GeOxygene: An Open 3D Framework for the Development of Geographic Applications

Mickaël Brasebin
Institut Géographique National, Laboratoire COGIT
2 Avenue Pasteur, 94160 Saint-Mandé, France
mickael.brasebin@ign.fr

INTRODUCTION

3D data are more and more available, resulting in increasing the need of 3D GIS. According to [Worboys, 1995], a 3D GIS might be able to model, represent, manage, handle and support decisions based on 3D phenomenon: 3D GIS are not only about visualization but also modelling and analysis.

Some others authors also define 3D GIS by a list of functionalities expected. [Pouliot, 2006] said that a 3D GIS must own the same abilities as a 2D GIS extended to the third dimension. In [Pouliot, 2006] [Poupeau, 2008] and [Rahman, 2001], this list is detailed. A 3D GIS has to provide:

- Creation and edition of geometric primitives in 3D (point, line, polygon and polyhedron),
- Navigation and visualization in a 3D universe,
- Loading and exporting different kinds of data,
- Geometric and semantic querying,
- Geometrical operators : shape transformation (translation, rotation, 3D buffer) and operators between two geometries (Boolean operators, distance operators)

In order to integrate all these functionalities, 3D GIS would be modelled in a unified model [Poupeau, 2008]. It is thus important to have an internationally recognized model able to integrate different data sources, for applying operators and for sharing data and methods. In this work, the ISO model is chosen because of its standardization, reliability and extendibility.

As the third dimension is indispensable to solve 3D specific spatial issues, such as intervisibility, representation of noise pollution or impact of an antenna, the laboratory decided to develop its 3D framework based on the elements presented before in order to lead researches in these domains. This happened because current 3D GIS do not provide all features expected. This framework which contains the main geometrical operators of 3D GIS is designed by extension of GeOxygene: the 2D capitalising platform of the laboratory and implementing according to the ISO model [ISO, 2003].

In a first section, an overview of 3D GIS shows the current state and presents the shortcomings. Then, 3D actual knowledge of the laboratory and the 2D interoperable platform GeOxygene are presented. The implementation of the module and a sample of use are evocated. The paper ends with the future improvements and applications of this platform.

CONTEXT

3D GIS Context

The possible fields of application of 3D GIS are well known. These systems have applications in numerous domains (Territory development, risk and pollution, geology, ocean studies, tourism, telecommunication, transports, geomarketing and defence) where 3D analysis is needed. Since years, the need of 3D GIS is known [Rahman, 2001]; however, current commercials GIS do not provide tools necessary for such studies [Poupeau, 2008]. They are designed as simple 3D viewer. Generally, these systems allow analysis on 2.5D data (such as DTM) and building reconstruction by extrusion, but neither 3D model nor evolved analysis tools are implemented. Analysis is often limited to flooding assessment, visibility maps and sun exposure calculations. Moreover, even in the open source world only few GIS provide 3D analysis and they offer similar possibilities as commercial GIS.

Beside GIS, we can take a look at spatial DBMS. In the open source domain, PostGIS and MySQL enable the storage of 3D coordinates for their available primitives: point, line and polygon (they are based on an implementation of the OGS specifications: SFS 1.1). Only the commercial DBMS Oracle provides the storage of polyhedron according to the B-REP model (composed by a set of polygons forming a shell). Nevertheless, none DBMS include true 3D geometric operators. 3D DMBS can't be used to manage 3D operators.

Thus, 3D GIS are necessary and actual tools do not yet ensure to benefit from a real 3D model. For the need of the laboratory, a 3D framework had to be developed. The decision to use an international 3D model has been taken in order to permit the platform to be reusable by other developers and to integrate different data sources compliant with this model. The ISO model seems to be the best candidate thanks to its position of international standard. To reinforce the notion of universality, this platform is made free in order to enable the sharing of developments, functions and methods between researchers. Moreover it should provide all the functionalities presented in the previous section in order to be used for analyzing 3D phenomenon.

3D skills in the laboratory

Another reason of the development is the willing to have a centralized 3D platform which unifies all the 3D developments in the laboratory. Since 2000, several researches have been led and some paralleled developments have been achieved.

The first steps in this domain were made by De La Losa for his PhD [De La Losa, 2000]. He proposed to choose B-REP as geometric representation, designed a complex 3D topological model and some methods to integrate and corrected data to support this model.

