Towards Integrated Data Management and Analysis for Spatio-Temporal Digital Terrain Models

Ruiqi Liu[1], Paul Kuper[1*], Thomas Grombein[1], Christoph Naab[1], Mulhim Al-Doori[2], Martin Breunig[1]

[1] Geodetic Institute, Karlsruhe Institute of Technology, Germany – (ruiqi.liu, paul.kuper, grombein, naab, martin.breunig)@kit.edu
[2] University of Science and Technology of Fujairah, United Arab Emirates – m.aldoori@ustf.ac.ae

Keywords: Geospatial Data Modelling, Geospatial Data Management, Spatio-Temporal Data Management, GeoDBMS, Big Geospatial Data Analysis, Digital Terrain Models.

Abstract

Digital Terrain Models (DTMs) are still one of the most emerging products derived from remote sensing technologies such as LiDAR. A DTM usually is defined as the current digital representation of the earth´s surface. However, due to natural events and human influences the earth's surface is constantly changing. Therefore, the ability to analyse DTM data over time allows to understand and quantify changes, which are crucial for environmental monitoring, disaster management and urban planning. In this paper we introduce a GeoDBMS-based approach to manage, sustainably provide, analyse and visualize spatio-temporal DTM data. First, we review current approaches for the management and temporal modelling of DTM data. Then we present an event-based time-stamping spatio-temporal data model that meets special requirements for the management and analysis of DTMs to monitor spatial changes within the DTM in time. Based on this model, we present methods for the management and analysis of spatio-temporal DTMs. Thereby the model is extended by an appropriate server-infrastructure and a graphical user interface that enables to query, analyze, visualize and export time series of DTMs. Data consistency and accuracy for DTMs are also considered. A use case is conducted based on spatio-temporal DTM datasets of South-West Germany, demonstrating the suitability of the approach. Furthermore, geodetic applications and new DTM research directions are shown. Finally, conclusions are drawn from the paper and an outlook is presented describing our future research based on new use cases, including the analysis of DTMs and Digital Surface Models (DSMs) in the United Arab Emirates.

1. Introduction

As well known, "the Digital Terrain Model (DTM) is a statistical representation of the continuous surface of the ground by a large number of selected discrete points with known xyz coordinates in an arbitrary data coordinate field" (Miller, 1958). Light Detection and Ranging (LiDAR) is a key technology to acquire DTMs. Therefore, large and small scale DTMs are used in many spatial and spatio-temporal geo-disciplines such as urban planning, environmental monitoring, and disaster management.

In practice, the discrete point clouds acquired by laser scanning campaigns, are interpolated and transferred into continuous areal objects of a DTM resulting in Triangulated Irregular Networks (TINs) and in raster-based grid representations, respectively. The same is true for Digital Surface Models (DSM) containing vegetation and buildings. For repeated "just on demand" usage of these vector and raster geometries, their standardized storage and retrieval in a Geodatabase Management System (GeoDBMS) is indispensable. Such a system should optimally provide sufficient support for the management and analysis of spatio-temporal DTMs in vector and raster representation.

In this paper, we focus on the development of a GeoDBMS and associated tools for the management and analysis of DTMs and spatially selected parts of DTMs including their time series. We evaluate our approach with raster datasets produced by the State Office for Geoinformation and Land Development (LGL) in Baden-Württemberg, Germany.

Some authors (e.g. Tseng et al., 2013; Kamp et al., 2023) call time series of DTMs also "Multi-temporal Digital Terrain Models". We are using the term "Spatio-Temporal Digital Terrain Models" instead of multi-temporal DTM to focus on spatial changes of the DTM in time. Spatio-temporal DTMs are essential to examine geomorphic phenomena such as landslides or all kinds of environmental monitoring.

The paper is structured as follows: In section 2, we refer to related work followed by section 3, introducing an event-based time-stamping data model being adapted to the modelling and management of spatio-temporal DTM data. In section 4, our approach of a spatio-temporal data management and analysis for spatio-temporal DTMs is presented. Section 5 describes the implementation of the approach and in section 6 a practical field test with data from South-West Germany is treated. Finally, section 7 presents DTM research needs for emerging geodetic applications and in section 8 conclusions are drawn from the experiences of our approach gained so far. Last, but not least, an outlook is given on our future DTM research with DTM and DSM data from Dubai and the United Arab Emirates.

2. Related Work

The significance of Digital Terrain Models for remote sensing and geographical studies has been proven by early research work of (Miller, 1958; Doyle, 1978; Weibel and Heller, 1991; Zhilin et al., 2004). Characteristic applications of spatio-temporal LiDAR-derived digital terrain models to estimate landslide volumes were published by (Tseng et al., 2013). (Kamp et al., 2023) studied error sources and uncertainties of spatio-temporal DTMs derived from LiDAR platforms being used for geomorphological studies.

Early research about array-based raster databases has been published by (Baumann et al., 1997). Later, services for the access of raster data and operations have been added (Baumann, 2010; Hu et al., 2018). Furthermore, (Baumann et al., 2018) have shown that data cubes are a suitable concept to provide raster data interfaces for spatial and spatio-temporal data analysis. In data cubes the code is "shipped to the data". Therefore, the communication costs arising when sending data from the server to the users, are minimized. Vector representations such as TINs can be efficiently stored by object-relational DBMS/GIS such as the open software products PostgreSQL/PostGIS using one of the widely spread spatial access methods (e.g. Quadtree or R-Tree).

For DTM data analysis ready-to-use open source geometric and topological libraries such as the Geospatial Data Abstraction Library (GDAL, 2024) have been developed, which also allow the transformation from raster to vector representations and vice versa. A prototype 3D/4D GeoDBMS, developed to support the analysis of landslides, was developed by (Breunig et al., 2010). In (Breunig et al. 2020), it has been stated that database support for big geospatial data analysis will be one of the big future research directions in GIScience.

