

Online Visualization of Streaming Data

Caroline Atzl, Bernhard Vockner and Manfred Mittlboeck
RSA iSPACE Salzburg, Austria

Abstract

The integration of streaming data into web mapping applications in combination with contextual information is currently drawing attention in geospatial research. This article discusses how to communicate streaming data in the form of a live mapping dashboard. We created our own prototypical online dashboard visualization for this purpose, tailored to be used in different contexts and roles, and discuss the advantages of the approach. The live mapping dashboard uses modern visualization techniques, including dynamic clustering and self-updating symbology. We validated our approach in the context of a use case in the field of forest-based supply value chains. We present the data flow from the data collection to harmonization, right up to the presentation of real-time data. The prototype is evaluated in continuously performed field tests.

Keywords:

live mapping dashboard, streaming data, real-time data

1 Introduction

Nowadays, real-time data are indispensable. Globally, there are a vast number of dynamic and stationary sensors. These sensors provide different kinds of information. For example, there are weather stations that send live information about temperature, humidity and air pressure, and traffic sensors that measure weight in motion or speed, and which count the vehicles that pass. There are also many dynamic sensors, built into all kinds of devices, such as smartphones, and forms of transport (planes, buses, trains, trucks ...), which send the current position in combination with contextual information. The generic term for this kind of information is “real-time GIS data”, defined as “a continuous stream of events flowing from sensors where each event represents the latest state of the sensor” (Gorton, 2014).

Thanks to open data initiatives, access to streaming data is increasing. The challenge is to extract the right data out of that growing amount of data and to present it in an accurate and understandable way. In this context, we discuss how to visualize streaming data in a community dashboard that is tailored to be used in different contexts for different purposes. To evaluate our design concept, we prepared a prototypical online dashboard visualization for streaming data. Our approach uses various visualization techniques, which are especially

suitable for streaming data. These include dynamic clustering and self-updating symbology. We utilize our prototype in the field of forest-based supply value chains. In order to improve our prototype continuously, we integrate feedback from experts and users who are participating in on-going field evaluations.

2 State-of-the-Art

Real-time GIS data that is analysed, processed and visualized in the right way and provided to the right people at the right time can be an essential support, especially in time-critical situations, planning scenarios and fleet management. Many domains use real-time GIS data to get additional and meaningful information. Such data are used, for example, in:

- Emergency management: real-time information about water levels in flooded areas, knowing where the action forces are located at the moment
- Industry: tracking of products, warehouse stock, truck routes
- Transportation: arrival times, traffic jams
- Environmental: weather information
- Advertising: shopping centres, restaurants, pubs.

Dashboards are treated as solutions to visualize historical as well as streaming data in a relatively simple manner (Froese et al., 2013). In the GI domain, the ArcGIS Operations Dashboard, an app that shows real-time data feeds including different widgets, is widely used. To apply the Operations Dashboard, an ArcGIS organizational account is required. The Operations Dashboard can be used as a desktop app as well as in a web browser. In our case, we use open-source software to create the live mapping dashboard, as this approach does not depend on licensing (unlike, for example the ArcGIS Operations Dashboard), and offers more flexible configuration possibilities for a wide variety of purposes.

Guidelines on how to design user-friendly web maps (including dashboards) were proposed in Atzl (2015), while Froese et al. (2013) present the lessons learned from designing visualization dashboards. The results show that within visualizations, less is often more. Users preferred to have a prepopulated dashboard with default charts, which they can edit, if they want to avoid starting from scratch with a blank canvas (Froese et al., 2013).

To integrate streaming data into dashboards, there are two principal methods that enable client–server communication. The data dissemination can be either pull- or push-based. The pull approach requires the client to periodically retrieve data from the server using a certain time interval to get updates (Bhide et al., 2002). In contrast, if the server supports push mechanisms, the client automatically gets data that have changed (Bhide et al., 2002). For example, if a (transactional) web feature service (WFS, WFS-T) is applied, the request for the data happens at fixed intervals on the client-side using e.g. a trigger (pull). On the other hand, if the web socket protocol is used for data transmission, the client subscribes to receive data immediately when a change is made (push). By default, the push-based approach does not archive the data, whereas a pull-based approach does this by definition.