In [Ramos, 2003], the first implementation of 3D has been made. He developed a 3D GIS using a B-REP model for geometry and two topological models (1 for network and 1 for solids). The result was a functional GIS with functionalities such as visibility maps calculation, optimal path queries, spatial index and database storage. As this work was led during an Industrial PhD, the code was provided to EADS.

For his PhD, [Rousseau, 2004] proposed and implemented some methods to correct errors on the relief with available vector data (DTM, error assessment, vector data) in order to minimise the impact of DTM errors on risk simulation.

[Poupeau, 2008], for his PhD work, implemented a prototype named Cristage. It was based on its own geometric model. It enables analysis based on crystallographic concepts. Objects are simplified in their bounding crystalline mesh in order to create a relationship graph for assessing the evolution of the relations between these objects.

Thus, the first aim of the 3D platform is to provide an open core for the studying of 3D phenomenon. But this core will be extended by the different works presented in this section, in order to unify them and to provide more tools which could be reused by other developers. In the next section, the 2D platform of the laboratory is presented before its improvement to 3D.

The GeOxygene platform

GeOxygene [GeOxygene, 2009] is an open source (under LGPL licence) framework designed for the deployment of 2D GIS applications. This platform is used for research developments in the COGIT laboratory of the IGN (French National Mapping Agency). The aim of GeOxygene was to create a common core for the laboratory in order to share developments and to simplify maintenance of codes. GeOxygene ensures interoperability by implementing ISO standards and OGC specification. The standards implemented are:

- ISO 19107 (Spatial schema),
- ISO 19109 (Rules for application schema),
- ISO 19111 (Spatial referencing by coordinates),
- ISO 19115 (Metadata).

GeOxygene is achieved with Java technologies; this choice was made for the portability of applications developed with. Thus, technologies used during the development have respected these 2 constraints: the programming language and to stay open. Spatial operators are provided by external libraries, for example Java Topology Suit API [Vivid Solutions, 2003] is used for basic geometric operations and the “Triangle” library for triangulation [Triangle, 2005].

In GeOxygene data are stored in a PostGIS Database [GeOxygene, 2009]. The library OJB serves to create a reliable mapping between the DBMS and Java Objects. Information can thus be handled without using SQL query [Badard, 2003] [Badard, 2004].

[Pouliot, 2006] wrote that GeOxygene has the potential to become a 3D GIS because the ISO 19107 spatial schema enables 3D modelling. Thanks to the extensibility of the GeOxygene model, it stays possible to use some more specific 3D models (Tin modelling, Voxel …).

The next section presents how the third dimension is integrated in the actual platform and the elements provided by the core for developers.

IMPLEMENTATION OF A 3D MODULE

Presentation of the schema

The first step of the implementation was to find how to fill the technologic gap between 2D and 3D. A new schema of the architecture [Figure 1] of GeOxygene was made in order to evaluate 3D impacts in the whole platform for existing developments. This new schema integrates the old one. Geometric classes and operations are implemented in the core which communicates with output tools (Viewer, servers, etc.).

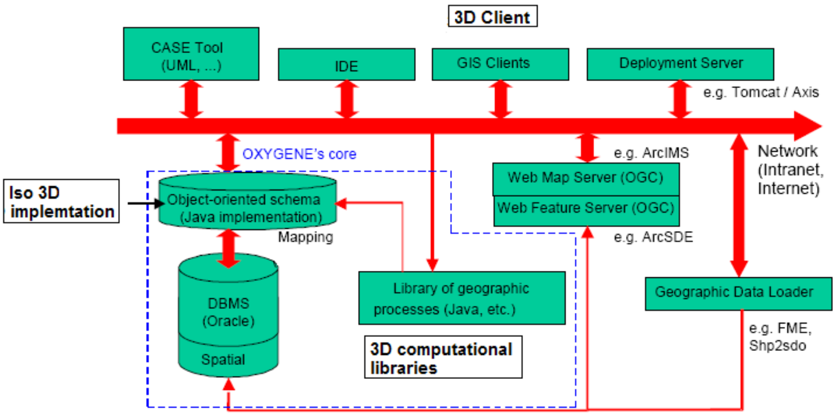

Figure 1: 3D impacts in the general structure of GeOxygene platform.

Thus, adding 3D causes three major modifications:

- Addition of classes of 3D geometries in the core,
- Insertion of 3D computational libraries,
- Integration of a 3D viewer.

An important constraint of the architecture is to integrate 3D in GeOxygene without modifying the existing 2D functionalities. Nevertheless, the 2D researches (semiotic, data matching, web services …) are not actually ready for processing 3D information but might be upgraded in the future thanks to the contribution of their authors.

