TOPOLOGICALLY CONSISTENT MODELS FOR EFFICIENT BIG GEO-SPATIO-TEMPORAL DATA DISTRIBUTION

M. W. Jahn[a], P. E. Bradley[b], M. Al Doori[c], M. Breunig[a]

[a] KIT - Karlsruhe Institute of Technology - Geodetic Institute, Germany
(markus.jahn, martin.breunig)@kit.edu
[b] KIT - Karlsruhe Institute of Technology - Institute of Photogrammetry and Remote Sensing, Germany - erik.bradley@kit.edu
[c] AUD - American University of Dubai - Department of Electrical and Computer Engineering, United Arab Emirates - maldoori@aud.edu

Commission IV, WG 7

KEY WORDS: big geo-spatial data, big geo-spatio-temporal data, nd-databases, nd-topology models, topological consistency, parallel databases, parallel geo-analytics and -simulations

ABSTRACT:

Geo-spatio-temporal topology models are likely to become a key concept to check the consistency of 3D (spatial space) and 4D (spatial + temporal space) models for emerging GIS applications such as subsurface reservoir modelling or the simulation of energy and water supply of mega or smart cities. Furthermore, the data management for complex models consisting of big geo-spatial data is a challenge for GIS and geo-database research. General challenges, concepts, and techniques of big geo-spatial data management are presented. In this paper we introduce a sound mathematical approach for a topologically consistent geo-spatio-temporal model based on the concept of the *incidence graph*. We redesign DB4GeO, our service-based geo-spatio-temporal database architecture, on the way to the parallel management of massive geo-spatial data. Approaches for a new geo-spatio-temporal and object model of DB4GeO meeting the requirements of big geo-spatial data are discussed in detail. Finally, a conclusion and outlook on our future research are given on the way to support the processing of geo-analytics and -simulations in a parallel and distributed system environment.

1. INTRODUCTION AND RELATED WORK

Geo-spatio-temporal or in general nd-topology models are very useful to automatically check the consistency of polytope models in emerging GIS and CAD applications such as Building Information Modelling (BIM), city and infrastructure planning, subsurface reservoir modelling, and BIM-GIS integration. Furthermore, they will help to keep the overview by navigating through complex 3D (spatial space) and 4D (spatial + temporal space) above- or sub-surface models consisting of big geo-spatial data generated from terra- or petabytes of point cloud data. There is no doubt that the consistent storage and efficient retrieval of big geo-spatial data will significantly improve today's usability of geo-database management systems for emerging applications such as reservoir modelling or the computation of energy and water supply in future (smart) mega cities. Furthermore, unless a very efficient indexing of the topology is possible, topological data models are not well suited for handling big geo-spatial data streams (Li et al., 2016). This, however is a challenge, as the worst-case storage complexity of the topology is quadratic in the number of objects (Bradley and Paul, 2010).

This is the first in a series of papers being concerned with the issue of processing big geo-spatial data. The aim of this article is to present a theoretical model for spatio-temporal temporal online analytic processing (ST-TOLAP) which also supports simulation analytics. The following papers deal with implementation and tests. We here focus on a mathematical model for topological consistency checking and on the preparation of DB4GeO (Breunig et al., 2016), our service-based geo-spatio-temporal database architecture, to handle big geo-spatial data with embedded complex analytics and simulations on parallel database system architectures. A brief discussion about big data in conjunction with GIS is given in (Goodchild, 2016). A generic model for spatio-temporal data is presented in (Oosterom et al., 2002). In (Bradley and Paul, 2010) a topological model for data without restriction on dimension (including spatial or temporal) is presented based on the notion of Alexandrov topology. The rest of the paper is organized as follows: Section 2 gives a general overview on big geo-spatial data challenges and introduces big data concepts and techniques including spatio-temporal data management, workflows, online programming paradigms and NoSQL systems. Section 3 highlights topological consistency checking and introduces a theoretical model based on the concept of the *incidence graph*. Section 4 presents the redesinged object model of DB4GeO wich now contains the topological consistency constraint in order to support a topological big geo-spatio-temporal data distribution within the computer cluster. The concept of topological consistency from Section 3 is realised as a geo-spatio-temporal *incidence graph*. This *incidence graph* supports the checking of topological consistency on geo-spatio-temporal polytopes in particular, and nd-objects in general. Finally, Section 5 gives a conclusion and an outlook on our future research.

2. BIG DATA CHALLENGES, CONCEPTS AND TECHNIQUES

In this section, we will aim to reflect on general issues of big data management and processing and current challenges to the field caused by the ever-growing GIS and remote sensing data sets.

2.1 Challenges

Over the past years new technologies have been developed to handle large amounts of daily produced structured and unstructured data with more or less high user interaction. Those technologies

may be used to handle spatial archives and analytics, especially concerning geo-spatio-temporal data. The five big V's defining big data (value, variety, veracity, velocity and volume) suit to geo-spatio-temporal data and their common use cases. Those data sets consist of large amounts of information concerning moving, morphing, geo-spatio-temporal topology and the trend of attributes of the given geographic objects (high volume). Geo-spatio-temporal data are structured, processed and analyzed by scientists of a broad variety of different expertise, working tightly together to generate added value of measured data (value). Because of the number of scientists, their different use cases and missing standards geo-spatio-temporal data could be seen as relativly heterogenous (variety). In any case, the data needs to be analyzed quickly (velocity) and needs to be trustable (veracity). It is to mention that all geographic information is subject to uncertainty (Goodchild, 2016). So when dealing with veracity of big geo-spatio-temporal data we have to deal with uncertainty information about the data.

Due to the complexity and the large scale of GIS and remote sensing data, it is desirable to identify and analyse geographic objects when designing complex distributed systems. For example, in urban planning, there is an interest in current land cover and land use data objects at various spatial and temporal hierarchies. Easy and efficient programing of these systems can be challenging (Ma et al., 2014). GIS experts are focused on accurate mappings and handling relevant big data related to human and natural risks. This data can be unstructured, which increases the challenge of extracting meaningful content out of it, in particular aggregation and correlation of multisource real time data that includes ground surveys and remote sensing images. One possible approach is to utilise a combination of different resolutions to enable the analysis of areas at semantic level rather than to focus on one particular resolution.

