

Data consistency checks for building of 3D model: A case study of Technical University, Delft Campus, The Netherlands



Tarun Ghawana

Integrated Spatial Analytics Consultants
New Delhi
India
tarungh@gmail.com

Sisi Zlatanova

GIS, OTB, Delft University of Technology, Delft
The Netherlands
s.zlatanova@tudelft.nl

Abstract

GIS is changing rapidly its face with the advancement of computing technologies. From merely a 2D representation of real world features, it is moving to present more and more applications in 3D formats. Today 3D GIS is an essential way of handling the spatial data in urban planning. In cities where every inch of land is measured and is consumed by construction of buildings, mere 3D visualisation of the features is not enough. For certain applications which require extensive details of the building architectures, it becomes a prerequisite to maintain a topological accuracy among the features to be represented. This paper describes the process of topologically correcting 2D features to serve as the basis to create topologically correct 3D city models. The 3D model is created from an existing 2D large-scale topographic map and LIDAR height data. Keywords: Urban planning; GIS; 3D modelling; topological validation; DEM; TIN; breaklines

Introduction

Various methods exist to create 3D models and all of them have shortcomings. The idea of the test presented in this paper was to investigate the possibilities of existing software to create a 3D model that complies with the conceptual model of CityGML. CityGML is an OGC standard for presenting real-world features in 3D with different levels of detail (Gröger et al 2008). To perform the tests, we have taken the campus of the Delft University of Technology, Delft, the Netherlands as a case study. The campus features are converted from different formats using geometrical and topological functions of several GIS and spatial DBMS.

We have used geometric and topological functions of three software packages to counter various errors and warnings so as to achieve the maximum topological correctness in the 2D dataset itself. For the time the test was performed, only the CityGML LOD1 (i.e. block or extrusion model) was considered. Using 2D topologically correct model leads to fewer errors while building a topologically correct 3D extrusion model. An example of topological requirement for a 3D extrusion model is of two adjacent buildings having a common wall. If there is improper topological relation between the two building features, the common wall between the two building features will be represented by two different overlapping features in the dataset despite sharing the same geometrical space in reality. Starting from a topologically correct 2D model, it is possible to create a topologically 3D extrusion model (Ledoux and Maijers, 2008).

The process of creating a 2D topologically correct model

The software used was ArcGIS 9.2 (ESRI 2009), Feature Manipulation Engine (FME) (Safe Software 2009), GAWK (freeware software, version 3.1.6) and Oracle Spatial 10g (Oracle Spatial 2009). ArcGIS was also selected due to its availability in many offices. FME and Oracle Spatial were introduced at a later stage only for checking the topological consistency of the features. The data sets used for creating the 3D model were LIDAR data (with a resolution of three points per square meter) and the large-scale (1:1000) topographic map of the Netherlands (GBKN). These data sets were selected because many municipalities have them as base data sets to perform their daily work on planning and maintenance of cities.

The resolution of the LIDAR data can vary per data set but it will not influence the presented approach (only the accuracy of the obtained model). The original LIDAR elevation data points were used for two purposes: to define an approximate value for the height of the buildings and to derive z-value for all the surface features. The points were filtered out using open source software 'GAWK.' To reduce the amount of points to be processed, in the first case only the points with values above three meter were used. In the second case only those point which have elevation values less than 20 cm were retained. Campus buildings and street layers were retrieved from the GBKN data set. The GBKN data set is not object-oriented, i.e. it consists mostly of loose lines, points and (rarely) polygons, which are not even checked against topological rules.

The campus buildings are actually the building footprints on the ground. Street layer features contain a lot of information

about transportation network but for the test, the roads, footpaths, cycle tracks and traffic crossings were taken into consideration. These layers were processed through several geometric procedures in ArcGIS to create features and assigned z-coordinates to them. These features were validated for topological consistency in ArcGIS, FME and Oracle Spatial.

As mentioned above, for this test, we have considered only the LOD1 of CityGML for all kind of features. Since LOD2 and LOD3 require vertical details that cannot be derived from existing 2D maps, we have not considered them in the study. The steps performed in the test can be summarised as follows:

1. Creating features from 2D lines
2. Assigning attributes to the 2D features with respect to the semantic model of CityGML (LOD1)
3. Assigning z-values to all surface features (at this step the z-values are assigned as attributes)
4. Checking the topological consistency of the created surface features
5. Adding z-values as z-coordinate to the terrain features to create the terrain surface
6. Computing a height-value for the buildings which used to extrude the buildings.

