Transcription of the presentation of...

„TileGen – An open source software for applying cartographic Generalisatin to Tile-Based Mapping“

...held at the International Cartographic Conference 2013 in Dresden, Germany

Ralf Klammer, TU Dresden, 28.08.2013

Not everybody will be familiar but surely everybody has already heard of the term ***„Web 2.0“*** *[O'Reilly (2005)]*, which basically concludes – from my point of view – the shift in the structure of communication of users in the web from ***pure consume*** to an ***active production*** of web contents … the so called '***ProdUser***' *[Budhathoki, 2008]*! In relation to cartography, these trends result in the emergence of a new kind of '***neocartographic*** *[Gartner und Schmidt, 2010]* ***mapping community'***, which is primarily characterized by a very heterogeneous composition of participants – from cartographic experts to ordinary persons.

My aim is to find the possibilities for broadcasting, spreading and applying the concept of web generalization services within the “neocartographic mapping community” and to detect the requirements for further developments on the existing 'WebGen'-project *[Foerster et al., 2008]*.
What is “WebGen”? Everything began with the basic, but great idea of sharing knowledge, functionalities and algorithms on cartographic generalization over the internet, with primary focus on the scientific community → “***WebGen***”
A 1st idea of broadcasting the project to a larger audience, especially national mapping agencies, was to adept the accessibility of the services to the communication structure to the OGC standard of ***Web Processing Services*** (WPS). This stage of development is the current status-quo of the project → “***WebGen-WPS***”
But what has to be done to make this accessible for the audience of Web 2.0? Two major points of action are possible:

1. apply “WebGen” within different neocartographic (Web 2.0) mapping applications and detect need for further developments
2. develop “WebGen” to make it accessible for users, as adaption to the Web 2.0 communication → “***WebGen 2.0***”

Point 1 is the origin for the development of ***TileGen***, as many neocartographic are build as or use (at least in background) the technology of tile-based maps!

What is ***tile-based mapping***? Tile-Based mapping is a technique that has already been used for a long time (1980s) within computer games, it has got a significant boom since GoogleMaps was published (mid 2000s) and is – by now – a commonly accepted and widely used technology for efficient web mapping!
The term ***tile-based*** signifies the display of a huge digital image by a combined visualisation of smaller, single images … called tiles or in our case ***map tiles***.
Accordingly, ***tile-based mapping***, is the process of computing all the single map tiles, used for the overall display of a huge ***tile-based map***. As you can see in the definition

of Sample & Ioup (2010), each map tile is always based on a discrete zoom level, which corresponds to a fixed map scale, and the tiles are addressed by a discrete tiling scheme which is needed for a fast and uncomplicated assembling of the overall map view at the client. Finally, the term ***tile-based mapping system*** summarizes the process chain, beginning with data manipulation and definition of cartographic parameters over the rendering, storage management and server communication to the final display for example in a web browser.

Let's have a look on the tasks that have to be mastered when you want to set up a tile-based mapping system!

1. First of all, you'll ***organize your data***. Most of us will do that within in a standard GIS.
2. Secondly, you have to ***define the appearance*** (colors, symbolisation, etc.) of the tile-based map by defining the cartographic parameters for each (finally) available zoomlevel (map scale). This can be a very abstract process as these parameters have to be set up – in most rendering libraries - in an XML-file, defining styles and rules (comparable to OGC-standards SLD & SE).
3. Third, you ***initialize*** the completely automatic ***rendering*** of all necessary map tiles.
4. Now, the tile-based map is completely calculated but you need to display it to your audience. Therefore, you have to ***build a mapping framework***, which is mostly a JavaScript-framework (e.g. Lealflet, OpenLayers, etc.) within the context of web mapping.
5. Finally, you can ***add mash-ups, nice interactive elements or statistics*** to the tile-based map, to describe the favored subject as good as possible.

The point at which we can apply generalization functionalities seems to be clear, we generalize the geometries when we organize the data and store them as multiple forms of representation. But this is only applicable in a manual process flow. Tile-Based mapping systems are mostly build as automatic process flows, wherefore it is necessary to define the favored generalization tasks within the cartographic parameters and let them execute automatically directly while rendering the map tiles. ***Managing these tasks is the job of TileGen!***

Basically, TileGen is a ***graphical user interface*** (GUI) that enables the map author to visually ***define the appearance*** of the tile-based map by getting an ***immediate preview*** of the specified cartographic parameters. This visual impression also enables the map author to directly identifying graphical lacks in map but also supports the application of schematic transformations.
On the other hand, TileGen acts like an ***analysis tool***, as it automatically evaluates certain topological constraints within the map tiles and offers these information to the map author.
But the real core of TileGen is, that it enables the map author to test different spatial transformations and 'play' with the requested parameters. This can be especially beneficial when a functionality has many input parameters. In the sense of re-usability, there will be no functionality implemented directly to the software. That is

the point where we need web generalization services! WebGen-WPS should act here as a kind of outsourced library for generalization functionalities.
After the user has tested different functionalities and played with values of the parameters, s/he can export the corresponding definitions of the WPS-Execute parameters and request the functionalities automatically while rendering the map tiles!

In order to prove the basic concept but also to test the applicability ot WebGen-WPS, I've implemented a proof-of-concept version in Python (PyGTK). The applied rendering library is Mapnik, which emerged as the primary rendering tool for the major map of OpenStreetMap and became in the meantime THAT standard rendering library for tile-based maps.
You can see, that the user is able to:

- read an XML-file, containing all cartographic parameters
- rendering map tiles for an immediate preview, within an definable extent
- select certain cartographic parameters, by selecting the corresponding definitions of layer, style & rule
- get information on automatic topological evaluation (currently on data density and minimum distances)
- access the WebGen-WPS, select a functionality and 'play' with the parameters
- evaluate the results of processing

The proof-of-concept implementation demonstrates the general feasibility of the proposed concept as well as the advantage of visual definitions on the definition of tile-based maps. There is still a lot to do (export & automatic processing, generic browser based online-version), but this depends on some further developments on WebGen-WPS, that have to be applied previously. These are especially:

- a more generic requesting of the WPS (especially no more GML as data format)
- make it more dynamic by let users upload own functionalities (algorithms)
- would be good request also the evaluation processes at the WebGen-WPS

If you like the basic concept and you're interested in further information, than have a look at the websites of the projects or contact me!

Sources:

- Budhathoki, N. R.; Bertram, B.; Zorica, N.-B. (2008). *Reconceptualizing the role of the user of spatial data infrastructure.* GeoJournal 72(3-4): 149-160.
- Gartner, G.; Schmidt, M. (2010). *Moderne Kartographie-Technologische Entwicklungen und Implikationen.* Kartographische Nachrichten, 6/2010, Kirschbaum, Bonn, S. 299-305.
- Foerster, T.; Burghardt, D.; Neun, M.; Regnauld, N.; Swan, J.; Weibel, R. (2008). *Towards an Interoperable Web Generalisation Services Framework – Current Work in Progress.* Proceedings 11th ICA Workshop on Generalisation and Multiple Representation, Montpellier.
- O'Reilly, T. (2005). *What Is Web 2.0? - Design Patterns and Business Models for the Next Generation of Software.* O'Reilly Media, URL:http://www.oreilly.de/artikel/web20.html.
- Sample, J. T.; Ioup, E. (2010). *Tile-Based Geospatial Information Systems*. Springer Science+Business Media.