The availability of a wide range of analysis methods facilitates the success of a data mining task, however this relies heavily on the data and GIS experts ability to configure the appropriate selected algorithm. This challenge requires familiarity and knowledge of these algorithms. Big data being temporally distorted due to changes of the urban state or agronomical distortions may cause cyclic data changes, however the class theme may stay persistent.

The differing backgrounds of GIS experts and variability of computing skills can raise another challenge in big data management. This variability can generate what is known as a semantic gap caused by the lack of homogeneity between low-level information such as information extracted from an image and high-level information such as urban experts analysis (Assuncao et al., 2015). During the last decade of GIS big data management, region based image analysis methodologies (object based) was adopted to deal with this problem. Such as, initiatives on the use of domain knowledge for classifying urban objects. This approach generated new challenges when attempting to formalise and exploit knowledge such as the difficulty of building knowledge-based systems due to the implicit nature of expert knowledge added to the challenge.

Another area that will benefit from big data research progress is the issue of scalability and the ability of exploiting big distributed data sets of images that do not fit into memory. The need to rethink current data analysis algorithms is evident and will impose more pressure on data analysts as graphical data sets grows exponentially. This will also affect the ability to deal with data imprecision, evaluate and correct errors in graphical raw data or segmentation data. This gives rise to the need to define sets of robust algorithms capable of incorporating data errors and imprecisions by defining the appropriate methodology to evaluate and correct errors or imprecisions in data and therefore on knowledge. This methodology will need to show significant success in combining all the information available on studied areas regardless of their media to enhance that data analysis process and therefore reflect positively on knowledge generation and management.

In summary the challenges facing GIS big data management is in the design and development of data analysis platforms adopting multi-level analysis to use all available data sources and methods to develop interdisciplinary data analysis methodologies integrating and merging data and knowledge from different domains such as geology, geophysics, environment sciences, and data mining.

2.2 Workflows

Traditional workflows depend on data transfer by streaming the data to some analyzing unit. Sensors record some entities in the first step and stream them for further modelling or analytics. Present research topics address edge-based computing as a modern way by pre-analyzing and filtering relevant data by sensor-networks to reduce data transfer. A second step is the modeling of the geographic object in form of a 3d representation in combination with thematic attributes. Further monitoring and modeling leads to moving and morphing multi dimensional representations of the geographic object of interest. To run analytics on the generated data as the third step, data usually needs to be transferred to some GIS or analyzing tool. Obviously, data transfer is a bottleneck when dealing with big data. Processing analytics also overloads the hardware capacities of the scientists' laboratories. In case of simulations e.g. finite element method, scientists usually write their own simulation applications or use specialized software products. Simulation results are written to some persistent memory to copy from for further analytics and visualization. Even only for the visualization of the simulation results the data transfer between main memory and the graphics cards memory turns out to be a bottleneck when dealing with large simulation results. In-situ solutions help to solve that problem by visualizing and analyzing the data side by side to the running simulation in real-time, concurrent, also on high performance computing clusters (Rivi et al., 2012). Scientists are able to see results immediately while running the simulation. Time consuming simulations do not need to run until the end if real-time results show strange behaviors.

Nevertheless, modern workflows use DBMS's (database-management-systems) to store the representations and integrate some analytic intelligence into the DBMS. Some of the analyzing work and even modeling different representations can be done faster by the "intelligent" geo-spatio-temporal DBMS itself, running centralized on more powerful servers than the workstations of the scientists. Furthermore, DBMS's are built to select, query and analyze large amounts of data efficiently where else file based solutions tend to fail. Multi-user support and privacy controls are further advantages of DBMS's. But parallel multi-dimensional DBMS's for geographical use are still research topics nowadays and therefore not commonly used. Achieving global scientific modeling and analytics on big geo-spatio-temporal data leads to high performance computing and its distributed/parallel data

processing power on clusters with new ways of data management such as NoSQL-DB (Not-Only-Standard-Query-Language-Database) support. Virtualization, cloud-based software products and services will be established to outsource the hardware into data centers or data warehouses which are built to archive big data and solve analytics centralized in a parallel multi-user environment. Programming paradigms such as vectorization, parallel algorithms, data streaming algorithms (Li et al., 2016), calculations on GPGPUs (General-Purpose-Computation-on-Graphics-Processing-Unit), the map-reduce programming model or test- and behavior-driven-development show new ways of developing efficient analytic software products to the environmental, economical or any other science community with different skills in computer sciences or programming to solve their tasks. To bring all of those modern technologies, programming paradigms and scientists together new workflows have to be developed to ease the way of cooperative working. From the geo-informatic point of view this is one of the challenges to support the efficient production of added value to geo-spatio-temporal data.

2.3 Online processing paradigms

In the following we refer to two main online processing paradigms. The first focuses on the efficient transaction processing known as OLTP (online transaction processing) and the second on analyzing the data known as OLAP (online analytic processing). Both, OLTP and OLAP are of great importance for geo-spatio-temporal DMS (data-management-systems). While modern OLTP-applications may focus on real time processing of sensor data in sensor-networks, OLAP-applications focus on complex analytics of structured data usually done in data warehouses. Both online processing paradigms hinder each other when working on the same datastock. Data warehousing can be differentiated by SOLAP (spatial OLAP, GIS interaction with OLAP), TOLAP (temporal OLAP, evolution of dimension instances available through the definition of temporal dimensions in OLAP), S-TOLAP (spatial TOLAP, GIS interaction with TOLAP), ST-OLAP (spatio-temporal OLAP, OLAP capabilities on spatio-temporal data-structures) and finally ST-TOLAP (spatio-temporal TOLAP, TOLAP capabilities combined with spatio-temporal data-structures) (Vaisman and Zimanyi, 2009). This paper focuses on ST-TOLAP, the most general form where geographic objects move and morph over time and carry some thematic attributes as geo-spatio-temporal data sets combined with spatial free dimensions of OLAP which evolve over time by integration of temporal dimensions (TOLAP).

