Solar Potential Analysis and Integration of the Time-dependent Simulation Results for Semantic 3D City Models using Dynamizers

Kanishk Chaturvedi, Bruno Willenborg, Maximilian Sindram, Thomas H. Kolbe

Chair of Geoinformatics, Technische Universität München, Germany

3DGeoInfo Conference 2017
Melbourne, Australia
26-27 October 2017
Semantic 3D City Models and Simulations
Issues with current Semantic 3D City Models

► Most of the simulations involve time-dependent properties

- Variations of spatial properties: change of a feature’s geometry, both in respect to shape and to location (e.g. moving objects)

- Variations of thematic properties: changes of physical quantities like energy demands, solar irradiation; air quality in streets and buildings

- Variations of appearance properties: changes of textures or materials of city objects

- Variations with respect to sensor or real-time data

► Lack of support of such time-dependent properties

Solar Potential Simulation

- Simulation tool developed by Chair of Geoinformatics, TU Munich

- Operates on the CityGML standard

- Estimates solar energy production for the roofs and facades of the 3D buildings

- Implemented successfully for many cities such as London, Berlin, New York, and Helsinki

Willenborg, Sindram, Kolbe, 2017
Solar Potential Simulation use case in our study

OGC Future City Pilot Phase 1

District Bruz in Rennes, France

- \(\sim 6 \times 12\) km

- 10,916 buildings

- \(\sim 89,827\) Wall and roof surfaces

- \(\sim 4,324,505\) generic attributes

http://www.opengeospatial.org/projects/initiatives/fcp1
Solar radiation types

- Direct radiation
- Diffuse radiation
- Global radiation

- Reflected radiation is not considered
Generic Attributes

- Results are persistently stored and **linked** to their belonging city object

  ➜ Combine results with other data, like:
  
  - Energy demand estimation

- Data fusion: Enhanced analytic capabilities

```xml
<bldg:WallSurface gml:id="UUID_01_WS_1">
  <gen:doubleAttribute name="globalRadYear">
    <gen:value>77004.913</gen:value>
  </gen:doubleAttribute>
  <gen:doubleAttribute name="globalRadMonth_01">
    <gen:value>4293.446</gen:value>
  </gen:doubleAttribute>
  <gen:doubleAttribute name="globalRadMonth_02">
    <gen:value>5563.502</gen:value>
  </gen:doubleAttribute>
  <gen:doubleAttribute name="globalRadMonth_03">
    <gen:value>7010.33</gen:value>
  </gen:doubleAttribute>
  <gen:doubleAttribute name="globalRadMonth_04">
    <gen:value>7180.839</gen:value>
  </gen:doubleAttribute>
  <gen:doubleAttribute name="globalRadMonth_05"> <!-- all months in a year -->
    <gen:value>7180.839</gen:value>
  </gen:doubleAttribute>

  <gen:doubleAttribute name="globalRadMonth_06"> <!-- diffuse radiation -->
    <gen:value>4010.239</gen:value>
  </gen:doubleAttribute>
  <gen:doubleAttribute name="globalRadMonth_07"> <!-- direct radiation -->
    <gen:value>4010.239</gen:value>
  </gen:doubleAttribute>
</bldg:WallSurface>
```
Storage and Visualization of Simulation results

Simulation results stored as multiple static attributes
Limitations with CityGML 2.0

► Can we represent such time-dependent properties in a more dynamic way?
► Can we perform analyses and queries on timeseries data?
► Can we easily recognize and model patterns based on existing data model?
► Can we visualize timeseries data in a more interactive way?

► Yes - the solution is **Dynamizer ADE** for CityGML 2.0
City Objects

- Buildings
- Transportation Objects
- Water Bodies
- Vegetation

- Spatial Properties
- Thematic Properties
- Appearance Properties

Dynamizers

- Sensors
- Databases
- External Files

Tabulated data e.g. from simulation results or sensors

Chaturvedi & Kolbe, 2016
Example Scenario

CityGML object

<cityObjectMember>
<dyn:Dynamizer>
  <dyn:attributeRef> //bldg:WallSurface [@gml:id = 'UUID_01_WS_1'] /doubleAttribute[@name = 'globalRadMonth'] /gen:value
  <dyn:startTime> 2015-01-01T00:00:00Z </dyn:startTime>
  <dyn:endTime> 2015-12-31T00:00:00Z </dyn:endTime>
  <dyn:dynamicData>.. </dyn:dynamicData>
</dyn:Dynamizer>
</cityObjectMember>

Replacening dynamic attributes using XPath

One dynamic attribute which changes with time

Simulation Results

<table>
<thead>
<tr>
<th>Estimated</th>
<th>Global Radiation</th>
</tr>
</thead>
<tbody>
<tr>
<td>JAN-15</td>
<td>4293.446</td>
</tr>
<tr>
<td>FEB-15</td>
<td>5563.502</td>
</tr>
<tr>
<td>MAR-15</td>
<td>7010.33</td>
</tr>
<tr>
<td>DEC-15</td>
<td>4010.239</td>
</tr>
</tbody>
</table>