The spatial schema described in the ISO 1907 standard [ISO, 2003] was, in the 2D version, partially implemented. [Figure 2] shows the geometric model of GeOxygene, but as it was at first a 2D platform, some classes were present but unimplemented because they represent some 3D objects (for example the *GM_Solid* class). The development of the 3D module was a great opportunity to implement these classes in accordance with the ISO standards. Furthermore, some 2D classes needed to be upgraded in order to benefit from 3D coordinates or methods (for example the area calculation for GM_OrientedSurface class). The result is an addition of classes and methods which enable to keep the whole functioning of the platform. The resulting implementation is a core of geometric classes which could be extended by other models compliant with ISO specifications such as CityGML [CityGML, 2008].

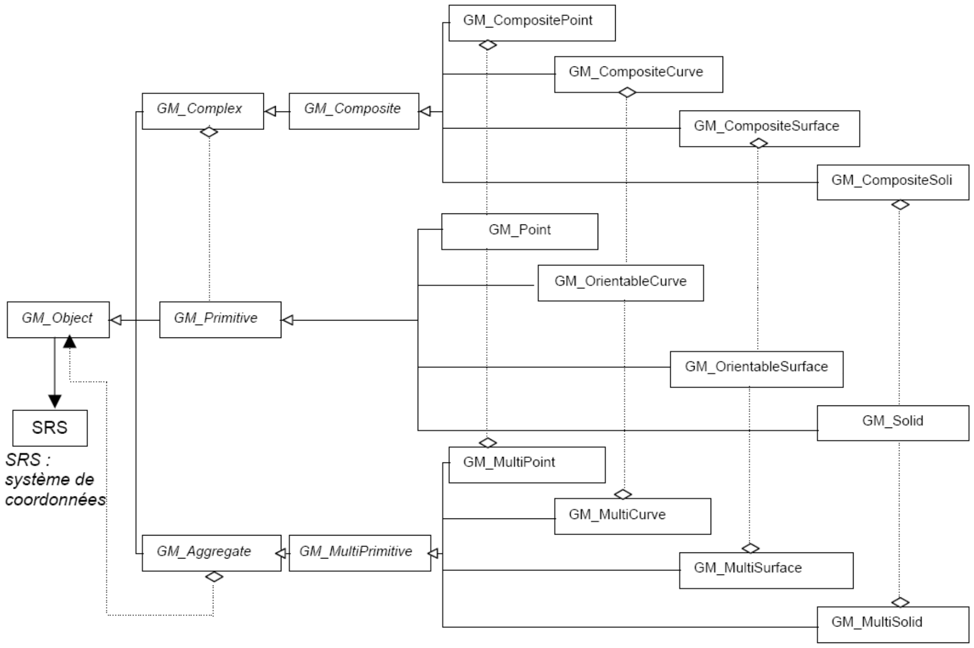

Figure 2: The geometric classes of the ISO model in the core of GeOxygen.

One issue of implementation was the use of spatial operators on objects. In GeOxygene, they are actually provided in 2D by the Java Topology Suit API [Vivid Solutions, 2003] which considers 2D coordinates and is consequently inefficient for 3D geometries. In order to keep the cohabitation between traditional GIS and 3D operations, the actual geometrical methods were modified in order to use the right operator in accordance with the dimension of the coordinates thanks to the function *dimension()* introduced in [ISO, 2003].

The distinction between 2D and 3D operators is guaranteed, as 2D operators are supported by external opensource API, the same process was reused for 3D operators. A particular attention was made to choose OpenSource libraries. The aim is to cover all functionalities presented in the first section of this article. A JNI link of the library of tetrahedral meshes generation “tetgen” [Hang Si, 2006] and the library of Boolean operators JGeom [Frick, 2004] contributed to development geometric operators. Thus, are yet available in this platform:

- Distance measures between 2 solids
- Area calculation
- Volume calculation
- Movement operators (Translation, rotation)
- Convexity measure of solids
- Barycentre calculation
- Boolean operations (Intersection, union, difference) for solids
- Tetrahedral meshes generation for volumes
- Triangulation of 3D surfaces

3D buffer is actually the most important missing operator but will be integrated in the future.