A point of interest is the hardware setting to use for ST-TOLAP. Distributed DBS's (database-systems) suit best for distributed allies being part of the workflow while parallel DBS's are made for massive OLTP where several DBS servers host a copy of the same DBS to provide massive multi-user usage or for OLAP to solve one complex analytic query in parallel on one large data set distributed on a cluster as mentoined in common computer science literature. A Cluster usually is made of high speed connected racks consisting of high speed connected blades which are basic computers with a HDD (Hard Disk Drive) or a SSD(Solid State Drive), RAM (Random Access Memory) and a CPU (Central Processing Unit) in case of a Shared-Nothing-System. In case of a Shared-Disk-System the blades share a number of HDDs or SSDs. High efficiency is expected by Shared-Nothing-Systems if the data is well distributed such that every blade has average work load to do for nearly all transaction/processe-types. Shared-Disk-Systems are not as dependent on data distribution as Shared-Nothing-Systems but synchronization efforts could slow down the system. In-Memory grids use only RAM as main storage to reduce read and write operation times.

2.4 NOSQL systems

NoSQL Systems are DMS's (data-management-systems) that seem to be well suited to handle Big Data. Four groups are to mention. These are Key-Value-Stores, Extended-Record-Stores, Document-Stores and Graph-DBs. Each system differs in data-quality and -quantity. While Key-Value-Stores manage huge quantities of data, data itself has less structure. On the other hand Graph-DBs manage less data quantities but have more structure. The map-reduce programming model (e.g. hadoop) is a suitable mechanism to solve some analytics on big data. It is from great interest how this model may be used for ST-TOLAP purposes. Further investigations need to be done concerning the speedup and scaleup of parallel processing in ST-TOLAP and the physical or virtual data integration. Some operations on specific geo-spatio-temporal data distributions might result in too large intra-communication or synchronization overheads. Therefore, how the data is being distributed, the load-balancing and sharding, has impact on the processing time of geo-spatio-temporal queries. For example, if algorithms dependent on the local neighbourhood of spatial objects the local neighbourhoods need to be accessible within one node/blade as far as possible to reduce intra-communication (e.g. simulations, interpolations etc.). As a second example, if algorithms depend on distributed local neighborhoods of spatial objects the local neighbourhoods should be distributed across the cluster such that every node/blade has nearly the same work load to query or calculate all pieces (distributed R-Tree for searching in parallel or editing etc.). In both cases reading data from other nodes will turn into a massive communication between the blades for certain tasks within a Shared-Nothing-System or synchronization efforts within a Shared-Disk-System. A good topological structure of the geo-spatio-temporal data will help the controlled distribution.

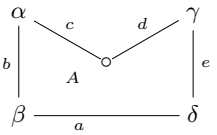
3. FIRST STEP: THEORETICAL MODEL FOR TOPOLOGICAL CONSISTENCY

In the literature there are various differing definitions of the notion "topological consistency", cf. e.g. (Li, 2006, Kang and Li, 2005, Rodrguez et al., 2010, Dušan and Branislav, 2004). Probably the one closest to our point of view is found in (Dušan and Branislav, 2004), in which topological consistency usually refers to the lack of topological errors, like unclosed polygons or dangling nodes. Instead, we will define it as the equality of the topological model with the topology derived from geometry in a certain way. The idea is that a model is consistent, if and only if it is properly embedded into Euclidean space. The main advantage of this new definition is that in this case, costly geometric computations can be avoided in topological queries by using the topological model only. Our notion of topological consistency then guarantees the correctness of topological query results.

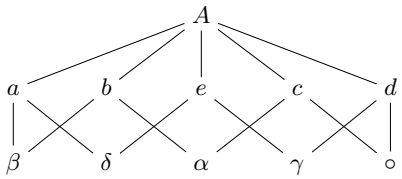
Let P be an n-dimensional polytope. We associate to P the following finite topological space $X(P)$ which we call *cell space* of P: the points of $X(P)$ are the interiors of all k-faces of P for $k = 0, \ldots, n$. The topology on $X(P)$ is the one generated by the *bounded-by* relation $>$ on the open faces: $a > b$ (or $b < a$) if b is in the boundary of a. For example, if P is a polygon (we call

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-4-W5-65-2017 | © Authors 2017. CC BY 4.0 License.

it $\mathcal{P}_1$):

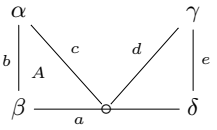


then $X(P) = X(\mathcal{P}_1)$ can be depicted as:



In the literature, the topology generated by the *bounded-by* relation is often called *incidence graph*. It is a so-called *finite* T_0*-space* or *poset*. The T_0 indicates that the relation $>$ yields an acyclic graph structure on the points of $X(P)$. An introduction to finite topological spaces can be found in (Barmak, 2011).

A geometrical realisation of a polytope can be obtained e.g. by assigning coordinates to the vertices of a boundary-representation model. However, if not enough care is taken, then one can obtain something like this (we call it $\mathcal{P}_2$):

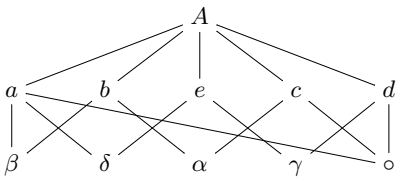


Here, there is a topological inconsistency: in the *bounded-by* topology, $\circ$ is in the boundary of the slanted edges only. However, in this geometric realisation, $\circ$ is in the boundary also of the punctured horizontal edge. Another problem is that in this geometric realisation the interior is disconnected. This is not what we usually think of as a polytope (or here: polygon). In any case, we can construct another finite topological space $\bar{X}(P)$ which extends $X(P)$ as follows: the points are all non-empty intersections $a \cap b$ for $a, b \in X(P)$. The relation $\prec$ is defined as follows: Let $a, b \in X(P)$, then $a \prec b$ if $a < b$, and for $i = a \cap b \neq \emptyset$: $i \prec a$ if $i \neq a$. The space $\bar{X}(P)$ is called the *overlay space* of P. We say that P is *topologically consistent* if the overlay space $\bar{X}(P)$ coincides with the space $X(P)$. In the above inconsistent example, we have that $\circ \prec a$ in addition to $<$ in the overlay space. In any case, the overlay space $\bar{X}(P)$ is also a finite T_0-space.