Representing data in standardized ways, such as OGC TimeseriesML, OGC Observations & Measurements
<-- timeseries data for representing interleaved values -->
<dyn:dynamicData>
  <dyn:AtomicTimeseries>
    <dyn:dynamicDataTVP>
      <tsml:TimeseriesTVP>
        <gml:description>Example of TimeSeries object with interleaved values</gml:description>
        <tsml:defaultPointMetadata>
          <tsml:PointMetadata>
            <tsml:quality xlink:href="http://www.opengis.net/def/tsml/1.0/quality/Good" xlink:title="Good"/>
            <tsml:uom code="KWh"/>
            <tsml:interpolationType xlink:href="http://www.opengis.net/def/waterml/2.0/interpolationType/Continuous" xlink:title="Continuous"/>
          </tsml:PointMetadata>
        </tsml:defaultPointMetadata>
        <tsml:point>
          <tsml:MeasurementTVP>
            <tsml:time>2015-01</tsml:time>
            <tsml:value>4293.446</tsml:value>
          </tsml:MeasurementTVP>
        </tsml:point>
        <tsml:point>
          <tsml:MeasurementTVP>
            <tsml:time>2015-02</tsml:time>
            <tsml:value>5563.502</tsml:value>
          </tsml:MeasurementTVP>
        </tsml:point>
      </tsml:TimeseriesTVP>
    </dyn:dynamicDataTVP>
  </dyn:AtomicTimeseries>
</dyn:dynamicData>
Better visualization of timeseries data

City model of Rennes, France
http://rennes.virtualcitymap.de/
Better management of timeseries data (I)

- Storing timeseries data in CityGML database for better analysis
- 3DCityDB can dynamically be extended to support storage of CityGML models with ADEs (Yao & Kolbe, 2017)
- Ongoing implementation with CityGML Dynamizer ADE

Diagram:

- CityGML with Dynamizer ADE
- 3DCityDB
- Dynamizer ADE
  - Dynamizer Core Module
  - Timeseries Metadata Module
  - Timeseries Module
- ADE 2
- ADE 3
Better management of timeseries data (I)

- Storing timeseries data in CityGML database for better analysis
- 3DCityDB can dynamically be extended to support storage of CityGML models with ADEs (Yao & Kolbe, 2017)
- Ongoing implementation with CityGML Dynamizer ADE

CityGML with Dynamizer ADE

3DCityDB Importer/Exporter

CityGML Module

Dynamizer ADE

Dynamizer Core Module

Timeseries Metadata Module

Timeseries Module

Dynamizer ID
Reference attribute
Atomic Timeseries
Composite Timeseries

Timeseries ID
Timeseries Type
Unit of measurement
Interpolation type
First and last timestamp
Etc.

Time 1, Value 1
Time 2, Value 2
Time 3, Value 3

Allows performing sophisticated queries on timeseries data
Better management of timeseries data (II)

► In many scenarios, such simulation results (timeseries data) may also belong to different sources
  ● External files (CSV, Excel sheets)
  ● Cloud based services (Google Spreadsheet, Fusion Tables)
  ● External databases
  ● External web services (OpenSensors, Wunderground, Thingspeak, OGC Sensor based services)

► Different data sources have
  ● different structures,
  ● data formats, and
  ● Interfaces

► How to work with all of such data sources in an interoperable way?
Mini Sensor Observation Service (Mini SOS)

- Very basic and lightweight service which can be instantiated for a given tabular source
- Mini SOS operates on arbitrary tabular data source
- Mini SOS provides multiple interfaces
- No internal/own storage of observation data within the Mini SOS

<table>
<thead>
<tr>
<th>Interface</th>
<th>Sensor Observation Service</th>
<th>Sensor Things API</th>
<th>52° North REST API</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mini SOS</td>
<td>Data Adapter</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Adapter</th>
<th>Relational</th>
<th>Proprietary Services</th>
<th>CityGML Dynamizer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CSV, Excel Google Spreadsheets JDBC Connections</td>
<td>OpenSensors Mathwork Thingspeak Any other</td>
<td>Atomic Timeseries Composite Timeseries</td>
</tr>
</tbody>
</table>
Easy access to CityGML Dynamizer Timeseries

- Mini SOS allows accessing timeseries data from CityGML Dynamizer
Conclusions and future work

► Solar potential simulation tool
- Estimates solar energy production for the roofs and facades of the CityGML buildings
- Enriches CityGML features with time-dependent results

► Dynamizer ADE
- Enhances static city models by dynamic and time-dependent properties values in standardized ways
- Supports complex patterns based on statistics and general rules
- Also provides direct explicit links to sensors
- Intended to become integral part of CityGML 3.0

► OGC Future City Pilot Phase 1

► Ongoing Work
- Import and Export of CityGML Dynamizer ADE with timeseries in 3DCityDB
- Easy retrieval of timeseries data using simplified Mini SOS