9. CONCLUSION AND FUTURE WORK

Overall, the mapping process developed in this research is capable of handling arbitrarily large-sized CityGML documents given a reasonable amount of memory and storage allocation. It facilitates the seamless interaction between unmarshalling CityGML elements to Java objects with the help of citygml4j and mapping Java objects to graph entities using Neo4j's Java Core API.

The matching process can disambiguate the common syntactic ambiguities existing in GML between XLink and in-line object declarations. All changes made to the old city model in Section 8 were identified correctly. In addition, although LOD2 data were used in the test scenarios, changes in other LODs can also be detected. Moreover, geometric objects such as points, line segments, polygons, surfaces, etc. can be matched correctly even with altered identifiers. Furthermore, buildings can be organized in a grid layout or an R-tree based on their spatial allocations. These strategies offer a noticeable boost in overall performance.

Found deviations are attached to their respective sources in the graph database and can be transformed to WFS requests complying with the official OGC standards. In case of complex XML properties, such as CityGML generic attributes and external references, although the update procedures can be formally represented by graphs, the ordinary WFS is not expressive enough in such scenarios. Thus, vendor-specific extensions allowed by the WFS standard, such as defined by the virtualcityWFS, can be employed.

Some improvements and extensions are possible in the near future. For instance, momentarily, only the modules `Building` and `Appearance` are implemented. Other CityGML modules, like `CityFurniture`, `Transportation`, `Bridge`, `Tunnel`, etc. can be included in the future. Moreover, it is previously assumed that both CityGML input documents are provided in the same spatial reference system, which is not always the case in practice. Therefore, one future task is to integrate the transformation between different spatial reference systems in the implementation. In addition, more thorough tests are required to evaluate application outputs against all different types of geometrical deviations. Finally, the methods and algorithms proposed in this research can be extended and applied to enable a version control system for collaborative work in modelling and storing digital 3D city models in the future (Chaturvedi et al., 2015).

References

[1] M. Bakillah, Y. Bédard, M. A. Mostafavi, and J. Brodeur. "SIM-NET: A View-Based Semantic Similarity Model for Ad Hoc Networks of Geospatial Databases." In: *T. GIS* 13.5-6 (2009).

[2] M. d. Berg, O. Cheong, M. v. Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications*. 3rd ed. Springer, 2008.

[3] T. Bray, J. Paoli, C. M. Sperberg-McQueen, E. Maler, and F. Yergeau. *Extensible Markup Language (XML) 1.0*. Fifth edition. W3C. Nov. 2008.

[4] K. Chaturvedi, C. S. Smyth, G. Gesquière, T. Kutzner, and T. H. Kolbe. "Managing versions and history within semantic 3D city models for the next generation of CityGML." In: *Selected papers from the 3D GeoInfo 2015 Conference*. Ed. by A. A. Rahman. Springer, 2015.

[5] S. Cox, P. Daisey, R. Lake, C. Portele, and A. Whiteside. *OpenGIS Geography Markup Language (GML) Implementation Specification*. Specification OGC 03-105r1. Version 3.1.1. Open Geospatial Consortium, 2004.

[6] S. DeRose, E. Maler, D. Orchard, and N. Walsh. *XML Linking Language (XLink) Version 1.1*. W3C. May 2010.

[7] M. J. Egenhofer and R. D. Franzosa. "Point-set topological spatial relations." In: *International Journal of Geographical Information Systems* 5.2 (1991).

[8] M. J. Egenhofer and J. Herring. *Categorizing binary topological relations between regions, lines, and points in geographic databases*. Technical report. Department of Surveying Engineering, University of Maine, 1991.

[9] G. Gröger. *Modeling Guide for 3D Objects - Part 1: Basics (Rules for Validating GML Geometries in CityGML)*. `http://en.wiki.quality.sig3d.org/index.php/Modeling`. Version 0.6.0. Accessed: 2017-03-01. SIG3D - Special Interest Group 3D, Dec. 15, 2010.

[10] G. Gröger, T. H. Kolbe, C. Nagel, and K.-H. Häfele. *OpenGIS(R) City Geography Markup Language (CityGML) Encoding Standard*. Version: 2.0.0. OGC. Apr. 2012.

[11] A. Guttman. "R-trees: A Dynamic Index Structure for Spatial Searching." In: *Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data*. SIGMOD '84. ACM, 1984.

[12] J. W. Hunt and M. D. McIlroy. *An Algorithm for Differential File Comparison*. Computing Science Technical Report. Bell Laboratories, June 1976.

[13] C. Nagel. *citygml4j - The Open Source Java API for CityGML*. `https://github.com/citygml4j/citygml4j`. Version 2.4.3. Accessed: 2017-03-01. 2017.

[14] G. Navratil, R. Bulbul, and A. U. Frank. "Maintainable 3D Models of Cities." In: *Proceedings of the 15 International Conference on Urban Planning, Regional Development and Information Society*. Real CORP, 2010.

[15] S. H. Nguyen. "Spatio-semantic Comparison of 3D City Models in CityGML using a Graph Database." Master's thesis. Department of Informatics, Technical University of Munich, May 2017.

[16] A. Olteanu, S. Mustière, and A. Ruas. "Matching imperfect spatial data." In: *Caetano, M., Painho, M.(Es.), Proceedings of 7th International Symposium on Spatial Accuracy Assessment in Natural Resources and Environmental Sciences. Lisbon*. 2006.

[17] Oracle Corporation. *Java Architecture for XML Binding (JAXB)*. `http://docs.oracle.com/javase/tutorial/jaxb/intro/arch.html`. Tutorial. Accessed: 2017-03-01. 2015.

[18] R. Redweik and T. Becker. "Change Detection in CityGML Documents." In: *3D Geoinformation Science: The Selected Papers of the 3D GeoInfo 2014*. Springer, 2015.

[19] H. Tveite. "An accuracy assessment method for geographical line data sets based on buffering." In: *International journal of geographical information science* 13.1 (1999).

[20] P. A. Vretanos. *OGC® Web Feature Service 2.0 Interface Standard*. OGC® Standard 09-025r2. Version 2.0.2. Open Geospatial Consortium, July 2014.

[21] Y. Wang, D. J. DeWitt, and J. Y. Cai. "X-Diff: an effective change detection algorithm for XML documents." In: *Data Engineering, 2003. Proceedings. 19th International Conference*. Mar. 2003.