This notion of topological consistency was introduced in (Bradley, 2015) in a slightly different formulation in the context of configurations of polygons. Here, we can also consider configurations of objects of the following kind: First of all, we can define $X = X(P_1 \cup \cdots \cup P_\ell)$ and the overlay space $\bar{X} = \bar{X}(P_1 \cup \cdots \cup P_\ell)$ for polytopes $P_1, \ldots P_\ell$ in the same way as for a single polytope. Again we call $P_1 \cup \cdots \cup P_\ell$ *topologically consistent* if $X = \bar{X}$. Then, an n*-primitive* $\mathcal{P}$ is the interior of a polytope P of dimension n minus the union H of finitely many polytopes of dimension $\leq n$, provided that this union is topologically consistent. A *primitive* is defined as an n-primitive for some n. We call the closure of $P \setminus H$ the *closure* $\mathrm{cl}\,\mathcal{P}$ of $\mathcal{P}$, and write $X(\mathrm{cl}\,\mathcal{P})$ for the set of interiors of the faces of $P \setminus H$ including $\mathcal{P}$. Again, the overlay space $\bar{X}(\mathrm{cl}\,\mathcal{P})$ can be defined, together with topological consistency.

The final step is to build up spaces from primitives: let $C = \mathrm{cl}(\mathcal{P}_1) \cup \cdots \cup \mathrm{cl}(\mathcal{P}_\ell)$ be such a space. Then again we extend the definition of the cell space to obtain $X(C)$, and the overlay space $\bar{X}(C)$. Notice that the elements of $X(C)$ are in general not open cells in the sense of topology, but can be viewed, like in the case of cell complexes, as building blocks for a space C.

In our example of the topologically inconsistent polygon $\mathcal{P}_2$, $\bar{X}(\mathcal{P}_2)$ has the same points as $X(\mathcal{P}_2)$. The difference is in the topology: $\bar{X}(\mathcal{P}_2)$ can be depicted as



If the goal is to decide whether a geometric realisation of a space is topologically consistent or not, then it is enough to find the difference between X and $\bar{X}$. However, if one wants to tell *by how much* the geometric realisation is inconsistent, one can compute the *Betti numbers* of the skeleta of X and $\bar{X}$ and compare them. The Betti numbers b_i of a simplicial complex can be intuitively interpreted as the number of i-dimensional holes, where b_0 is the number of connected components, b_1 the number of loops, b_2 the number of voids etc. For a finite poset X, there also exist Betti numbers by associating to X the *order complex* $K(X)$, a simplicial complex whose k-simplices are the chains $a_0 < \cdots < a_k$ of length k. Then $b_i(X)$ is defined as $b_i(K(X))$.

The *dimension* of a poset X is the length of the longest chain in X. In (Bradley and Paul, 2013), this is seen as a form of *Krull dimension*. If the dimension of X is n, then the $n-1$-skeleton X_{n-1} is obtained from X by removing all points which are at the top of a chain of length n. Iterating this process yields the skeleta X_k with $k = 0, \ldots, n$, where $X_n = X$. We now define the numbers $b_i^k(X) = b_i(X_k)$ as the Betti numbers of the skeleta of X. If $X = X(C)$ for a space as above, then we finally can define the *topological defect numbers* as

$$\mu_{k,i}(C) = b_i^k(X(C)) - b_i^k(\bar{X}(C))$$

In our example, we have

$$\mu_{k,0}(\mathcal{P}_2) = 0, \quad k = 0, 1, 2 \tag{1}$$

$$\mu_{1,1}(\mathcal{P}) = 1, \quad \mu_{2,1}(\mathcal{P}_2) = 0 \tag{2}$$

which can be seen as follows: $X_2(\mathcal{P})$, $X_1(\mathcal{P}_2)$, $\bar{X}_2(\mathcal{P}_2)$ and $\bar{X}_1(\mathcal{P}_2)$ are all connected, and $X_0(\mathcal{P}_2) = \bar{X}_0(\mathcal{P}_2)$. This implies (1). In order to see the (2), observe that $X_1(\mathcal{P}_2)$ is the loop-graph with vertices $a, b, c, d, e, \alpha, \beta, \gamma, \delta, \circ$, which has $b_1(X_1(\mathcal{P}_2)) = 1$; and $\bar{X}_1(\mathcal{P}_2)$ has two loops with the same vertices, and its first Betti number is two. We have used the fact that for 1-dimensional posets, the Betti numbers coincide with the Betti numbers of the underlying graphs, the reason being that the order complex of a 1-dimensional poset is a graph. This explains the first part of (2). In higher dimension, things are not quite so simple, but if a poset has a unique maximal element, then all higher Betti numbers vanish (as such a space is contractible) (Barmak, 2011). This explains the second part of (2).

What remains for future work is to find efficient algorithms for computing Betti numbers of finite posets.