Because it is essential to validate and to test the core developments, a viewer is integrated in the platform. Non-programmer users can thus benefit from existing functionalities without having to develop and programmers can use it to interface its developments. The viewer uses Java3D as 3D-library rendering. The reliability ensured by Sun and its experience assures the library to cross ages and to be a safe start point for 3D visualisation. The 3D environment is integrated in a Swing interface [Figure 3] developed at the laboratory, enhancement of the interface of Cristage, which enables to access all the operations described above. Bases of 2D GIS interface were kept with layers management on the left and tools in the upper bar. Concerning interface, the main difference with 2D GiS is the movements in the space assured by Java3D and available from the central panel.

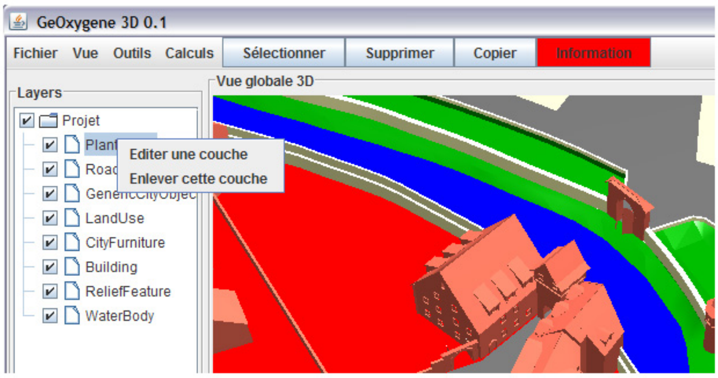

Figure 3: The interface of the 3D viewer.

As Java3D uses its geometric schema optimized for rendering, it is necessary to introduce some rendering classes in order to build a bridge between rendering geometries and model geometries. These rendering classes contain information in term of Java3D objects which traduce the representation of a *FT_Feature* object. [Figure 4] presents the model of 3D rendering. The abstract class *AbstractRepresentation* describes the necessary methods to access Java3D information. Each FT_Feature object which needs to be rendered is associated to one object of an implementing class of *AbstractRepresentation*. Thus, it is possible to benefit from all attributes of a *FT_Feature* object and not only geometric information. Nevertheless, basic representations are available. The implemented classes Objet0D, Objet1D, Objet2D and Objet3D are used to represent the geometry of 3D object according to its dimension and some parameters (Colour, size, transparency). Thus, a *FT_Feature* object with a *GM_Point* geometry is associated to a rendering object of the class Objet0D and represented by a sphere.

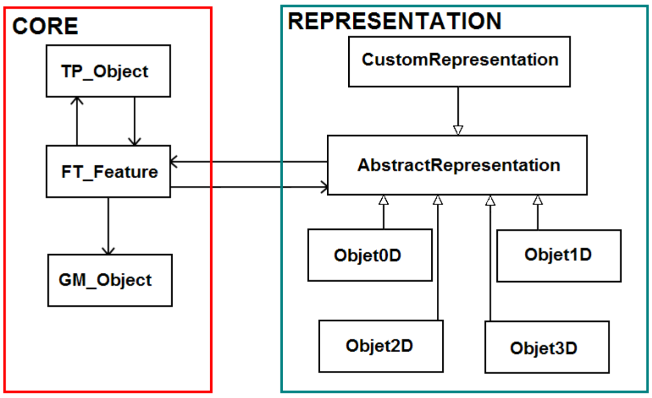

Figure 4: Rendering classes integration in the global schema.

[Figure 5] shows two samples of *CustomRepresentation*. In the first picture, the colour of the wall depends on its orientation, in the second one the heights of buildings are in accordance with their area. It is thus possible to develop some advanced rendering, which depends on semantic, geometric or topologic parameters, by developing implementing classes of *AbstractRepresentation*. These ones are useful to make some thematic 3D map. Furthermore, adding a dimension multiply the opportunities: texts, transparency, enlightenment have a new sense in 3D and thematic representation can benefit from these new rendering parameters.

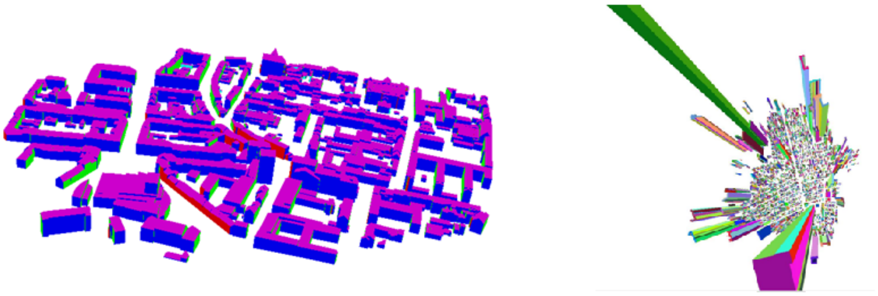

Figure 5: Two samples of uses of *CustomReprsention* implementations.