One of the earliest authors to mention temporal GIS and temporal data models were (Langran and Chrisman, 1988). Over the years, the type of spatio-temporal data models in GIS changed from layer-based temporal snapshot models via object-oriented changes models to event-based models where the event (not the object) is in the centre of interest. Regarding the geometric and topological changes of TINs (Polthier and Rumpf, 1994), the events are changes of the geometry and the topology of a triangulated irregular network. Whereas during a certain timestamp only the topology (i.e. the discretization of the TIN) is allowed to be changed and the geometry is constant, between two timestamps only the geometry (i.e. the coordinates of the TIN) is allowed to be changed and the topology stays constant. With this model, dynamic changes in TINs can be monitored on the level of the triangles rather than on the point cloud. The following presented model takes a broader approach focusing on raster data. The efficient management of DTM raster datasets requires methods to query spatially and temporally selected regions, different resolutions, and events.

3. An Event-Based Time-Stamping Data Model for Digital Terrain Models

3.1 Requirements and Challenges

On the space scale, spatio-temporal DTM datasets are usually generated by remote sensing technologies such as LiDAR. They can cover a large area representing the corresponding elevations at a specific timestamp. On the time scale, multiple DTM datasets of the same region are collected with their discrete timestamps, in order to reflect dynamic elevation changes during a period of time. By providing insights into historical elevation changes of large-scale and small-scale areas, spatio-temporal DTM data are very useful in a wide range of fields, such as urban planning, environmental monitoring, and disaster management.

The requirements for managing and analysing spatio-temporal DTMs include the ability to query big data sets based on spatial, temporal and spatio-temporal properties and behaviour. It should be easily possible to manage and analyse different DTMs of the same timestamp as well as the same DTM at different timestamps. The latter includes the analysis of the changes of the DTM during different timestamps to monitor the continuous changes of the geometry and the topology of a DTM. Note that the comparison of different DTMs in time is not a trivial operation. Therefore, the GeoDBMS must provide a system of modules for the spatio-temporal analysis of DTMs. Spatio-temporal database queries have to be applied to a single or to several DTMs at one or several timestamps. As the GeoDBMS then supports advanced spatio-temporal data analysis, we speak of an integrated approach for the data management and data analysis of DTMs. In a properly implemented system, it should also be possible to export the DTM datasets in various data formats and resolutions.

3.2 Model Overview

The data model introduced in the following has its roots in snapshot-based (Armstrong, 1988) and event-based (Peuquet and Duan, 1995) approaches. For managing DTM datasets, the snapshot model has the advantage of showing the current elevation at multiple specific time points from a holistic scale. However, its weakness is that it lacks the explicit representation of dynamic elevation changes over time. The event-based model can capture dynamic elevation changes between different event states, but compared to the snapshot model, it is more complex and cannot offer a view of the entire region due to its focus on individual events.

Our approach combines these two models being specifically adapted to the management of spatio-temporal DTM datasets. It integrates the advantages of both approaches. Therefore, we speak of an "Event-Based Time-Stamping Data Model" (Liu, 2024).

As the left part of Figure 1 shows, like the snapshot model, the Event-Based Time-Stamping Data Model can organize DTM datasets of the same region into multiple time-stamping layers T_1, T_2, …, and T_n with different resolutions. In this example, each time-stamping layer is composed of 12 DTM datasets, each belonging to a different event (corresponding to one timestamp). The colour differences of the shaded areas indicate the elevation changes in the corresponding region. In this way, the model can manage spatio-temporal DTM datasets and reflect global changes. To capture local changes within each sub-region more effectively, the right part demonstrating the event-based perspective, considers DTM datasets at different timestamps, e.g. at T_1 each DTM dataset in the corresponding sub-region can be represented respectively as the event E_1, E_2, ..., and E_{12}.

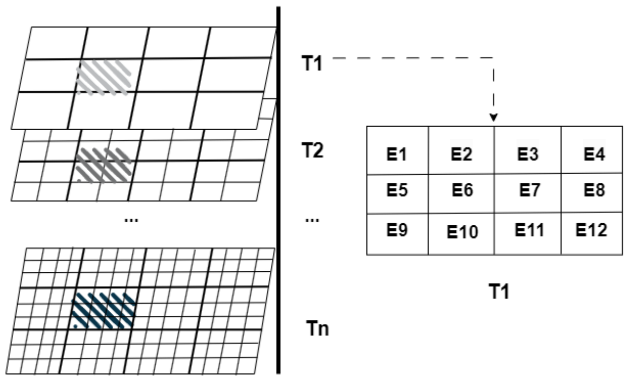


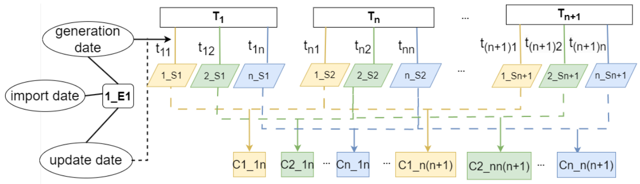
Figure 1. The Event-Based Time-Stamping Data Model.

Overall, the Event-Based Time-Stamping Data Model segments spatio-temporal DTM data into different layers based on timestamps. Each time-stamping layer is a composition of DTM data record events, which can reflect dynamic changes in DTM data between timestamps.

3.3 Model for Spatio-Temporal Analysis

Based on the Event-Based Time-Stamping Data Model, further spatio-temporal analysis can be conducted, as illustrated in Figure 2. The spatio-temporal analysis in this paper focuses on comparing DTM datasets of the same region with different timestamps, in order to reveal the elevation changes over time. Here, the comparison is based on the "status", which refers to the elevation value reflected in each region at a given timestamp. Whereas an event here is defined as a collection of actions such as generate, update etc., the status is the result of the event, i.e. it consists of the changed elevation values of the DTM. Assuming there are multiple regions with IDs <1, 2, …, and n>, at the same timestamp they have the same status, e.g., at timestamp T_1, the corresponding status is <1_S1, 2_S1, …, n_S1>.