These steps are discussed in detail in the following sections.

Creating features from 2D lines

The large-scale map of the Netherlands is not an object-oriented map. It is created basically for visualisation (printing) purposes, in spite of being digital. While looking at the GBKN data set, several problems were encountered: incomplete features (e.g. missing street lines, because they are used for bicycle paths), crossing lines, missing lines, etc. Many building footprints were closed polygons but contained duplicated points or straight lines had many unnecessary intermediate points. GBKN data contains a rich set of features, but for the 3D model only streets and buildings were considered. All the considered features were created manually by copying lines and snapping points to obtain polygons. The features were grouped in two different layers, i.e. 'building' and 'street'. Each of these layers was processed individually to create features (closed polygons).

After this manual operation, many problems still remained. For example, many gaps were encountered between the building and street polygons. The building footprints were corrected first using editing functions of ArcGIS such as 'cut polygons' or 'autocomplete' polygons. These corrections (performed in the layer 'building') introduced incorrectness (e.g. crossing lines) with other layer 'street.' To correct this data discrepancy, building footprints layer was modified in ArcGIS according to the street layer boundaries. The corrections were introduced in the 'building layer' since building footprints were adjusted using some tolerance values (i.e. we expected that the errors are most likely due to tolerances). Some small features of the street layer were deleted to maintain the geometrical consistency between the two datasets.

As soon as all the building footprints and streets were created, the layers were integrated to be able to delineate the remaining area as grass. First the streets were used to create a non-overlapping coverage and new 'grass' polygons were created in all the spaces between the streets (for this we again used the editing and merging functions of ArcGIS). Then the layer 'buildings' was used to create polygonal holes in the layer with the non-overlapping grass features. This means that we erased the feature space of buildings from the layer 'grass' with non-overlapping features. The result was a single feature layer having only the grass features.

Adding attributes to the features

Some attributes were created as columns in the feature layers associated tables. The attributes introduced are building heights, building name, road or street name. For making the layers compatible with CityGML LOD 1, some other attributes for buildings are building type, function, usage, construction and demolition year, while for transportation class these are traffic area, street name, street auxiliary area.

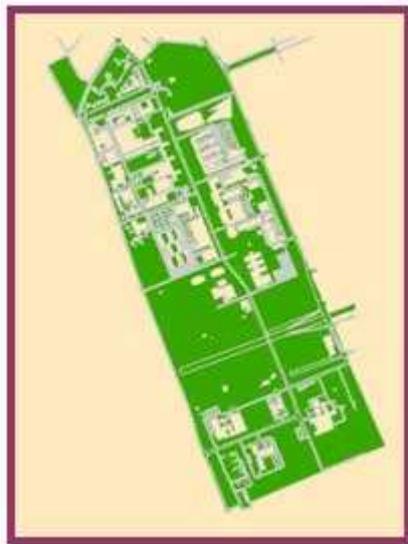


Figure 1. Streets polygonal layer without building footprints

Adding z-values (as attributes) to surface objects and buildings

At this stage of the processing, street and building layers contained features only with 2D values. To provide the z-values, the filtered LIDAR points were used. ArcGIS Spatial Analyst provides interpolation methods to generate continuous surfaces from the LIDAR points. These continuous surfaces are generally stored in the raster format in ArcGIS. The raster surfaces divide the feature space in grids of a given resolution. The filtered points were used to create two raster surfaces of 1 meter and 5 meter resolution. 3D Analyst extension of ArcGIS software provides the facility to convert 2D features to 3D by associating the numerous point elevations values to different segments from a raster surface. In this case, we used 1 meter resolution raster surface. The values from the raster surface were added to the geometry of each vertex of each feature as its z attribute. This attaches the base of the feature closely with the surface layer.

Topological consistencies of feature layers

This task was performed at several levels in 2D environment of different software. ArcGIS topological functions were used initially on individual layers to provide the first check of topological inconsistencies. However, after a manual check of the merged campus layer features, many errors were discovered. It was decided to use topological tools of the FME software and Oracle Spatial 11g topological model allowed to highlight more inconsistencies which could not be detected by ArcGIS. The sections below discuss the topology check in the three software packages ArcGIS, FME and Oracle Spatial.