In addition, there is a trend in enhancing real-time GIS data with alerts or notifications. “Geofencing” is a term that is often used within this context. Geofencing enables remote monitoring of geographic locations, which are defined by virtual boundaries; an automatic trigger sends a message when a tracked device enters or exits a location (Reclus & Drouard, 2009).

The combination of streaming data and notifications based on geofences in dashboards can be used to support time-critical decision making. Due to the huge amount of data in these visualizations, special methods and techniques are needed to represent the information in a simple, user-centric way.

3 Use Case Description

The use case used to validate our approach in visualizing streaming data in dashboards is situated in the context of fleet management in the forest-based supply value chain. There are different roles involved in the process, from managing woodpiles and loading/unloading logs to delivering them. The different roles perform different tasks and thus need different information and applications.

For example, trucks carry woodpiles from forest storage locations to sawmills. The truck drivers use a prototypical app, which sends their individual positions and additional contextual information (e.g. positional accuracy, horizontal ground speed, loading/unloading status) at a specific interval (e.g. 30 seconds). This information is provided as a standardized WFS(-T) that is integrated and visualized in a web map, which is part of the dashboard. This app is important for the office workers to manage truck routes and estimate delivery times.

The live mapping dashboard contains further web applications, tailored to the different roles, for community-based management of the number of woodpile storage locations and logs (in solid cubic metres). For example, the logger can mark the woodpile locations and the available solid cubic metres, and the truck driver can input both the planned deliveries and the number of solid cubic metres already delivered. This information is available in various web mapping applications in the dashboard, which are especially suited for the target users.

There is a further web mapping application for creating, editing and deleting geofenced areas, which are represented as polygons on the map. If a truck driver enters one of these geofenced areas, the driver’s precise location is highlighted on the map. For instance, if the geofences represent a 15-minute drive-time buffer around the sawmill locations, the map can be used to inform the workers in the sawmill that a truck will be arriving soon.

In our implementations, we applied the pull-based mechanism to send a frequent request to the WFS(-T) to ask for new data from, or to send new data to, the truck drivers. The pull-based approach is used for showing information that does not have to be saved or updated.

The sections which follow describe the workflow of our approach and the prototypical dashboard visualization in more detail.

4 Data Flow

The integration of real-time data into the web mapping dashboard follows a multi-tiered architecture. This information is collected using a prototypical app (SenserAPP III) for mobile devices that can be placed inside vehicles. For the overall architecture, we used open-source software components and frameworks.

Figure 1 shows the workflow of our approach. The SenserAPP III streams data collected by the trucks to a geofence service. A geofence service is used to check/change the status of the truck (e.g. loading/unloading is only possible near particular places, defined by polygons) and to transmit the truck data to an OGC sensor observation service (SOS).