The comparison between different timestamps of the same region can be quantified as C. As table in Figure 2 shows, taking the region with ID 1 as an example, the elevation changes between 1_S1 and 1_Sn are represented as <C1_1n>.



Time stamp	Region ID	Generation date	Event state	Elevation change
T_1	1	t_{11}	1_S1	
Tn	1	t_{n1}	1_Sn	C1_1n

Figure 2. Detection of elevation changes based on the Event-Based Time-Stamping Data Model.

By incorporating events into spatio-temporal analysis, dynamic changes between timestamps can be captured. An event here is defined as the collection of actions related to spatio-temporal DTM datasets, such as generation, import, and update, with dates being recorded. The *generation date* captures the time when the airborne scanner flies over the current area to generate the DTM data. The *import date* records the date when the DTM dataset is imported into the DBMS, which can provide traceability when there is a data loss. The *update date* indicates the date when the DTM datasets is updated due to data accuracy improvements, error corrections, or other factors. If there is an update occurrence, the original DTM dataset and the generation date will be replaced by the updated version.

4. Data Management and Analysis

To facilitate the integration of data management and analysis for spatio-temporal DTM datasets, a system architecture framework has been developed, which also cares for data consistency and accuracy.

4.1 DTM Data Management System Architecture

To meet the system requirements for the efficient spatio-temporal data management and analysis of DTMs we developed the following system architecture shown in Figure 3.
The system is designed to include not only a fundamental DBMS, but also a service infrastructure and a web client application. First, the DBMS manages and retrieves spatio-temporal DTM datasets. Internally the spatio-temporal DTM data are adapted to the Event-Based Time-Stamping Data Model. Thus the DBMS supports functions such as data management, spatio-temporal queries, spatio-temporal analysis, and export of DTM datasets. The data management refers to operations such as the import, update, and deletion of DTM datasets in the DBMS. For spatio-temporal retrieval, the DBMS should support to query the corresponding DTM datasets, by event attributes, shapes, and timestamps. In the spatio-temporal analysis module, the DBMS is able to compare DTM datasets of the same region having two different timestamps as well as to calculate the degree of elevation changes during the corresponding timestamps (e.g. the years of the laser scanning campaigns in which the data was acquired). The export module allows users to export DTM datasets including the changes of the resolution, format, or tiles.

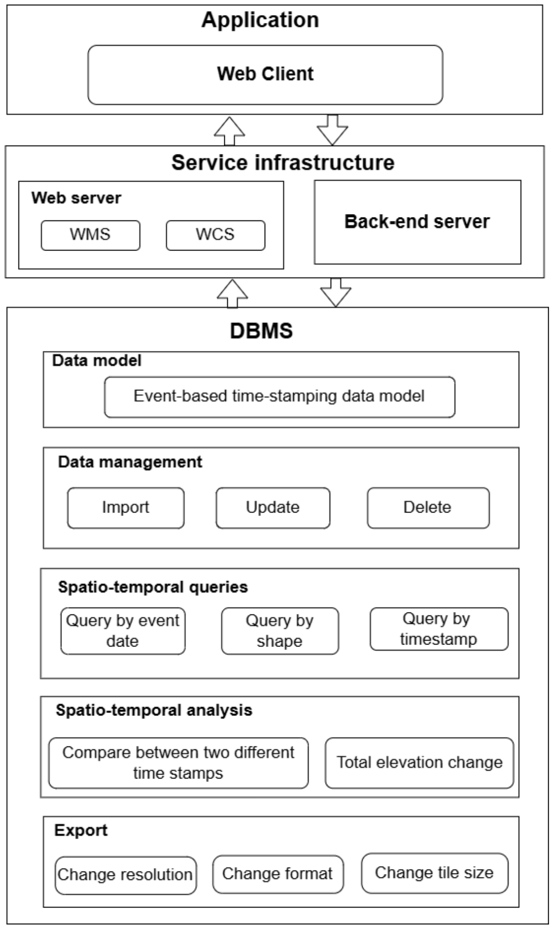


Figure 3. The DTM data management system architecture.

The service infrastructure supports the transfer of DTM datasets between the DBMS and the applications, by providing web services such as WMS and WCS through a web server. Besides, the DTM datasets is also expected to be transferred by connecting the DBMS and the front-end client through a back-end server. For a user-friendly access to the data and functionality, an advanced front-end client has been developed for users to interact with the DBMS.

4.2 Data Consistency and Accuracy

To guarantee data consistency as well as the same accuracy of the originally collected and the stored DTM datasets in the DBMS, the DTM data management architecture has to provide spatial integrity constraints to be checked on demand. An example of such an integrity constraint is the calculation of the Root Mean Square Error (RMSE) to determine if there is any difference of the same DTM dataset before and after having been imported into the DBMS. For public authorities dealing with DTMs, this is an extremely important integrity constraint. Thus, the experts are able to check at any time, if there have occurred spatial changes within the DTM, i.e. modifications of the *x*-, *y*- and *z*-coordinates, respectively, representing the location of the single raster cells. As well known, the RMSE between two Digital Terrain Models DTM_1 and DTM_2 can be determined as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(DTM1_i - DTM2_i)^2} \quad (1)$$

Formula (1). Root Mean Square Error between the elevation values of two DTMs used as integrity constraint for DTM datasets.

Whereas $DTM1_i$ represents the i^{th} pixel value of the original DTM acquired outside the DBMS, $DTM2_i$ is the corresponding pixel value of the same DTM stored in the DBMS, and N refers to the total number of pixels (raster cells) of the DTM. Thus, *DTM2* may reflect changes of the elevation values of *DTM1*. The RMSE is used as a Boolean consistency check (result is true or false) for imported DTMs (Liu, 2024).

By checking the Digital Terrain Model directly after importing the data into the DBMS, any internal changes concerning the representation of this DTM - we call it now *DTM2* - can be detected. In the case of PostgreSQL/PostGIS it has been shown that no internal changes of the DTM occur, i.e. the representations of *DTM1* and *DTM2* are equivalent (Liu, 2024).

