

```

## Programme de génération des itinéraires

```
# #!/usr/bin/python
# -*- coding:UTF-8 -*-

"""
Objectif du programme: générer des routes à partir d'une même origine vers un
ensemble de points
"""

__author__ = "Alexandre Gobeil"
__date__ = ": 2016/07/28 $"

# /*****/

import arcpy
#Module maison de gestion d'erreurs
from errorHandling import ErrorHandling

if arcpy.CheckOutExtension("Network") == "NotLicensed":
    print "Pas de licence Network Analyst"
    exit()

def main():

    try:

        #Je défini l'environnement
        arcpy.env.workspace =
r"C:\Users\Alexandre\Documents\Essai\JCC_Model\JCC_Model_v1_1.gdb\model"
        arcpy.env.overwriteOutput = True

        #Les layers de network, points d'origine et destination
        networkDataset = "model_ND"
        origine = "origine"
        stop = "destinations"

        #paramètres impedance "minutes" ou "Length"
        impedance = "minutes"
        accumulation = ["minutes", "Length"]

        #Nouvelle couche de lignes (disabled ou enabled pour coordonnée Z

arcpy.CreateFeatureclass_management(r"C:\Users\Alexandre\Documents\Essai\JCC_Mod
el\JCC_Model_v1_1.gdb\model", "layer_length", "polyline", "template", "",
"ENABLED")

        #Je fais le tour des points de destination (@Shape c'est la
géométrie...)
        with arcpy.da.SearchCursor(stop, ["SHAPE@"]) as cursor:
```



```

i = 1

for element in cursor:

    #Pour chacun des points, je créé un layer de Route (objet
    mémoire) et je prends le résultat pour traitement ultérieur

    outRouteResultObject = arcpy.na.MakeRouteLayer(networkDataset,
"layer_" + str(i), impedance, "", "", "", accumulation)

    outNALayer = outRouteResultObject.getOutput(0)

    #Ici je prends un ensemble de sublayers (qui comprend les stops
    et les routes)

    subLayerNames = arcpy.na.GetNAClassNames(outNALayer)

    stopsLayerName = subLayerNames["Stops"]

    #J'ajoute le point de départ et le point d'arrivée

    arcpy.na.AddLocations(outNALayer, stopsLayerName, origine)

    arcpy.na.AddLocations(outNALayer, stopsLayerName, element)

    #Je produit l'itinéraire
    arcpy.na.Solve(outNALayer, "SKIP")

    #Je fais le tour des sublayers et si c'est celui des routes,
    j'ajoute cela à un featureClass
    for sublayer in outNALayer:
        if sublayer.name == "Routes":
            sublayer.name = sublayer.name + str(i)
            arcpy.Append_management(sublayer, "layer_length")

    i+=1

except:

    e = ErrorHandler()
    e.printStackTrace()
    e.writeToFile(r"C:\Windows\Temp\logfile.txt")

if __name__ == '__main__':
    main()

```

## Module de gestion des erreurs

```
class ErrorHandler:
```

```
    def __init__(self):
```

```
        self.error()
```

```
    def error(self):
```

```
        self.t = datetime.datetime.now()
```

```
        self.e = sys.exc_info()[1]
```

```
        tb = sys.exc_info()[2]
```

```
        tbinfo = traceback.format_tb(tb)[0]
```

```
        self.pymsg = "PYTHON ERRORS:\nTraceback info:\n" + tbinfo
```

```
    def printError(self):
```

```
        arcpy.AddMessage(self.t)
```

```
        arcpy.AddError(self.e.args[0])
```

```
        arcpy.AddMessage(self.pymsg)
```

```
    def writeToFile(self, filename):
```

```
        self.filename = filename
```

```
        f = open(self.filename, 'a')
```

```
        f.write(str(self.t) + '\n')
```

```
        f.write(str(self.e) + '\n')
```

```
        f.write(str(self.pymsg) + '\n')
```

```
        f.write(
```

```
"\n*****next*****  
***** \n")
```

```
        f.close()
```