ArcGIS Topology

Initially, in ArcGIS, topology was built for two layers of streets and buildings. Topological rules implied no overlapping of the features, whether in the same layer or between different layers (see Figure 2). Using topological functions, errors were listed and finally removed.

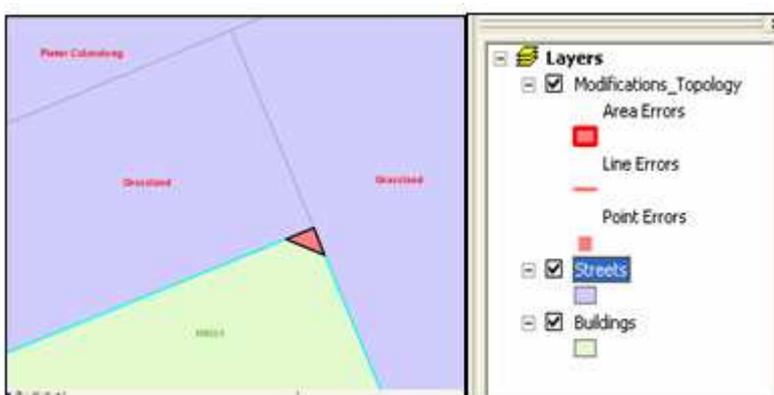
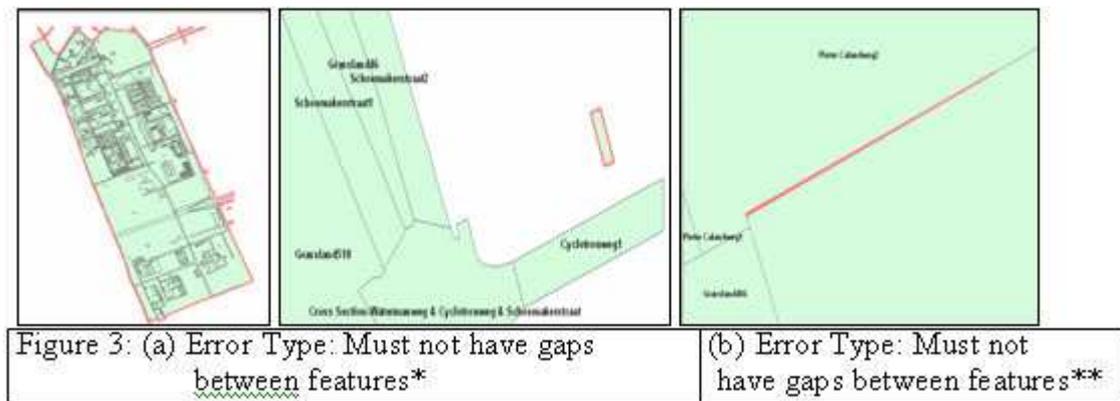


Figure 2: Error type: Must not overlap with other layer features Once we got the topologically corrected individual layers, we used ArcGIS geoprocessing functionalities to create merged layers of all campus features (i.e. buildings, streets with grass). The merged layer created from ArcGIS topologically corrected buildings layer and ArcGIS topologically corrected streets layer was used for another ArcGIS topological check. This was necessary to check the gaps between the features

in the same layer. This time, new topological rules were created which implied 'no overlapping' or 'no gap' between the features in the same layer. For the gaps between the adjacent features as shown in the figure 3 (b), ArcGIS option of creating new features was accepted.



* Marked as exception in these particular
 ** Gaps filled by creating additional features cases

These features were later merged manually with one of the original feature. The original feature to be merged was decided by visualisation at different zoom levels. The rule of 'no gap between the features in the same layer' was not used earlier since it could have led to numerous errors since buildings or streets do not have a complete coverage. In the merged layer 'topology' (buildings, streets and grass), there were few such cases and were marked as exceptions (see the Figure 3a).

FME topology creation

To check the topology created with ArcGIS, we used the 'topology builder' function of FME. The FME workbench was used to create a model in which the ArcGIS topology-corrected merged campus layer was used as input shapefile. After executing the topology builder in FME we created an output shape file which was compared with the input one. The overlay of the input and output shapefiles showed that some input polygon features were not created in the output shapefiles. On checking out for the possible reason, we found that these were having multiple features included in one record. This means that one feature is an aggregation of many features. This happened around those street layer polygons which were having large grasslands or other surfaces included. After merging the buildings layer with this street layer, the merged layer still considers those polygons as single record in the attribute table. ArcGIS topology rules of non-overlapping and no gaps between the features could not detect this kind of behaviour. But FME Model had detected them as improper topological features and resulted datasets contained less faces.