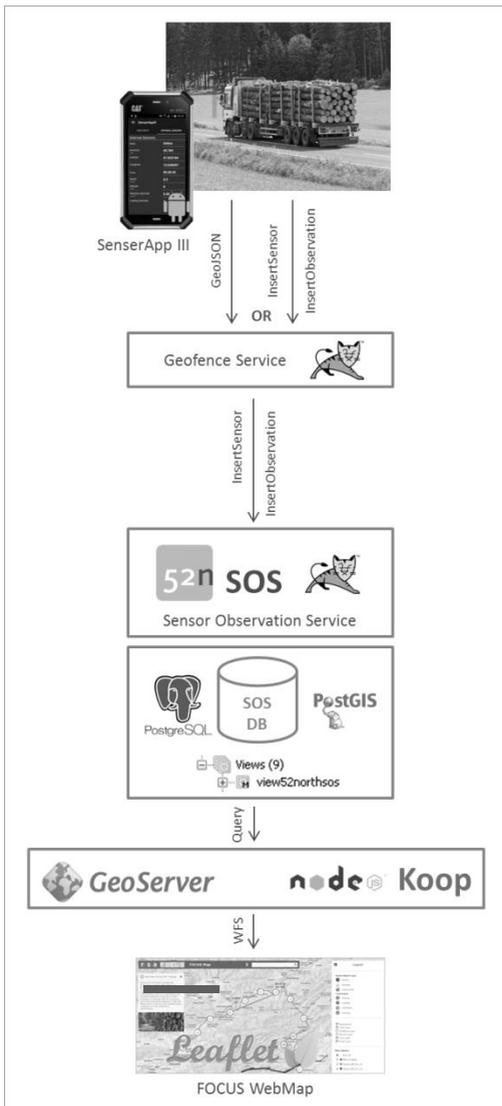


Figure 1:
Data flow, from collection of data
right up to visualization

To validate our proposed workflow, we applied the 52°North reference implementation of the OGC SOS (52°North, 2015). The OGC SOS stores the observations in a PostgreSQL/PostGIS database. Database views are used to pre-process the data and to provide them in a valid format to a GeoServer or Koop service. A GeoServer is a Java-based open-source server for organizing and publishing geospatial data (GeoServer, 2014); Koop provides a server that enables access to database content as services in different formats such as GeoJSON, Shapefile and KML (Koop, 2016). The GeoServer or Koop service is used to provide the map input. For visualization purposes, we used Leaflet, an open-source JavaScript library for interactive maps (Leaflet, 2015).

Further real-time data are transmitted using web mapping applications that were developed for updating/sending information about the woodpile storage locations using a WFS-T.

5 Dashboard Visualization

The final results are preconfigured web mapping applications that are part of the FOCUS live mapping dashboard. Figure 2 shows the home page of the dashboard. Within six apps, users can create/edit information. Within two streaming apps, the live information of the trucks can be visualized. Due to the potential use of different types of devices (smartphones, tablets and desktop PCs) in portrait or landscape mode, we implemented our applications with a responsive design approach. If the user turns the device or changes the window size of the browser, the menu is set automatically to the appropriate position. In addition, the header and the symbols are displayed in accordance with the screen width. The navigation area and the button for changing the base map are moved if the legend or the edit menu is opened so that the user can still use these buttons. If the height of the browser window is too small to show the whole content of the window, a scroll bar appears at the right-hand side.



Figure 2: FOCUS live mapping dashboard home page

One of the preconfigured streaming dashboard apps shows all data that have been collected within the last four hours (Fig. 3). The basic template of all the dashboard visualizations has the following layout elements:

- **Header:** The header contains logos that can be used to open the corresponding website in a new browser window. In addition, there is the map title, a button for toggling the information window and for showing the actual position of the user on the map, and an address search field. In the case of the editing maps, there are also buttons for user modifications (e.g. create, update, delete markers).
- **Information window:** This window provides a short description about the use case and the map content.
- **Change background button:** The button at the bottom-left can be used to switch from the street map to the satellite image, and vice versa.
- **Legend menu:** This menu provides information about the layer symbology. The user can turn the layers on and off, filter the content, and navigate within the overview map.
- **Navigation buttons:** In addition to map navigation using a mouse, there are three buttons, bottom-right, to zoom in, re-set the initial map extent, and zoom out.

There are layers that provide general environmental information, such as one showing woodpile storage locations, for example. This information is collected by mobile devices and can be clicked on the map for useful information, which appears in a pop-up window (Fig. 3).



Figure 3: Web map of the live dashboard showing a truck track over four hours

Depending on the GPS recording interval set in the SenserApp III (Fig. 1), the amount of data collected over the previous 4 hours could be very high. Thus, there may be many points on the map that have almost the same position. This would mean that if the zoom-level gets smaller, the number of overlapping points increases and the user does not know how many points exist at a given location. To overcome this, a dynamic clustering of points is implemented in the “Streaming App – 4h” (Fig. 2 bottom-right, Fig. 3). If the user zooms in or out, the clusters are adjusted to fit the actual map extent. It is also possible to click on the cluster markers, to zoom in, and to centre the map on this particular location.

If the maximum zoom level is reached and overlapping points still exist, a spiral-shaped visualization of the points is used (see Fig. 4). The spiral is sorted using the timestamp of the points so that the most recent point is the furthest from the actual location. Each single point can be clicked to open a pop-up window that contains specific contextual information. The colour of the points changes dynamically and indicates the state of the truck (e.g. green means that the truck is moving, and orange indicates that the truck is loading). In addition, the transparency value gets higher the older the timestamps of the data points. This means that the oldest point is completely transparent (the transparency value is one) and the most recent one has full colour (the transparency value is zero). This allows the user to recognize the direction in which the truck driver is moving and to identify the newest point quickly.



Figure 4:
Spiral spreading of overlapping points when the maximum zoom level is reached

The “Streaming App – Top5” web map (Fig. 2 top-right) contains the last five points of the active trucks (Fig. 5). In this visualization, the points are connected, the most recent point is highlighted, and the older points get a higher transparency value. If a truck is inactive (the last signal was sent more than 15 minutes ago), the specific truck track appears in grey.