Storage and loading

In order to enable the use of the platform, several sources of data are loadable. For each available dataset, the process is the same: Data are parsed and transformed into the model of the module and a default representation is build in accordance with geometric information. The extensibility of the model could enable to eventually work on more specific models.

The module can input 3 kinds of file datasets:

- The module was designed at first to use data from IGN. The Bati3D dataset contains buildings of the French territory in 3D. This data in XML format is readable and exportable.
- Data from the IGN 2D database RGE (a dataset composed by vector data in shapefile, DTM and orthophotos) are loadable. 2D vector data are mapped by linear interpolation on the DTM and extruded according to their height attribute.

- CityGML (version 1.0.0) files are readable and exportable. This format which is recognized by the OGC as a standard enables to share 3D data. Actually, only geometric and thematic information is considered.

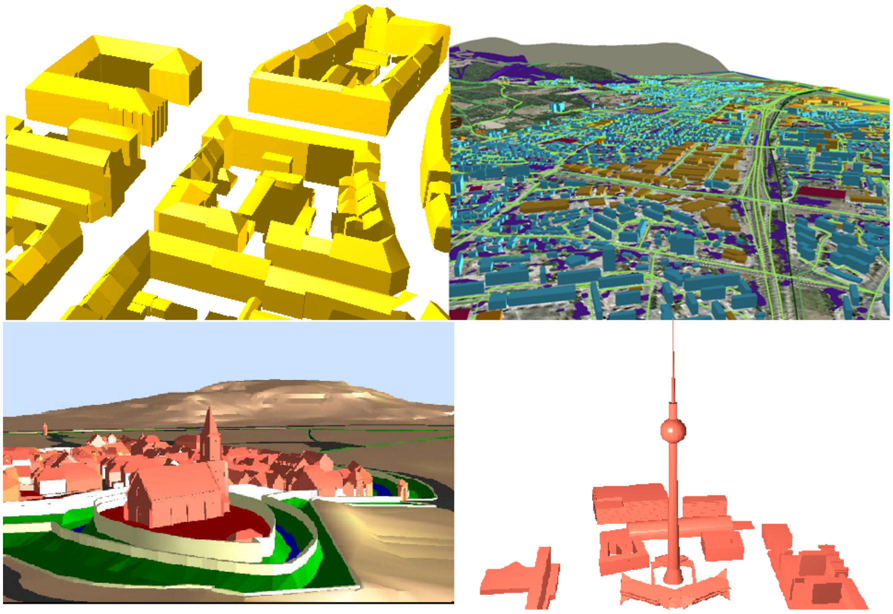

Figure 6: Screenshots of GeOxygene: from left to right from up to down: a Bati 3D dataset, RGE data, and 2 datasets of CityGML.

As in GeOxygene a mapping between Java classes and DBMS PostGIS is ensured by OJB, some trials have been made to also store 3D geometries with this way. The lack of a geometric type "Polyhedron" in PostGIS is a known issue. [Khuan, 2008] and [CampToCamp, 2007] provided some solutions by upgrading the DBMS. In order to find a simpler and faster to implement solution, a different way has been chosen.

For each FT_Feature object stored in the DBMS, an attribute called "dimension" which represents the dimension of its geometry is stored. When a feature represented by a solid geometry is stored, its geometry is stored as a *MultiPolygon* geometry in PostGIS and the attribute "dimension" gets the value 3. Thus, when objects are loaded from DBMS, the real dimension of the object can be detected. This method is quite simple but creates redundant information. It will evolve in the future either if PostGIS takes polyhedron geometries into account or by implementing [Khuan, 2008]'s solution.

The core of the 3D platform is presented. It is usable by anyone. From the interface, a user can load or export data and apply some treatment on it. As all the elements are present as open Java classes, an expert user can development its own application thanks to the core. Thus, he can extend the model, uses the presented operations or the work of other researchers to facilitate its own work.

The next section presents the first development made one the platform using this core.

Implementation of simplifications algorithms

Description

A fast algorithm of 3D simplification has been developed in order to test and validate the platform. Several algorithms have been studied: segmentation of buildings [Thiemann & Sester, 2004], Parallel shift [Forberg, 2004], crystallographic approach [Poupeau, 2007], and based on half plan modelling [Kada, 2007].