To solve this problem, we edited the input shapefile in ArcGIS. One of the polygons which were not created in the output datasets was selected. The tool 'breaking multipart feature into single feature' was used. This created more new features. Again, we ran the FME model with the edited input shapefile. This time we found a larger number of output faces (Figure 4).

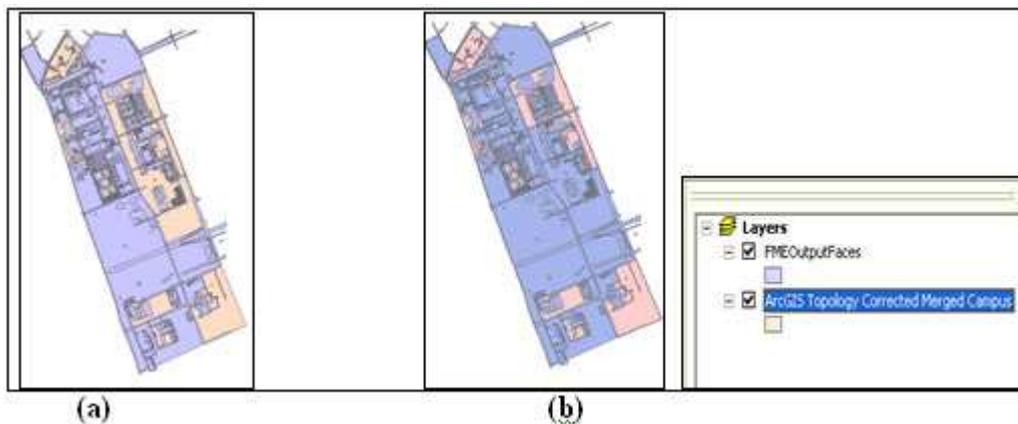


Figure 4: Overlay of FME Output Faces and ArcGIS Topology Corrected Merged Campus Features after first (a) & second (b) iteration of topology building in FME

After several iterations, the FME produced the topologically correct output data set, which was used to create tables with geometry in Oracle Spatial.

Oracle Spatial Topology Validation

Last checks of the topology were made with Oracle Spatial. Oracle Spatial topology model needs three separate oracle tables with three different geometries, i.e. 'node', 'edge' and 'face' (Oracle Spatial, 2003). 'Node' table contains data with point geometry, 'edge' table contains data with line geometry and 'face' table contains data with polygon geometry. To be able to fill in these tables, three shapefiles were also generated with the same types of geometries in FME. In addition to this, a 'universe polygon' shapefile was created, which contained all external boundary features of original layer but as a single feature polygon. This file is important to define the right and left faces for each edge.

It is essential to insert a record in the newly created oracle table 'Face_input_table.' This is necessary to create the universe face of the table. For this purpose, the following SQL statement was used:

```
"Insert into face_input_table (face_id) values (0);"
"Commit;"
```

and the following command to check the minimum face_id of the table:-

```
"Select min(face_id) from face_input_table;"
```

It should be noted that topology in Oracle Spatial has to be created by the user. We have used 4 oracle scripts (Oosterom et al, 2005) in SQL Developer to create the intermediate tables and to check the geometric and topological consistency of the datasets. The first two scripts create intermediate tables and check them for the presence of proper nodes, edges and faces. The third script checks the geometrical consistency of the datasets while the final script topologically validates the datasets. On running the second script, which checks for the existence of the proper faces and edges, we got the error (Figure 6 a, b)

"Not enough points left in the edge after rounding" leading to the error:
"- No edge found for face..."