For the calculation of the difference between the elevation values of the same region at two different timestamps, the following formula is used:

$$D(x, y) = DTM(x, y, t_m) - DTM(x, y, t_n) \qquad (2)$$

Formula (2). Difference between the elevation values in the same region at two different timestamps.

Whereas x and y represent the pixel position in the DTM dataset, t_m and t_n refer to the two timestamps. If $D(x,y)$ is positive, it indicates the elevation value decreases from t_m to t_n at the point *(x,y)*, whereas if $D(x,y)$ is negative, it means an increase during the period at the same position.

Furthermore, data accuracy can be monitored with the RMSE by comparing the decimal places of *DTM1* and *DTM2*.

5. Prototype Implementation

The prototype implementation of our approach is based on a PostgreSQL/ PostGIS geospatial database architecture as shown in Figure 4.

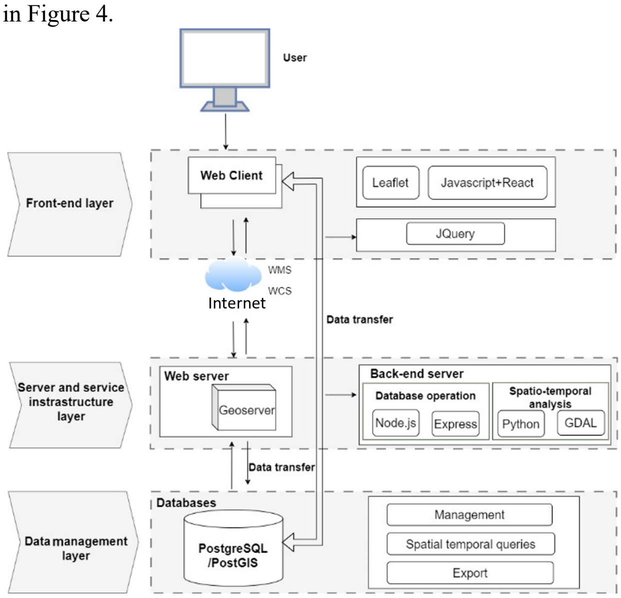


Figure 4. The prototype implementation architecture.

This architecture consists of the following three layers: The *data management layer* where the data are spatially and temporally stored and queried, the *server and service infrastructure layer* where the query results are provided and the data analysis is executed and finally the *front-end layer* where the communication between the users and the system takes place.

5.1 Data Management Layer

The *data management layer* of our prototype implementation uses PostgreSQL/PostGIS. In addition to PostGIS, also comparative studies with RasDaMan (Baumann et al. 1997; Baumann et al. 2018) were carried out (Liu, 2024).

Although this array-based DBMS showed a better performance in processing big raster data, PostgreSQL/PostGIS has advantages such as simple installation, extensive community support, and easy combination with vector data for spatial analysis. Therefore, PostgreSQL/PostGIS was chosen as the implementation basis of our prototype system.

The management of DTM datasets in PostgreSQL/PostGIS refers to operations such as data import, update, and deletion. PostGIS provides the package *Raster2pgsql* to import raster data in the same data format that (GDAL, 2024) supports. Therefore, the import of DTM datasets into multiple tables is realized according to the timestamp <T_1, T_2, …, and Tn>, as shown in Code 1.

Code 1. Import of DTM datasets into PostGIS.

```
raster2pgsql -s 25832 -I -M -C *.tif -F T1| psql -d database
-h localhost -U postgres -p 5432
```

The parameter *I* creates a spatial index on the raster column, which improves the query performance. With the parameter *M*, the raster table is "vacuum analysed". The vacuum command is to clean up dead tuples, which are rows that are no longer valid due to transactions such as deletions or updates. The parameter *C* applies constraints on the raster data. Constraints are rules that are applied on data columns, in order to register the raster data in PostGIS properly, such as *CHECK (ST_SRID(geom) = 25832)* is to restrict raster data meaning that only when the raster data's SRID is 25832, it can be registered.

Examples of a simple temporal and spatial database query, respectively, are shown in Code 2 and Code 3 examples below. Temporal queries aim to acquire DTM datasets from the time scale. As the example in Code 2 shows, DTM datasets are selected whose generation date is within the time interval.

Code 2. Query DTM datasets by time interval.

```
SELECT * FROM PUBLIC. T1 where generation_date
BETWEEN '2000-02-01'  AND  '2000-08-01'
```

Spatial queries aim to query DTM datasets from geographic space. This function enables users to specify regions of interest to extract the corresponding DTM datasets. Code 3 is an example of querying the DTM datasets selected by a polygon.

Code 3. Query DTM datasets by clipping the data within a polygon.

```
SELECT filename, ST_Union(ST_Clip(rast, geom))
FROM public.T1
CROSS     JOIN    ST_MakeEnvelope(457172.2,    5427640.5,
457445.7, 5427377.3, 25832)
AS geom
WHERE ST_Intersects(rast, geom)
GROUP BY filename;
```

Furthermore, our prototype system enables users to export DTM datasets with customized output settings, such as changing the resolution, tile size, and format.

Code 4. Export DTM datasets as a GeoTIFF file.

```
DROP TABLE IF EXISTS GeoTIFF;
CREATE TABLE GeoTIFF AS
SELECT lo_from_bytea(0, GeoTIFF_file) AS loid
FROM     (     SELECT     ST_AsGDALRaster(rast,'GTiff',
ARRAY['QUALITY=50'])
AS GeoTIFF_file
FROM public.DTM
);
SELECT lo_export(loid, 'GeoTIFF_1.tiff') FROM GeoTIFF;
```

Code 4 is an example of exporting DTM datasets. PostGIS provides the function *ST_AsGDALRaster()* and a series of large object(LO) operators to export DTM datasets in PostGIS as a GeoTIFF file.