As a result an algorithm of simplification based on Kada's solution has been implemented. The general idea of our algorithm was to keep the buffer fusion but the simplified shape is calculated from 2D projections in order to decrease the number of time-consuming 3D operations.

Firstly, faces of the roof are eliminated by calculating the slopes of the polygons of the building. Then, walls are classified by altitude: walls whose maximum altitudes only differ from a certain value (called z-fusion parameter) are gathered into Z-classes. These classes are subdivided into Cycle-classes which only contain walls forming cycles which will be project to the lowest Z of the class. One ring is formed by cycle-class. This process is represented in [Figure 7].

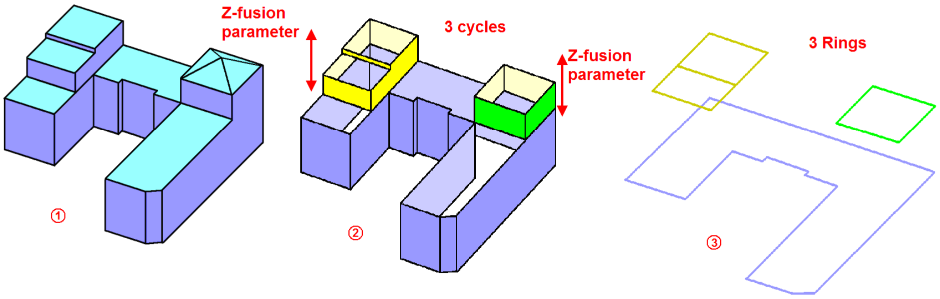

Figure 7: First steps of the simplification: divide the building into rings.

At this step, each ring is treated separately. For each segment of a ring, a buffer of a given width (width buffer parameter) is calculated. Recursively, each buffer intersecting an other merges with it provided that the resulting buffer has a width smaller than a given value (width buffer-fusion parameter). [Figure 8] illustrates the fusion of buffers. This operation is repeated until no more fusion is possible.

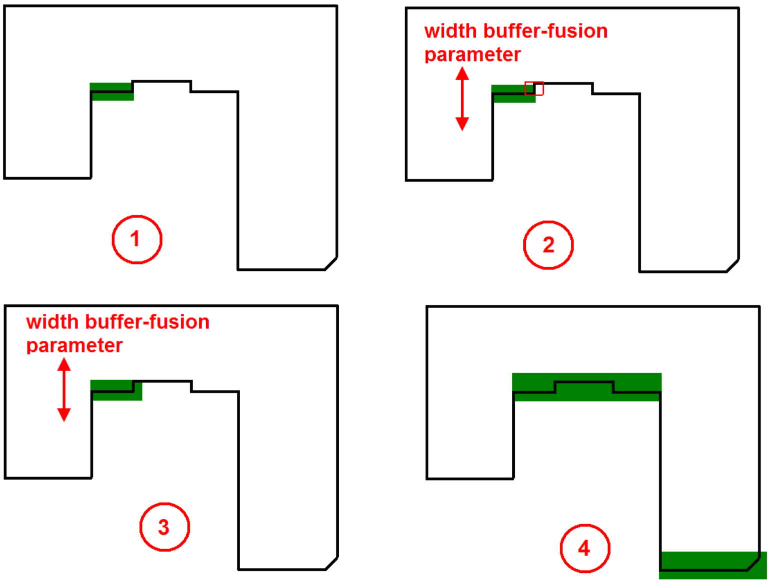

Figure 8: Pictures 1, 2, 3 show how buffers merge, picture 4 shows the final buffers.

Then, for each final buffer, an approximating segment is calculated as the longest central axis of a buffer. These segments are extended to recreate rings. Finally, each ring is extruded of z-fusion parameter in order to make walls. The roof faces are fitted to the new shape by attaching the basis points to nearest point belonging to the top of the new shape when it is possible. That means when they are enough faces to form a roof which cover all the building. A flat roof is created instead when such coverage is impossible [Figure 9].