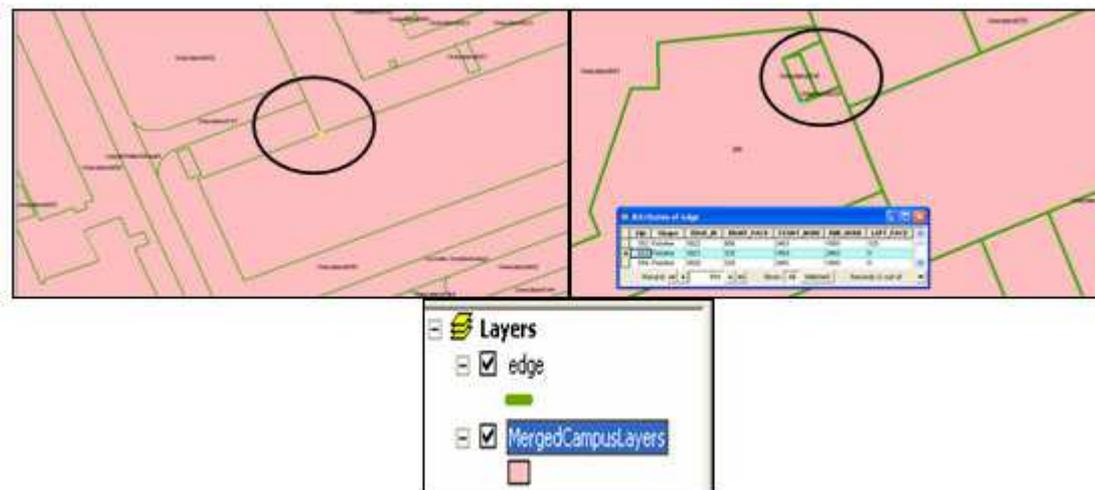


Figure 6: (a) & (b) Edges without enough points to create faces

These errors were removed by going back to the ArcGIS and editing the input shapefile for the particular node / edge / face (Figure 7). These errors highlighted topological inconsistencies, which could not be caught during the previous topological checks.

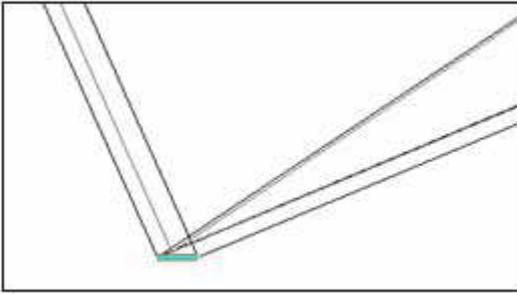


Figure 7: Errors highlighted in the Oracle Spatial Validation Process

Finally we got the Oracle Spatial 'call complete' after many iterations, which confirmed the building of correct 2D topology.

The whole process of topological consistency check made it clear that topological consistency checks should be used very carefully. The different conditions imposed in software can highlight the errors in features which may have to be removed. However, it may also ask to perform some error removal which may be technically wrong but in real world it is necessary to be in the current position. For example, the condition of 'no gap between features' in ArcGIS can ask to remove this error between two features which are not adjacent in a merged layer. In addition, it is also clear that the software packages have different topological rules, which implies that an algorithm used in one software package will not necessarily highlight and remove all the inconsistencies. ArcGIS provides good interface to check and correct inconsistencies. It has a list of errors and also possibilities to zoom on it to find errors quickly. However, manual checks on the data after error removals still showed some inconsistencies, which indicates that the topological rules of ArcGIS cannot detect all the inconsistencies.

FME topological functions did detect more inconsistencies; however, FME Workbench did not describe the listed errors in a clear and convenient manner. Furthermore, graphical interface to locate and edit them is missing. Therefore, output files were checked and edited again in ArcGIS. The Oracle spatial validation function (after topology was built) discovers many new inconsistencies, which were not detected by ArcGIS and FME. The errors were visualised and corrected again in ArcGIS, since Oracle does not provide an interface for editing. However, there is one concern with the usage of the Oracle Spatial topology validation function, ie it does not provide a list with all the errors. The errors are reported one by one, which may lead to doubtful situations sometimes, i.e. an error that is corrected, might need to be corrected again after few steps.

Adding Z-values to terrain

To create the surface features as 3D terrain features, the merged layer having surface elevations as z attributes was used as breaklines to create different TINs. Figure 8 shows the different TINs structured with different resolution using (or not) the layers with the geometry as breaklines. The effect of the breaklines was also different. TIN created directly from LIDAR data points (not shown here) were denser than from the TINs made from the raster surfaces.