5.2 Server and Service Infrastructure Layer

The *server infrastructure layer* is composed of two servers: the web server and the back-end server. The web server is realized by (Geoserver, 2024), which serves for sharing, processing, and publishing the DTMs. The back-end server provides the database connection, which is developed based on Node.js (Node.js, 2024) and the Express Framework (Express, 2024) so that users can directly operate the database through the back-end server. With these servers, DTM datasets can be transferred in two ways. One way is to publish the DTM datasets in PostGIS as web services such as WMS and WCS by the Geoserver. The other way is to communicate with the database and the front-end client through the back-end server, which can handle requests from the front-end client and access the database to generate data responses.
Besides, the back-end server also supports spatio-temporal analysis. For complex spatio-temporal analysis, (GDAL, 2024) - being more efficient than the corresponding PostGIS functions - is used as a library for reading, manipulating, and writing raster data. Therefore, spatio-temporal analyses such as the comparison of two versions of a DTM between two timestamps and the total change calculation, are handled by GDAL based on a Python server. Code 5 shows the elevation difference calculation between *DTM1* and *DTM2* representing two timestamps, computed by GDAL.

Code 5. Elevation difference calculation between two DTMs calculated by GDAL being integrated into the prototype system.

```
def DTM_difference(DTM_1_path, DTM_2_path, result):
    DTM_1 = gdal.Open(DTM_1_path)
    DTM_2 = gdal.Open(DTM_2_path)
    #Resample the first DTM data to the same as the second
DTM data

DTM_1_resampled=gdal.Warp(DTM_1_resampled_path,raster1,
format="GTiff",width=width2,
height=height2,dstSRS=projection2,resampleAlg=gdal.GRA_B
ilinear)
    # Calculation
    band1 = DTM_1_resampled.GetRasterBand(1)
    band2 = DTM_2.GetRasterBand(1)
    data1 = band1.ReadAsArray()
    data2 = band2.ReadAsArray()
    subtracted_data = data1 - data2
    output_band = output_raster.GetRasterBand(1)
    output_band.WriteArray(subtracted_data)
```

In GDAL, *GDAL.Open()* opens these two DTM datasets and *GDAL.ReadAsArray()* reads their bands as arrays. If the resolution of the two DTM data is different, *GDAL.Wrap()* is applied to up-sample the DTM data with the lower resolution to the same resolution as the other one. When they have the same resolution, they can be subtracted through the raster array calculation.

5.3 Front-End Layer

The *front-end layer* enables users to directly use the system via the web client. The (React Framework, 2024) was chosen as the development environment for creating a responsive GUI with JavaScript. With features such as component-based architecture and Virtual Document Object Model (DOM), React Framework enables quick dynamic updates of DTM data query results, as well as provides flexible integration with other libraries. Operations on the map such as drawing shapes, displaying popups, and visualizing DTM data are realized on the basis of (Leaflet, 2024). Data transmission in the web client is achieved by sending HTTP requests via (JQuery, 2024). The communication is followed by RESTful web services (Richardson and Ruby, 2008), which makes it easy to perform data interactions in the front-end client, such as sending GET requests to visualize database query results and POST requests to transmit DTM data to the server for spatio-temporal analysis.

Code 6. Drawing interaction to specify the region of a spatial query executed by PostgreSQL/PostGIS

```
map.on('draw:created', function(e) {
const polygon =
JSON.stringify(e.layer.toGeoJSON().geometry.coordinates)
;
const url =
`http://localhost:3000/query_by_polygon?polygonarray=${p
olygon}`;
$.ajax({
   url: url,
   type: 'GET',
   success: function(response) {
   alert('Specify the region successfully')
   },
   error: function(error) {
   console.log(error, 'error');
   }
  });
});
```

As shown in Code 6, when the draw event is finished, the polygon coordinates are sent to the backend server to execute spatio-temporal queries in PostGIS.

5.4 Integrated Spatio-Temporal Data Management and Analysis

Based on the implementation architecture, the workflow of integrating spatio-temporal data management and analysis is described in Figure 5.

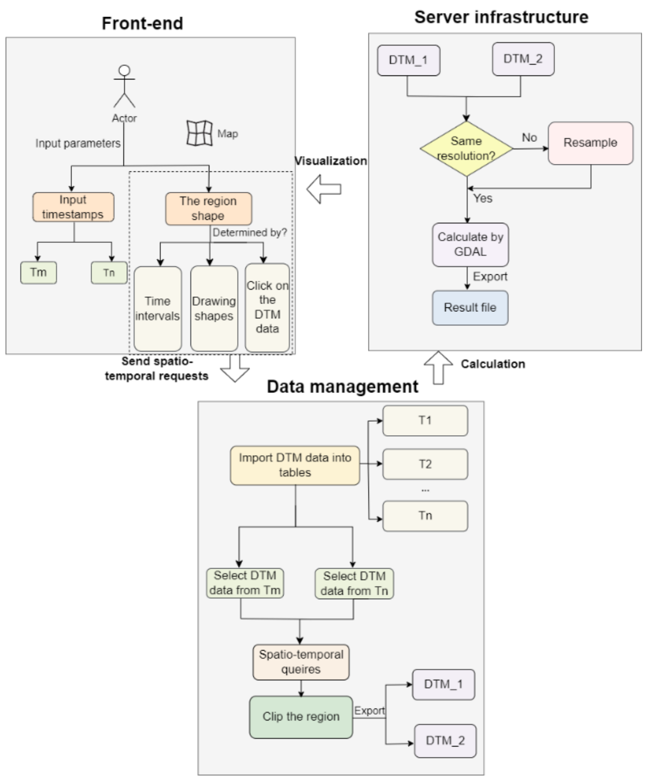


Figure 5. The workflow of spatio-temporal DTM data management and analysis.