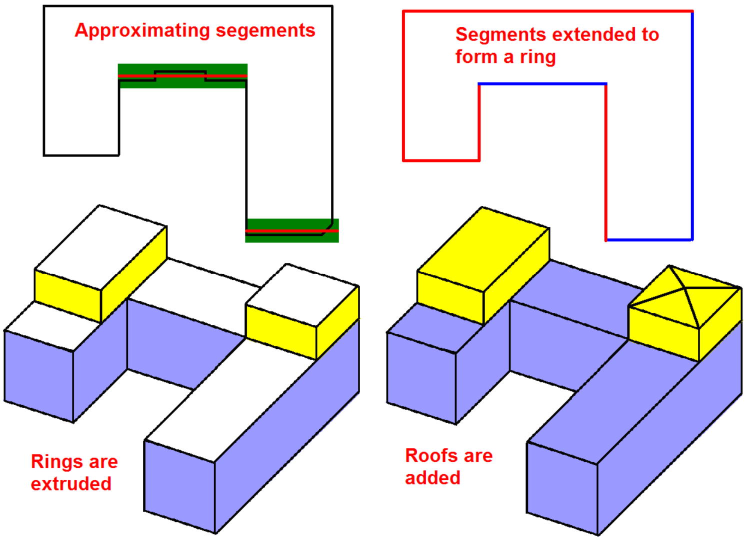

Figure 9: Reconstruction of the final building.

This algorithm does not benefit from all 3D operators. It only proves the adaptability of the platform. Nevertheless, this algorithm enables simplifying effectively and easily building from Bati3D database. Several levels of simplifications are available thanks to the different parameters. Z-fusion parameter has the greatest influence on the number of walls. [Figure 10] shows how this parameter influences the simplified shape.

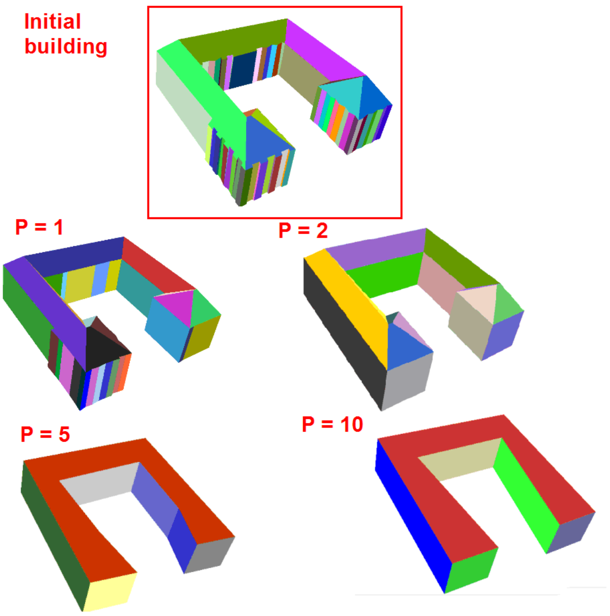

Figure 10: Different simplifications by changing width-fusion parameter (Parameter p). Each face of the building is represented with a random color to better visualize the number of faces.

Discussion

Width-fusion parameter in m	0 (initial building)	1	2	5	10
Number of facets	132	32	25	10	9
Time in ms	---------------	41	41	37	35

Figure 11: Statistics of execution for buildings of the Figure 10.

The simplified shapes have a reduced number of facets and are enough close from the initial appearance of the building. The algorithms works perfectly with buildings of Bati 3D. Buildings with some round shapes can be badly simplified because of the tolerance in roof detection and the difficult of mapping a roof. Furthermore, a squaring algorithm is tested in order to have a more coherent final shape (the two last shapes of Figure 10 need this kind of treatment).

This algorithm was tested on a 2.33 GHz computer with 2 GB Ram memory. It is very fast because the number of 3D operation is very limited. Working in 2D and reconstructing the objet is actually faster than using directly 3D operations.

Furthermore, the parameterization of the algorithm is quite easy because all parameters represent an understandable value in meter. The width-fusion parameter is the most important; it determines the level of simplification in the 2D projection. The Z-fusion parameter determines the number of stages in the final building. The number of facets decreases according to these parameters. Nevertheless, an attention has to be made in the choice of the parameters because larges values simplify the building as a surface.

CONCLUSION AND FUTURE WORKS

The finalization of a functional framework for 3D GIS applications based on ISO 19107 is in progress. It allows operations required for 3D GIS: visualization, geometric operators, data management and integration in a 3D model. This framework is the core of 3D development in the laboratory. The very first research is dealing with simplification: a first algorithm has been produced. Other 3D research presented in section 2 will be progressively integrated in order to enhance the capacity of development. Moreover, as 3D is integrated in the general model of GeOxygene, it might be possible in the future to upgrade 2D research on storage, semiotic and web services for being compatible with 3D. Nowadays, all the features presented her are implemented but their robustness must be improved before a release.

During 2009, this project will be released as an Open Source package in the GeOxygene homepage [GeOxygene, 2009]. Thus, 3D developers can benefit on a core owning all functionalities presented in this paper and an ISO compatible model. Methods respecting ISO standards can be plugged on this module allowing shared developments.