4. SECOND STEP: IMPLEMENTATION OF A GEO-SPATIO-TEMPORAL DATABASE ARCHITECTURE TO ENABLE BIG DATA ANALYSIS

Resuming the big data concepts and techniques from above and postulating a sound mathematical approach for a topologically consistent geo-spatio-temporal model, in a second step we introduce the theoretical and practical procedure of redesigning our geo-spatio-temporal database architecture called DB4GeO (Breunig et al., 2016) to implement parallel and distributed data-management concepts for a centralized workflow to reduce data transfer for ST-TOLAP in conjunction with simulation analytics. The main goal is to set up a hybrid data managment system which is able to provide a controlled data transfer from persistent storages to in-memory on RAM and provides a PlugIn-based service infrastructure which supports parallel algorithms to be plugged into the parallel database architectures where few different types of servers deal with the same data stock. Additional specialized NoSQL Servertypes running on the same cluster should be able to deal with other datatypes. In this way we address the issues of scalablity and use of distributed data-management raised in Section 2. If the analytic algorithms are parallelizable with acceptable synchronization overheads the analytic processing will be sped up and the general data transfer will be reduced because analytic processing will run on the same computer cluster, the same data stock and streams only the results of the algorithms to the clients. Archiving process-code versions and meta-data of the process-model for quering analytic work and reusing it with new data sets also leads to process-databases for historical scientific work. This has a lot of benefits for efficient research and collaboration even within one research center dealing with heterogenous data.

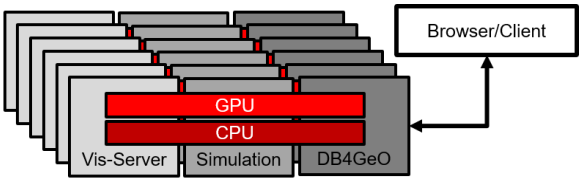


Figure 1. DB4GeO's approach to run parallel to the In-Situ framework of Paraview

In our present work we are connecting VTKs (Visualization Toolkit) (Schroeder et al., 2006) and ParaViews (Ahrens et al., 2005) In-Situ technology with DB4GeO and are exchanging the backend to a NoSQL-DB to support persistent big data storage. The connection to VTK and ParaView enables a lot of new features to DB4GeO not only in computational geometry and visualization for geo-spatio-temporal use. At the moment the DB4GeO architecture is based on db4o (database for objects) to handle the persistency of the stored geo-spatio-temporal objects and uses a RESTful (Representational State Transfer) Paradigm for web-communication (Breunig et al., 2016). But reading and writing to the physical hardware of VTKs geometry types and data sets is going to be managed, structured and distributed by DB4GeO in accordance with its own geo-spatio-temporal model and object model using a NoSQL-DB as backend to provide an efficient physical data integration in parallel DB environments. The operation-layer is going to be changed to provide integration of vtk/in-situ based source code, stored/archived, compiled and executable at run-time on a ParaView cluster. Results will be streamed to the clients or stored in DB4GeO for further processing. The source-codes for analytics shall be provided by modern programming paradigms such as test- or behavior-driven-development. In this way DB4GeO will run synchronized in parallel to the simulation and the visualization / analytics cluster to feed the two clusters from persistent shared nothing drives to the shared nothing In-Memory Grid and backwards (see Fig. 1). As theoretically prepared in Section 3 the first step focuses on the development of geo-spatio-temporal topological models to provide control of load-balancing and sharding. A suitable topological model is the key to have control over efficient data distribution for load-balancing and sharding in case of processes depending on topological constraints. DB4GeO is an object oriented geo-spatio-temporal DBMS prototype developed to handle moving and morphing volumes, surfaces, lines and pointclouds written in JAVA programming language (Breunig et al., 2016).

4.1 Object Model

Recently the object model has been redesigned to support ISO 19107 (Simple Feature Model) and ISO 19109 (General Feature Model) design patterns. Fig. 2 shows the class diagram of the newly implemented object/feature model and Fig. 3 shows the class diagram of the redesigned DB4GeO geometry model. Each DB object/feature (DB4GeO-DB4Object) within the object/feature model is now able to carry a spatial part (see Fig. 3), temporal part (DB4GeO-TemporalSequence, -TemporalInterval, -TemporalStamp) and/or a thematic part. Thematic classes are compiled at runtime and instances are referenced to the specific DB4GeO-DB4Object. A DB4GeO-DB4Object is a subclass of the abstract DB4GeO-Cell class which implements the basic general feature model (see Fig. 2). Each DB4GeO-Cell is part of a graph and carries its own thematic objects and tables of all child thematic objects where each table or object record belongs to one specific thematic class as schema definition.

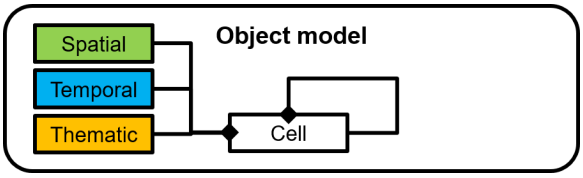


Figure 2. New object-model in DB4GeO (follows OGC's General Feature Model)

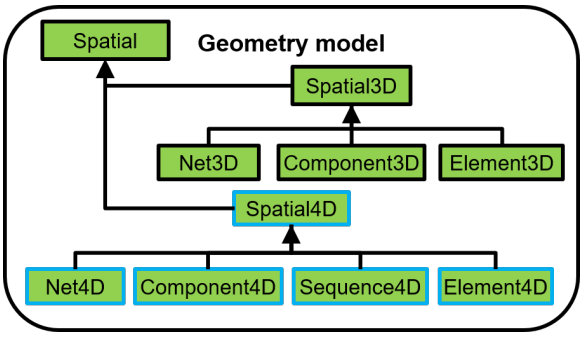


Figure 3. New class-hierarchy of geometry-package (allows cells to use all kinds of geometries and sets of geometry-package)

The DB4GeO-DB4Object graph can be arranged by needs e.g. level of detail (see Fig. 4), topological *incidence graphs*, modeling steps (from point cloud to simulation data set), CSG (constructive solid geometry) or data distribution etc. Specializations of the DB4GeO-Cell class provide functionalities to manage their child-instances and their spatial-, temporal- and thematic-part and the interconnection of those parts. This general api approach shall provide the ability to adapt to big data concepts as needed in a

parallel distributed DBMS's for ST-TOLAP in conjunction with simulation analytics.