Using the front-end client, the user can input analysing parameters including two timestamps *Tm* and *Tn*, and region shapes. The region shapes are determined by texts such as time intervals, or map interaction results such as drawing shapes to acquire the boundary of a region of interest. These parameters are sent as spatio-temporal requests to PostgreSQL/PostGIS. PostgreSQL/PostGIS internally stores DTM datasets into multiple tables according to the timestamp <T_1, T_2, …, and Tn>. With these parameters, spatio-temporal queries are executed with related DTM datasets from the tables *Tm* and *Tn*. Subsequently, DTM datasets are clipped by the region query and exported as two files, *DTM_1* and *DTM_2*. Then these two files are imported into the server infrastructure for further analysis. If the resolution is inconsistent, it is important to resample DTM datasets before the calculation. As described in section 5.2, GDAL is used to calculate the raster difference between these two DTM datasets. Finally, the analysis result is exported as a GeoTIFF file and transmitted to the front-end client for visualization.

6. Use Case: A Practical Field Test with DTM Data from South-West Germany

In order to evaluate the integrated spatio-temporal data management and analysis, we have tested real DTM datasets provided by the State Office for Geoinformation and Land Development (LGL) located in Karlsruhe, Baden-Württemberg, South-West Germany. Since the year 2000, LGL has held three Airborne Laser Scanning Campaigns *ALS_1* (2000-2016), *ALS_2* (2016-2021), and *ALS_3* (2022- expected 2028) for DTM data acquirement in the region of Baden-Württemberg. All three campaigns cover the same area, but different DTM data resolutions, with *ALS_1* providing 1x1m resolution, and *ALS_2* and *ALS_3* providing 0.25x0.25m resolution.

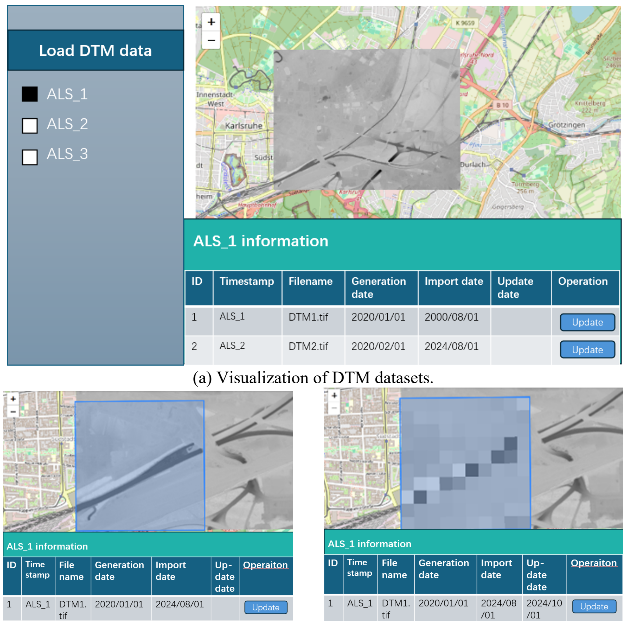
6.1 Data Management

To apply the Event-Based Time-Stamping Data Model to real DTM datasets, these airborne laser scanning campaigns *ALS_1*, *ALS_2*, and *ALS_3*, respectively, correspond to the timestamps *T1*, *T2*, and *T3*. Thus, we have imported these DTM datasets into three tables in PostgreSQL/PostGIS. Each table contains seven attributes that are *Rid*, *Rast*, *Filename*, *Timestamp*, *Generation date*, *Import date*, and *Update date*, as shown in Table 1. This example describes that the first DTM dataset from *ALS_1* was acquired on January 1st, 2000, imported into the database on April 1st, 2024, and updated on July 1st, 2024.

Attribute	Data type	Example
rid	integer	1
rast	raster	01000001000B000...
filename	text	32457_5427.tif
timestamp	text	ALS_1
generation date	date	2000-01-01
import date	date	2024-04-01
update date	date	2024-07-01

Table 1. Attributes and types of DTM data example.

On the front-end client, DTM datasets at different timestamps can be visualized as separate layers, and the visibility of each layer can be controlled through a layer tree interface, as shown in Figure 6(a). When the user selects a layer, the corresponding information table will pop up, with functions such as import, update, and deletion of DTM datasets. Figure 6(b) and Figure 6(c) show the result of updating the DTM dataset with a lower-resolution version.



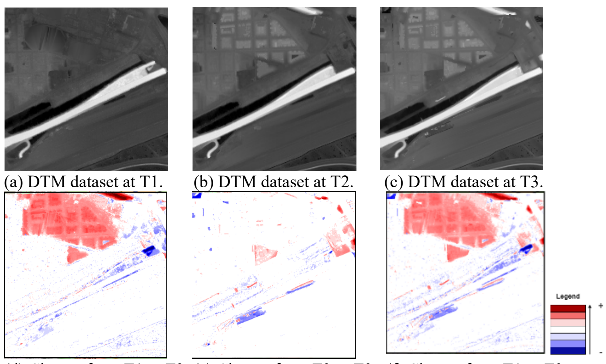
(a) Visualization of DTM datasets.
(b) Before updating a spatially selected region. (c) After updating the region with a lower-resolution version

Figure 6. Example of the “before-after” visualization: spatio-temporal DTM update operation

Compared to a direct use of PostGIS, our prototype system can manage spatio-temporal DTM datasets straightforward and more user-friendly.

6.2 Spatio-Temporal Analysis

As a prominent example of spatio-temporal analysis for DTM datasets in our prototype system, we implemented a spatial window query to compare the elevation values of a DTM region between two arbitrary timestamps. Figure 7 is an example to generate the elevation change from T1 to T2, T2 to T3, and T1 to T3. The deeper red areas reflect the increasing elevation, and the deeper blue areas show the decrease in the elevation value. When the user clicks on the result pixel, she or he can acquire the elevation change at this specific point.



(a) DTM dataset at T1. (b) DTM dataset at T2. (c) DTM dataset at T3.
(d) Change from T1 to T2. (e) Change from T2 to T3. (f) Change from T1 to T3.

Figure 7. Spatio-temporal analysis result showing DTM height changes.

In total, the results demonstrate that the approach for spatio-temporal DTM data management and analysis proposed in this paper is feasible and effective for the used datasets. However, several challenges were identified during the evaluation process:

(1) Provision of inconsistent coordinate systems in the sample data sets.