In a future step, new functionalities will be added. A new 2D5 module will manage DTM data, thus relief analysis, hydrologic analysis, DTM adjustments and DTM errors assessments may be provided. 3D topological queries and relationship graphs may be implemented in accordance with [De La Loza, 2000] and [Poupeau, 2008]. Some custom representations will be developed to integrate some pertinent custom representations for rendering relationships between buildings, thematic maps, place-names, etc.

The laboratory has been chosen to bring its 3D knowledge for the project TerraMagna of the business cluster Cap Digital [Cap Digital, 2009]. The goal of this project is to produce a 3D GIS for environment issues and territory development in Paris area. The platform may include the development of the laboratory for this project about two following subjects:

- Advanced geometric operators and a final building simplification algorithm
- Calculation of 3D building bounding solid in accordance with the city rules.

BIBLIOGRAPHY

Badard T. and Braun A., 2004, OXYGENE: A Platform for the Development of Interoperable Geographic Applications and Web Services. In proceedings of the 15th International Workshop on Database and Expert Systems Applications (DEXA'04), IEEE Press, August 30 - September 03, 2004, Zaragoza, Spain, pp. 888-892

Badard T. and Braun A., 2003, OXYGENE: An open framework for the deployment of geographic web services. Im proceedings of the 21st International Cartographic Conference (ACI/ICA), Durban, South Africa, August 10-16, 2003, pp 994-1004

Camp to Camp, 2007, Homepage of the project available at http://postgis3d.blogspot.com/

Cap Digital, 2009, Homepage of the business cluster available at http://www.capdigital.com/

CityGML, 2008, Homepage of the project available at http://www.citygml.org/

De La Losa A., 2000, Modélisation de la troisième dimension dans les bases de données géographiques, Thesis defended at the University of Marne La Vallée.

Forberg A., 2004, Simplification of 3D building data. In proceedings of the 6th CA Workshop on progress in automated map generalisation, Leicester.

Frick Urs, Gerber Samuel, 2004, Boolean Operators for Java 3D

GeOxygene, 2009, homepage-project http://oxygene-project.sourceforge.net.

Hang Si, 2006, TetGen : A Quality Tetrahedral Mesh Generator and Three-Dimensional Delaunay Triangulator

ISO, 2003, Geographic information — Spatial Schema ISO Draft International Standard 19107:2003.

Kada M., 2007, 3D Building Generalisation by Roof Simplification and Typification. in proceedings of ICC2007, the International Cartography Association Conference, Moscow (Russia), 2007.

Khuan C.T., Abdul-Rahman A. and S. Zlatanova, 2008, New 3D data type and topological operations for Geo-DBMS In: V. Coors, M. Rumor, E. Fendel and S. Zlatanova (Eds.); Urban and regional data management: UDMS annual 2007, Taylor & Francis, 2008, pp. 211-222

Pouliot, J., Desgagné, E., Badard, T., Bédard, K., 2006, SIG 3D: Où en sommes-nous et quelles sont les avenues de développement? Géomatique 2006. Montréal, Canada, October 25-26.

Poupeau B., 2007, A crystallographic approach to simplify 3D building. in proceedings of ICC2007, the International Cartography Association Conference, Moscow (Russia), 2007.

Poupeau B., 2008, Analyses et Requêtes de données geographiques 3D – Contributions de la cristallographie Géométrique. PhD in gis attended in 2008 (Université Paris Est)

Rahman, A.A., Pilouk, M. and Zlatanova, S. 2001. The 3D GIS Software Development: global efforts from researchers and vendors. Geoinformation Science Journal, Vol. 1, (13).

Ramos, 2003, Modelling and validating a 3D operational GIS, PhD in gis attended in 2003 (Université de Marne La Vallée)

Rousseaux, 2004, Coherence between altimetric and planimetric data in a context of risks PhD in gis attended in 2004 (Université de Marne La Vallée)

Thiemann F., Sester M., 2004, Segmentation of Buildings for 3D-Generalisation. In proceedings of the 6th CA Workshop on progress in automated map generalisation, Leicester, 2004.

Triangle, 2005, home page of the project : http://www.cs.cmu.edu/~quake/triangle.html

Vivid Solutions, 2003, JTS Topology Suite Homepage http://www.vividsolutions.com/jts/jtshome .htm.

Worboys, M. F., 1995, GIS: a computing perspective, Taylor & Francis, London.