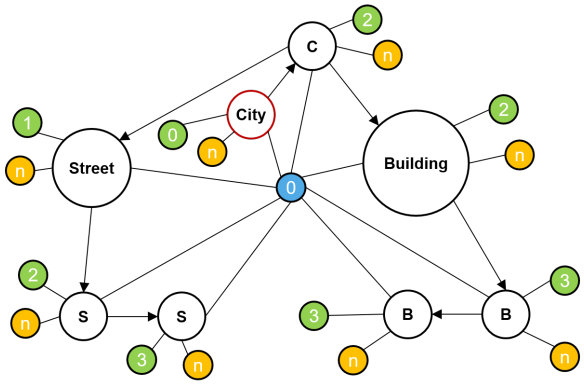


Figure 4. EXAMPLE: LOD-graph using new object-model;
green nodes: spatial parts with dimension;
blue node: time-stamp with dimension;
orange nodes: thematic parts with n dimensions;
white nodes: cells (red boundary: root node)

4.2 Geo-spatial simplices

The geometry model uses 0,...,3-dimensional simplices to support computational geometry tasks more efficiently. For a valid geo-spatial simplex (DB4GeO-Element3Ds) $c = \{p_0, \ldots, p_d\}$, $p \in \mathbb{R}^3$ (spatial space) of dimension $d := 0, 1, 2, 3$ at a certain time-stamp $t \in \mathbb{R}$ (temporal space) let $\bar{p}_i = p_i - p_0$ for $0 \leq i \leq d$. The constraint:

$$d = dim(span(\bar{p}_0, \ldots, \bar{p}_d)) \tag{3}$$

has to hold. The topological features are based on simplicial complexes with further constraints. As most GIS geometry cores the data types are classified by their dimension. Each 0,...,3-dimensional geo-spatial complex (DB4GeO-Component3D) is a well defined topological object. For a valid geo-spatial complex C within DB4GeO containing some simplices (or cells/polytops, in general) the following constraints have to hold:

C is topologically consistent (4)

All maximal cells are d-cells for $d \geq 0$ (5)

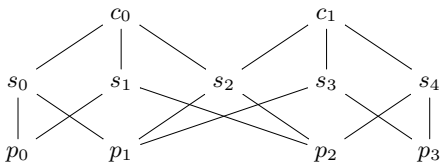
Betti numbers $b_0 = 1$ and $b_i = 0$ for $i > 0$ (6)

Every $(d-1)$-cell is boundary element of two d-cells only (7)

All d-cells are equally oriented (8)

We use here the notion of topological consistence as defined in Section 3. Constraint 5 states that all cells of a complex C within DB4GeO are d-dimensional. It has less-dimensional sub-cells but only for topological reasons as follows. Constraint 6 claims that there is only one connected component and there are no holes of any dimension within a complex C. Constraint 7 implies a One-To-One relation of neighbouring d-dimensional cells sharing one $(d-1)$-dimensional boundary cell (Breunig et al., 2016). This constraint ensures that a complex C is a d-dimensional manifold. And constraint 8 ensures that every complex C is an oriented manifold.

Geo-spatial complexes are collected in nets (DB4GeO-Net3Ds) with no topological constraints. However, those geo-spatial simplices and their sets exist at certain time-stamps only (so across $\mathbb{R}^3$ only) and because of that no temporal neighbors other than themselves are able to be identified. Navigating through geo-spatial complexes or even over neighboring geo-spatial complexes sharing boundary elements is partly done by the G-Maps paradigm (Breunig et al., 2016). The topological *incidence graph* for two neighboring geo-spatial triangle-simplices $c_0, c_1 \in C$ forming a valid geo-spatial complex at some time-stamp by the new object model looks like this:



It is to mention that the segments s_0 to s_5 are not referenced by the geo-spatial simplices of the geometry-model. Geo-spatial simplices in DB4GeO are defined on their points only. The edges or faces are able to be calculated at runtime if needed. But the cell-nodes could be instantiated to realize the topological *incidence graph*. How the spatial parts of those additional boundary cells are going to be integrated is a question of the applicational needs.

4.3 Geo-spatial simplices on spatio-temporal point tubes

With the help of spatio-temporal point tubes we are able to move and morph geo-spatial simplices over time (Breunig et al., 2016). Spatio-temporal point tubes are interpolation functions from $\mathbb{R}$ (temporal space) and a set of points in $\mathbb{R}^4$ (spatio-temporal space) to $\mathbb{R}^3$ (spatial space). They replace all $p_i \in \mathbb{R}^3$ (spatial space) from a geo-spatial simplex in the 3D model at time-stamp $t \in \mathbb{R}$ (temporal space) with a function f over t and a set of $p_{ij} \in \mathbb{R}^4$ (spatio-temporal space) for $0 \leq i \leq d$ and $j \in \mathbb{N}$. So each p_i becomes a function $f(t, p_{i0}, \ldots, p_{in})$ representing the i.th spatio-temporal point tube at a time-stamp t for n points $p_{ij} \in \mathbb{R}^4$. For a valid geo-spatial simplex and geo-spatial complex on spatio-temporal point tubes the above typical 3D model constraints have to hold, too. In this case for a valid geo-spatial simplex $c(t) = \{f(t, p_{00}, \ldots, p_{0n}), \ldots, f(t, p_{d0}, \ldots, p_{dn})\}$ of dimension $d := 0, 1, 2, 3$ in $\mathbb{R}^3$ with the explained definitions above let $\bar{f}(t, p_{i0}, \ldots, p_{in}) = f(t, p_{i0}, \ldots, p_{in}) - f(t, p_{00}, \ldots, p_{0n})$. The constraint:

$$d = dim(span(\bar{f}(t, p_{00}, \ldots, t, p_{0n}), \ldots, \bar{f}(t, p_{d0}, \ldots, t, p_{dn}))) \tag{9}$$

has to hold for each time slice at $t \in \mathbb{R}$. For a valid geo-spatial complex C within DB4GeO containing some cells of that kind the above constraints 4, 5, 6, 7 and 8 have to hold for each time slice at $t \in \mathbb{R}$, respectivly. But for each geo-spatial complex defined like that we are still missing true geo-spatio-temporal polytopes to setup some geo-spatio-temporal topology easily.