(2) Unsatisfactory loading time of DTM data in the GeoDBMS.
(3) Missing of big spatio-temporal data.

The shortcomings highlighted above provide scope for future research concerning the management of big DTM data and machine-learning based approaches for the preparation and analysis of DTM data including the comparison, i.e. detection and correction of incompatible coordinate systems.

7. Future Geodetic Applications and new Research Directions

There is an increasing need for DTM research in emerging geodetic applications such as the small-scale calculation of the Earth´s gravity field and the large-scale monitoring of floors by mobile robotic platforms providing high-precision measurements. Taking up the first mentioned application, the gravity field reflects the mass distribution in the Earth's system and provides important information for the monitoring of changes and anomalies related to climate change. Obviously, to get a globally homogeneous picture, the use of satellite-based data is indispensable. However, satellite observations are not able to capture the fine structures of the gravity field that are induced by topographic masses, e.g., the masses of mountains, oceans or ice sheets. These high-frequency signal components can be obtained by gravity forward modelling that uses a DTM to discretize the geometry of the topographic masses (Grombein et al., 2016).
For this purpose, each DTM grid cell represents a mass body defined by the DTM height, the DTM resolution and a suitable density assumption. Depending on the spatial distance between a mass body and the observation point (point of interest), the gravity effect can be calculated by (numerical) integration (e.g., Grombein et al., 2013). According to the superposition principle, the attraction of the entire topographic masses is then approximated by the sum of the effect of all mass bodies.
In principle, this procedure requires to calculate the impact of globally distributed mass bodies, which is a very time-consuming task. However, due to the distance dependency of gravity, the contribution of mass bodies decreases with increasing distance. This allows to use a coarser DTM resolution for mass bodies located far away from the observation point without losing significant precision. So far, this has usually been considered by using two DTMs with different resolutions for the near and far field around the observation point. However, for an efficient calculation, it would be desirable to let the GeoDBMS calculate and store a dynamically changing DTM resolution that is customized to the distance between a respective mass body and the observation point.
Taking up the second application mentioned above, also on a large scale, respectively within buildings and industrial halls, the provision of DTMs and their smart processing is essential. For example, in the construction industry, after the installation of flooring or the placement of screeds, an acceptance inspection takes place. E.g. in Germany, such a flatness control is typically carried out according to the German Industry Standard "DIN 18202:2019-07". The spatial representation then is a DTM with a fixed grid (e.g. a one or two-meter grid), from which parameters for flatness are derived.
In the past, such a control was done manually using a level or a tacheometer. As a result of automating this manual process with the a mobile robotic platform RITA (Naab, 2022), many advantages are created for the user including uncertainty modeling (Ulrich, 2016). In addition to fatigue-free work, consistent high accuracy, or on-site data evaluation, it is economically feasible, for example, to regularly control floors for wear and tear. Especially in high-bay warehouses with automated material handling equipment, the rocking of these industrial trucks is a safety concern and therefore continuous monitoring is required.
The robot also makes it possible to generate DTMs with a high resolution. The location accuracy of the individual data points is within a few millimeters due to the kinematic characteristics of the robot, even in combination with a tacheometer (Naab and Zheng, 2021). Due to the higher data density and the recurring monitoring of floors by the mobile robotic platform, effective and efficient management and analysis of spatio-temporal data is necessary. At the same time, it must be ensured that the determination of parameters for flatness control from the DTMs is performant.

8. Conclusions and Outlook

In this paper we presented an integrated approach for the storage, retrieval, analysis and visualization of spatial DTM changes. Based on advantages of former time-stamping and event-based approaches, a so-called event-based time-stamping data model was introduced. A prototype implementation of the approach was presented and evaluated with DTM datasets from South-West Germany. The result demonstrates that our method enables efficient management and visualization of DTM datasets across multiple separate layers, as well as it supports spatio-temporal retrievals and dynamic updates from the event perspective. Furthermore, it also enhances the ability of analysing elevation changes over time. Emerging geodetic applications dealing with different DTM scales, such as the calculation of the Earth´s gravity field (small scale DTM) and the monitoring of floors with mobile robotic platforms (large-scale DTM), showed the need of additional interdisciplinary DTM research.
From our preliminary experiences, we draw the following conclusions:

- The introduced spatio-temporal model for DTMs enables experts to deal with several events per timestamp (layer) for each DTM. Thus, an event summarizes the changed elevation values in a certain region of the DTM between two timestamps.
- Consistency checks between the collected and the stored DTM in the DBMS are of invaluable additional value.
- The proposed approach can also be applied to Digital Surface Models, which increases the user base even further.
- DTMs are an emerging interdisciplinary research field, being highly relevant for remote sensing, but also for other disciplines such as medicine (e.g. for retina analysis).

In our future research, we will focus on the interdisciplinary usage of DTM data management and analysis e.g. for autonomous driving (Wang et al., 2020), and DSMs containing vegetation and buildings, particularly in the rapidly expanded cities such as Dubai. Since the early 1980s, the United Arab Emirates have been undergoing remarkable urban expansion at an unparalleled pace. DTMs and high-resolution satellite images, both managed in a DBMS, will provide DTM representations on demand (Mina Al-Saad et al., 2021).

Acknowledgements

We sincerely thank the LGL (Landesamt für Geoinformation und Landentwicklung, Karlsruhe) for providing the DTM datasets and for accompanying the DTM project.

References

Armstrong, M. P., 1988. Temporality in spatial databases. *GIS/LIS 88 Proc.: Accessing the world*, 880-889.

Baumann, P., 2010. The OGC Web Coverage Processing Service (WCPS) Standard, *Geoinformatica*, 14(4)2010, pp. 447-479.

Baumann, P., Furtado, P., Ritsch, R., Widmann, N., 1997. The RasDaMan Approach to Multidimensional Database Management. *Proc. 12th Annual Symposium on Applied Computing (SAC'97),* San Jose/USA, February 28 - March 2, 1997.