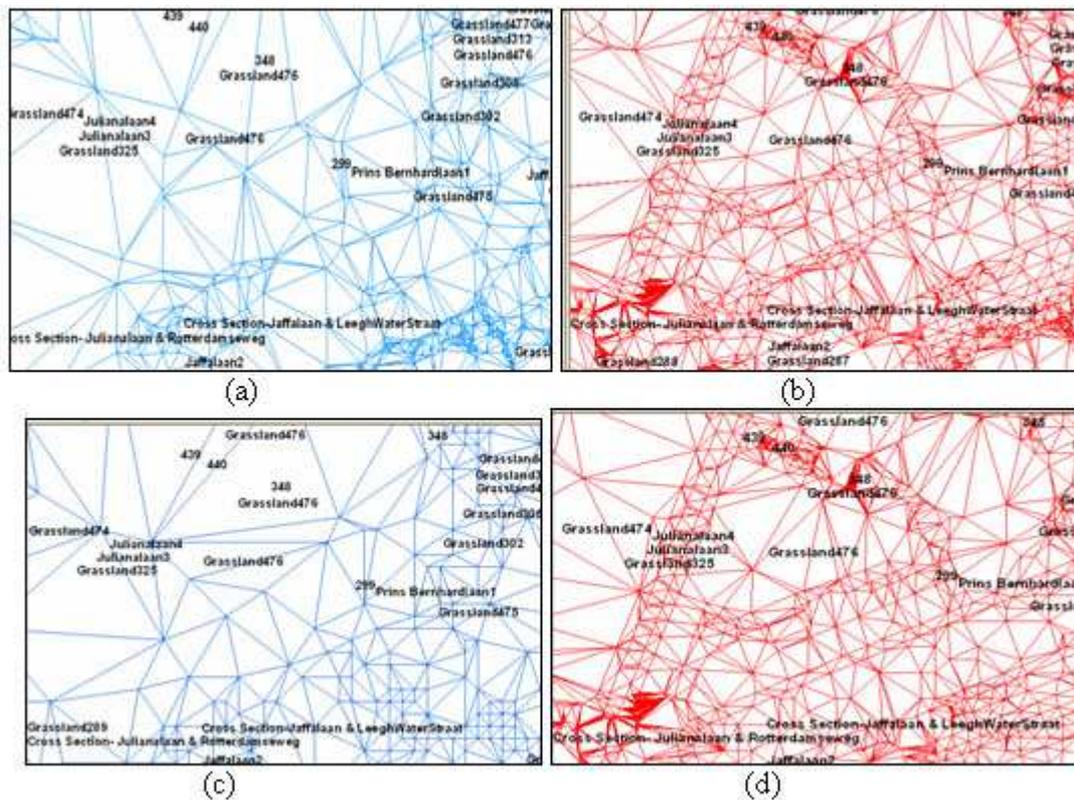


Figure 8 (a) TIN made from 1 meter interval Raster (without breaklines)
 (b) TIN made from 1 meter interval Raster (with Merged Campus as breaklines)
 (c) TIN made from 5 meter interval Raster (without breaklines)
 (d) TIN made from 5 meter interval Raster (with Merged Campus as breaklines)

3D reconstruction of buildings (LOD1)

To create the 3D extrusion model, we again used ArcScene. As immediately realised, ArcGIS extrusion procedure does not create topologically correct 3D buildings. Buildings that share a wall have this wall twice. If the buildings are of different height, the large wall does not contain a hole delineating the smaller wall. But the created 3D model is still suitable for visualisation and navigation. ArcScene allows to fly through the 3D model and to visually analyse it using rotation, tilting and zoom features. It is linked with the surface TIN using it as base height.

Conclusions and recommendations

We have presented our workflow to create a 3D model from large scale topographic map and LIDAR data. We created a topologically correct 2D data set and then we have assigned z-coordinates to be used for creation of the 3D model (LOD1 according to CityGML). The software that we have used was either commercially available or freeware. The general conclusion of this experiment is that the creation of topologically correct models is still a challenge. It should be also noted that if topological requirements to the model are not that strict, the whole creation of 3D model according to LOD1 of CityGML can be completed with the standard functions of the software packages we have tested, i.e. ArcGIS or FME. Although not discussed here, FME has the functions to create a model similar to the one presented here. Much more programming efforts will be required to perform the whole process using only Oracle Spatial 10g.

Results obtained through the different geometrical and topological processes on the spatial data features have confirmed that it is possible to create a consistent dataset only in 2D and 2.5D. Starting from a spaghetti data set, it was needed to perform quite many manual operations to construct the needed features. Quite elaborate programming efforts will be needed to automate completely this process. A very problematic step was assigning attribute values to all the features. Although some of these attributes could be obtained from the original layers of the topographic map, the work is predominantly manual. Much of the information required by the CityGML semantic model needs to be introduced in the tables manually.