4.4 Geo-spatio-temporal polytopes

The geo-spatio-temporal model is based on polytope complexes which are also loosely collected in nets (DB4GeO-Net4Ds).

It follows the Polthier and Rumpf model (Breunig et al., 2016). A geo-spatio-temporal polytope complex (DB4GeO-Component4D) is a collection of geo-spatio-temporal polytope sequences (DB4GeO-Sequence4Ds) for the spatial space coordinates combined with one single temporal-sequence (DB4GeO-TemporalSequence) for the temporal space coordinates. All geo-spatio-temporal polytope sequences within a geo-spatio-temporal polytope complex share the same temporal-sequence to reduce memory costs. A temporal-sequence is a linearly sorted interconnected collection of temporal-intervals (DB4GeO-TemporalIntervals), where else each single geo-spatio-temporal polytope sequence describes the movement and deformation of one geo-spatial simplex over time by temporally linear sorting interconnected geo-spatio-temporal polytopes (DB4GeO-Element4Ds). Therefore, a single geo-spatio-temporal polytope within a geo-spatio-temporal polytope sequence could be seen as one change in movement and/or shape of one single geo-spatial simplex by referencing a geo-spatial pre-simplex for the first time-stamp (DB4GeO-TemporalStamp) $t_0 \in \mathbb{R}$ (temporal space) of some temporal-interval referenced by the temporal-sequence of the geo-spatio-temporal polytope complex it is being part of and a moved/morphed geo-spatial post-simplex at the second time-stamp $t_1 \in \mathbb{R}$ of the same temporal-interval. The next geo-spatio-temporal polytope within the geo-spatio-temporal polytope sequence shares the last geo-spatial post-simplex of the last geo-spatio-temporal polytope as geo-spatial pre-simplex and adds a new changed geo-spatial post-simplex to itself. Therefore, special kinds of spatio-temporal point tubes mentioned in the model defined above do exist implicitly. Anyway, all geo-spatial pre- or post-simplices belonging to the same time-stamp of each geo-spatio-temporal polytope sequence within a geo-spatio-temporal polytope complex form a geo-spatial complex as a geo-spatial topological constraint. For a valid geo-spatio-temporal polytope $c = \{\hat{p}_0, \ldots, \hat{p}_{2d-1}\}$ in $\mathbb{R}^4$ with dimension $d := 1, 2, 3, 4$, points $\hat{p}_i = \begin{pmatrix} p_i \\ t_{i\%2} \end{pmatrix} \in \mathbb{R}^4$ (spatio-temporal space), $p_i \in \mathbb{R}^3$ (spatial space), two time-stamps $t_0, t_1 \in \mathbb{R}$ (temporal space) with $t_0 \neq t_1$, $0 \leq i \leq 2d-1$ and "%" as modulo operator let $\bar{p}_i = \hat{p}_i - \hat{p}_0$. The constraint:

$$d = dim(span(\bar{p}_0, \ldots, \bar{p}_{2d-1})) \tag{10}$$

has to hold where the points $\hat{p}_i$ containing the temporal coordinate t_0 belong to the geo-spatial pre-simplex and the points containing t_1 belong to the geo-spatial post-simplex. For a valid geo-spatio-tempoal complex C within DB4GeO containing some cells of that kind the above constraints 4, 5, 6, 7 and 8 have to hold, the same way as the geo-spatial complex constraints of the 3D model but in $\mathbb{R}^4$. Geo-spatio-temporal polytopes of that kind and their collections/sets may exist over spatial intervals (not in case of points) but have to have a temporal expension. They are able to have temporal neighbors at the time-stamp of their temporal boundary and we are able to set up a geo-spatio-temporal topological *incidence graph* by the new object model. As an example for a valid geo-spatio-temporal segment-polytope complex C (see Fig. 5) with two neighboring geo-spatio-temporal segment-polytopes $c_0, c_1 \in C$

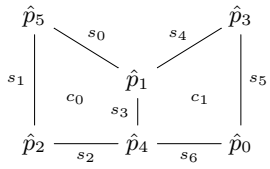
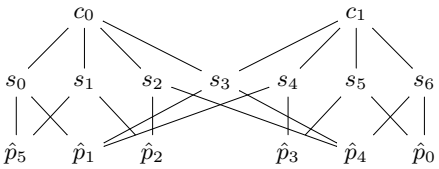


Figure 5. EXAMPLE: Valid geo-spatio-temporal segment-polytope complex C for two neighboring geo-spatio-temporal segment-polytopes $c_0, c_1 \in C$ with $s_2, s_0 \in c_0$ and $s_6, s_4 \in c_1$ and s_2, s_6 as geo-spatial pre-segment-simplices and s_0, s_4 as geo-spatial post-segment-simplices and s_1, s_3, s_5 as implicit linear spatio-temporal point tubes

the geo-spatio-temporal topological *incidence graph* will be:



There are a couple of different ways of dealing with geo-spatio-temporal changes technically. Every case has benefits for keeping track of geo-spatio-temporal topological consistency. If a temporal change leads always to some changes in position and/or shape of some geo-spatial simplex by definition the geo-spatio-temporal polytope complex needs to be split if at least one geo-spatial simplex of the geo-spatio-temporal polytope complex does not change over time because it contradicts with the definition. Splitting this kind of polytope complex leads to partial redundant temporal-sequences or redundant referencing of equal temporal-sequence parts. A second definition could be that no spatial changes are allowed by ongoing time but this would lead to redundant referencing of the same simplex (or parts of it) within the polytope as pre- and post-simplices (or pre- and post-parts of it, respectively). Investigations to invent models which are geo-spatio-temporal topologically consistent while keeping minimum redundancy and redundant referencing is part of our ongoing research.