Baumann, P., Misev, D., Merticariu, V., Pham Huu, B., 2018. Datacubes: Towards Space/Time Analysis-Ready Data.. In: J. Doellner, M. Jobst, P. Schmitz (eds.): Service Oriented Mapping - Changing Paradigm in Map Production and Geoinformation Management, *Springer Lecture Notes in Geoinformation and Cartography.* pp. 269-299.

Breunig, M., Bradley, P.E., Jahn, M., Kuper, P., Mazroob, N., Rösch, N., Al-Doori, M., Stefanakis, E., Jadidi, M., 2020. Geospatial Data Management Research: Progress and Future Directions, ISPRS International Journal of Geo-Information 2020,9(2), 95, MDPI, https://doi.org/10.3390/ijgi9020095.

Breunig, M., Schilberg, B., Thomsen, A., Kuper, P.V., Jahn, M., Butwilowski, E., 2010. DB4GeO, a 3D/4D geodatabase and its application for the analysis of landslides. *Geographic Information and Cartography for Risk and Crisis Management: Towards Better Solutions,* Springer Berlin Heidelberg, 83-101.

DIN 18202:2019-07, 2019, Toleranzen im Hochbau – Bauwerke. Berlin, Germany: Beuth Verlag GmbH.

Doyle, F.J., 1978. Digital Terrain Models: An Overview. Photogrammetric Engineering and Remote Sensing, Vol. 44, No. 12, December 1978, pp. 1481-1485.

Geoserver, 2024, Open source server for sharing geospatial data, geoserver.org (12 October 2024).

Express, 2024. Express - Node.js web application framework. https://expressjs.com/.

GDAL, 2024. translator library for raster and vector geospatial data, gdal.org/en/latest/ (12 October 2024).

Grombein, T., Seitz, K. und Heck, B., 2013. Optimized formulas for the gravitational field of a tesseroid. Journal of Geodesy 87(7):645–660. doi: 10.1007/s00190-013-0636-1.

Grombein, T., Seitz, K. und Heck, B., 2016. The Rock-Water-Ice topographic gravity field model RWI_TOPO_2015 and its comparison to a conventional rock-equivalent version. Surveys in Geophysics 37(5):937–976. doi: 10.1007/s10712-016-9376-0.

jQuery, 2024. A JavaScript utility library, jquery.com (12 October 2024).

Kamp, N., Krenn, P., Avian, M., Sass, O., 2023. Comparability of multi-temporal DTMs derived from different LiDAR platforms: Error sources and uncertainties in the application of geomorphic impact studies. Earth Surf Prcess Landf. 2023, May, 48(6), pp. 1152-1175. Doi: 10.1002/esp.5540.

Langran, G., Chrisman, R., 1988. A framework for temporal geographic information. Cartographica: The International Journal for Geographic Information and Geovisualization 15.3 (1988), pp. 1-14.

Leaflet, 2024. JavaScript library for interactive maps, Leaflet.js., https://leaflettjs.com (12 October 2024).

Liu, R., 2024. Spatio-Temporal Analysis of Digital Terrain Models supported by a Geodatabase Management System. Master Thesis, Karlsruhe Institute of Technology, Department of Civil Engineering, Geo and Environmental Sciences, Geodetic Institute, Chair of Geoinformatics.

Miller, C.L., 1958. The theory and application of digital terrain model. Thesis (M.S.). Massachusetts Institute of Technology, Dept. of Civil and Sanitary Engineering, http://hdl.handle.net/1721.1/67142.

Mina Al-Saad, et al (2021), Protection and Authentication of Dubai Digital Elevation Model using Hybrid Watermarking Technique, 4th International Conference on Signal Processing and Information Security (ICSPIS), pp. 1-4.

Naab C., 2022. A mobile robot for monitoring floor flatness in real-time. Journal of Applied Geodesy, vol. 17, no. 2, 2023, pp. 109-117. doi: 10.1515/jag-2022-0046

Naab C., Zheng Z., 2021. Application of the unscented Kalman filter in position estimation – a case study on a robot for precise positioning. Robotics and Autonomous Systems 2022, 147:103904. doi: 10.1016/j.robot.2021.103904.

Node.js, 2024, Node.js — Run JavaScript Everywhere. Node.js: https://nodejs.org/en (12 October 2024).

Peuquet, D.J., Duan, N., 1995. An event-based spatiotemporal data model (ESTDM) for temporal analysis of geographical data. International Journal of Geographical Information Systems 9.1 (1995), pp. 7-24.

Polthier, K., Rumpf, M., 1994. A Concept for Time-Dependent Processes. Goebel et al. (Eds.), Visualization in Scientific Computing, Springer, Vienna, pp. 137-153.

React framework, 2024, React library for component-based web and native user interfaces, react.dev (12 October 2024).

Richardson, L., & Ruby, S., 2008. RESTful Web Services. O’Reilly Media, Inc.

Ulrich, T., 2016. Uncertainty modelling of high-precision trajectories for industrial real-time measurement applications, PhD Thesis, Faculty of Civil Engineering, Geo and Environmental Science, Karlsruhe Institute of Technology, KIT Open ID: 1000060440.

Tseng, C.-M., Lin, C.-W., Stark, C.P., Liu, J.-K., Fei, L.-Y., Hsieh, Y.-C., 2013. Application of a multi-temporal, LiDAR-derived digital terrain model in a landslide-volume estimation. Earth Surf. Process. Landforms 38, Wiley & Sons, Ltd., pp. 1587-1601.

Wang, Y., et al., 2020. Advances in autonomous driving: The role of digital terrain models, *Journal of Transportation Research*, 45(3), pp. 123-139.

Weibel, R., Heller, M., 1991. Digital terrain modelling. Geographic Information Systems, Principles and Applications, Longman, pp. 269-297.

Worboys, M., 2005. Event-oriented approaches to geographic phenomena. International Journal of Geographical Information Science 19.1 (2005), pp. 1-28.

Zhilin, L., Zhu, C., Gold, Ch., 2004. Digital terrain modelling: principles and methodology. CRC press.