Our experiment showed that the topology checks use different algorithms in three different software we have tested. This resulted in different outputs and errors removal. ArcGIS topology was found to be having good interface to locate the errors but not sufficient enough to remove all the errors. FME Workbench topological function detected more topological inconsistencies, but Oracle Spatial topological and geometrical validation was the strictest. This once again confirms the advantages of topological models against geometrical. A major disadvantage of Oracle Spatial is the lack of

appropriate interface to correct the errors. Perhaps a combination between ArcGIS and Oracle Spatial or FME and Oracle Spatial could be the best option for correcting errors.

Another critical step is assigning z-values to different features. In our case, we have used the LIDAR data sets for needed z-values and for the height of buildings. The derived heights were stored as attributes and then used to create the surfaces and the 3D buildings. This work could be performed in ArcGIS (and FME). The approximate (media) height of the buildings can be useful only for LOD1. If LOD2 is aimed, other approaches have to be investigated in both directions i.e. how to derive the surfaces on the roof and how to store these surfaces with the extruded model. While assigning the z-values to the surface features and creating the TIN, it became apparent that the different resolution has different impact of breaklines introduced in the surface.

The initial idea of this test was to create a topological 3D model which ensures a connection with the base surfaces as well as the correctness of the extruded buildings. We could experience that refinement of 2D data can help immensely in producing topologically correct model. However, it was not possible to find commercial or freeware tools that could create a consistent 3D data model. Therefore, section GIST technology has begun the development of open software that creates a correct extrusion model from 2D topologically corrected data (Ledoux and Meijers, 2008).

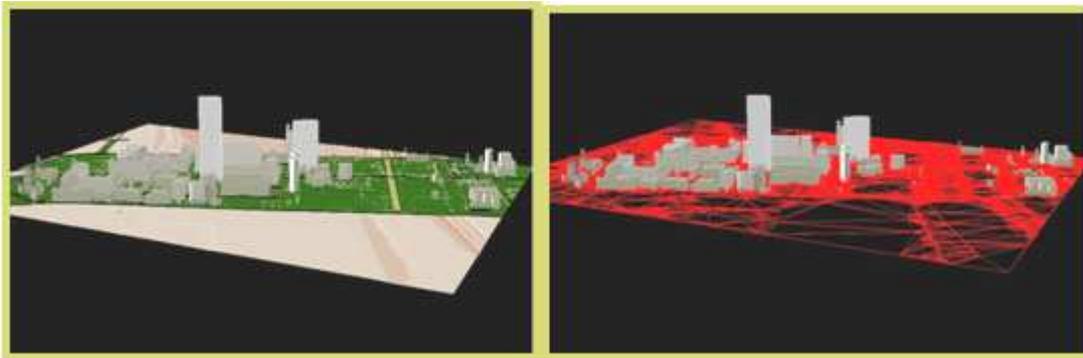


Figure 9: 3D reconstruction of Buildings in ArcScene

References

- ESRI, 2009, ArcGIS: A complete Integrated System, available at: www.esri.com/software/arcgis/index.html (last accessed January 2010)
- Gröger, G., T. H. Kolbe, A. Czerwinski and C. Nagel, 2008, OpenGIS® City Geography Markup Language (CityGML) Encoding Standard, OGC standard, available at www.opengeospatial.org (last accessed January 2010)
- Ledoux, H. and M. Meijers, 2009, Extruding building footprints to create topologically consistent 3D city models, In: Krek, Rumor, Zlatanova, Fendel (Eds.); Urban and Regional Data Management, UDMS Annuals 2009, CRC, pp. 39-48
- Oosterom, P. van, T. Tijssen and F. Penninga, 2005, Topology storage and use in the context of consistent data management, GIST Report No. 33, Delft, 2005, 54 p. available at www.gdmc.nl/publications (last accessed January 2010).
- Oracle Spatial, 2009, Oracle Database 10g Release 1 Spatial and Locator Technical Information, , available at: http://www.oracle.com/technology/products/spatial/10gr1_tech_info.html (last accessed January 2010)
- Oracle Spatial, 2003, Oracle Spatial Topology Data Model, Specifications, available at http://download.oracle.com/docs/cd/B19306_01/appdev.102/b14256/sdo_prttopo.htm, (last accessed January 2010)
- Safe Software, 2009, FME Desktop overview, available at <http://www.safe.com/products/desktop/overview.php> (last accessed January 2010)