4.5 Geo-spatio-temporal polytopes on spatio-temporal point tubes

With the help of spatio-temporal point tubes we are able to optimize the geo-spatio-temporal polytope complexes to a more dynamic and temporal scaleable data set and to keep the benefits of a true geo-spatio-temporal topology. Furthermore, we are able to simplify the class-hierachy by removing the polytope sequence class used to organize for moving and morphing of a single geo-spatial simplex. In this case, a geo-spatio-temporal polytope complex may be a collection of polytopes only and querying a polytope sequence is a geo-spatio-temporal topological algorithm and not a return of a specialiazed geo-spatio-temporal polytope sequence instance/object within a geo-spatio-temporal polytope complex. Anyway, the definition follows the geo-spatial simplex model on spatio-temporal point tubes. We evaluate the spatio-temporal point tubes at special time-stamps $t \in \mathbb{R}$ (temporal space) but not only for one simplex at time-stamp t for the materialization of a simplex at the specific time-stamp but with

evaluating two time-stamps $t_1, t_2 \in \mathbb{R}$ with $t_1 \neq t_2$ and use the returned spatial coordinates for the pre-, respectivly post-simplex of the geo-spatio-temporal polytope.

5. CONCLUSION AND OUTLOOK

In this paper we introduced the theory of a geo-spatio-temporal topological model to support the topological consistency check of geo-spatio-temporal polytopes in particular and nd-objects in general. Challenges and techniques for the handling of big geo-spatial data and data distribution have been discussed. With the help of DB4GeO's geo-spatio-temporal and object model, topological *incidence graphs* are able to be stored in the database. By the use of big data property graph databases as backend of DB4GeO, these *incidence graphs* are going to be used for the controlled distribution of big geo-spatio-temporal data across cluster nodes and for topological analysis by the functionality of the used backend database. To execute topological consistency analytics, those topological models may be extended while not necessarily touching the geometries themselves and to reorganize the whole geometric structure.

In our future research we will continue on the examination of the introduced topology model focusing on dynamic objects. It will include the development of a suitable service infrastructure for efficient parallel processing of geo-analytics and -simulations in a parallel and distributed system environment. Furthermore, we strive for a centralized workflow for the storage and processing of big geo-spatio-temporal data. Efficient calculations of Betti numbers will help to topologically analyse geo-spatio-temporal data sets. As applications we plan to have a look on near-to real-time applications and on big geo-spatial data applications of Dubai City in the United Arab Emirates.

ACKNOWLEDGEMENTS

This work is partially funded by the Deutsche Forschungsgemeinschaft (DFG) under grant BR 3513/12-1 and BR 2128/18-1. The anonymous referees are thanked for valuable suggestions.

REFERENCES

Ahrens, James, Geveci, Berk, Law and Charles, 2005. *ParaView: An End-User Tool for Large Data Visualization, Visualization Handbook*. Elsevier.

Assuncao, M., Calherios, R., Bianchi, S. and Netto, M., 2015. Big data computing clouds: Trends and future directions. *Journal of Parallel and Distributed Computing* 79–80, pp. 3–15.

Barmak, J., 2011. *Algebraic Topology of Finite Topological Spaces and Applications*. LNM 2032, Springer.

Bradley, P., 2015. Supporting data analytics for smart cities: An overview of data models and topology. In: A. Gammerman, V. Vovk and H. Papadopoulos (eds), *Statistical Learning and Data Sciences SLDS 2015*, LNCS 9047, Springer, pp. 406–413.

Bradley, P. and Paul, N., 2010. Using the relational model to capture topological information of spaces. *The Computer Journal* (1), pp. 69–89.

Bradley, P. and Paul, N., 2013. Dimension of Alexandrov topologies. arXiv:1305.1815v1 [math.GN].

Breunig, M., Kuper, P., Butwilowski, E., Thomsen, A., Jahn, M., Dittrich, A., Al-Doori, M., Golovko, D. and Menninghaus, M., 2016. The story of db4geo - a service-based geo-database architecture to support multi-dimensional data analysis and visualization. *ISPRS Journal of Photogrammetry and Remote Sensing* 117, pp. 187–205.

Dušan, J. and Branislav, B., 2004. Elements of spatial data quality as information technology support for sustainable development planning. *Spatium* 11, pp. 88–83.

Goodchild, M., 2016. GIS in the era of big data. *Cybergeo : European Journal of Geography [online]*.

Kang, H. and Li, K., 2005. Assessing topological consistency for collapse operation in generalization of spatial databases. In: J. Akoka (ed.), *Perspectives in Conceptual Modeling. ER 2005*, Lecture Notes in Computer Science, Vol. 3770, Springer, Berlin, Heidelberg.

Li, S., 2006. On topological consistency and realization. *Constraints* 11(1), pp. 3151.

Li, S., Dragicevic, S., Castro, F., Sester, M., Winter, S., Coltekin, A. and Pettit, C., 2016. Geospatial big data handling theory and methods: A review and research challenges. *ISPRS Journal of Photogrammetry and Remote Sensing* 115, pp. 119–133.

Ma, Y., Wu, H., Wang, L., Huang, B., Ranjin, R. and Zomaya, A., 2014. Remote sensing big data computing: Challenges and opportunities. *Future Generation Computer Systems* 51, pp. 47–60.

Oosterom, P. v., Maessen, B. and Quak, W., 2002. Generic query tool for spatio-temporal data. *Int. J. Geographical Information Science* 16(8), pp. 713–748.

Rivi, M., Calori, L., Muscianisi, G. and Slavnic, V., 2012. In-situ visualization: State-of-the-art and some use cases. Partnership for Advanced Computing in Europe White Papers.

Rodrguez, M., Brisaboa, N., Meza, J. and Luaces, M., 2010. Measuring consistency with respect to topological dependency constraints. In: *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 182–191.

Schroeder, W., Martin, K. and Lorensen, B., 2006. *The Visualization Toolkit (4th ed.)*. Kitware.

Vaisman, A. and Zimanyi, E., 2009. What is spatio-temporal data warehousing? *International Conference of Data Warehousing and Knowledge Discovery* pp. 9–23.