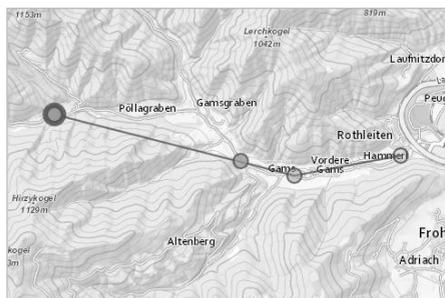


Figure 5:
Last five points, with different transparency values, connected by lines

The live mapping dashboard contains another app that shows the different sawmill locations and geofences within a 15-minutes radius (Fig. 2 “Edit App – Geofences”, Fig. 6). The actual truck position is visualized as a pulsing point to identify the actual trucks immediately. Therefore, a web socket is used to show only the most recent position of each truck; older points are not relevant. If a truck enters one of the geofences, an alert can be sent to inform the customer that a delivery is imminent.

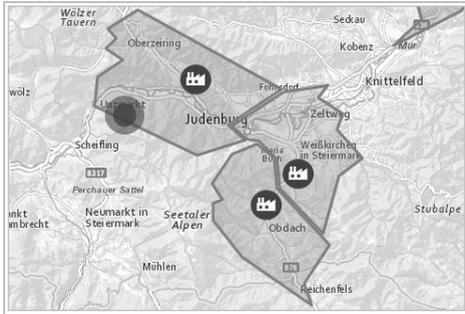


Figure 6:
Blinking point shows that a geofence has been entered

The data stored within the database are also used to create informative charts. For example, Figure 7 shows pie-charts as symbols used in the “Edit App – Creator” and “Edit App – Driver”. The charts represent the percentage of solid cubic metres per sawmill, because each woodpile storage location can contain logs for different sawmills. The size of the symbol shows the proportion of solid cubic metres in relation to the overall amount of wood available on the map.

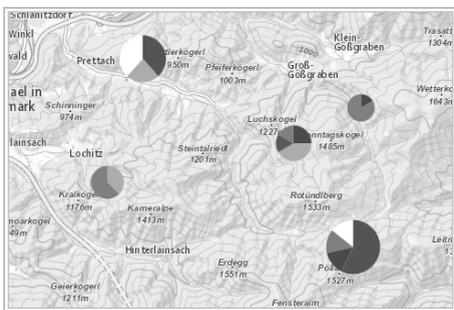


Figure 7:
Percentage representations of proportions of logs per sawmill

These different examples of visualization strategies for showing real-time data in a live mapping dashboard illustrate how different roles and contexts need different information and representations. The whole dashboard was evaluated in field tests and the visualizations were adapted with respect to the needs of the target users and their feedback.

6 Validation

Throughout the whole project, our prototype was evaluated by experts, who provided us with feedback from users testing the prototype directly in the field. The applications were continuously improved to fulfil the requirements of the users. Major remarks included that the size of the symbols in Figure 7 should indicate the solid cubic metres as percentages of the absolute value and that the fonts were too small on some mobile devices. We also tested the apps on different devices and in various web browsers.

7 Conclusion

Streaming different kinds of geospatial data in combination with real-time visualizations is still a major challenge. In this article, we have presented a solution that uses a WFS(-T) to integrate streaming data into a live mapping dashboard in the context of fleet management. Our prototype includes tracks and charts that dynamically represent information such as the proportions of logs per sawmill at each site.

Using a WFS(-T) requires a periodical request to the service for an update of data. We also used the web socket protocol that pushes data updates to the client after a connection is established. In contrast to the WFS(-T) approach, data in a web socket is not automatically stored. We therefore used the web socket approach to visualize only data regarding whether a truck was inside a geofence or not. In other cases, access to previously collected data is needed. Thus, the WFS(-T) approach was chosen for most data sets because of its storing of data in a (geo-)database and the possibility to create, update and delete entries from the client-side. In our case, we used both approaches depending on the content and functionality of the different web mapping apps of the live dashboard.

In contrast to other dashboard visualization approaches, such as the ArcGIS Operations Dashboard, our approach is based entirely on open-source products and offers more flexible configuration possibilities for a wide variety of purposes.

In addition, there are many possibilities for communicating and visualizing streaming data in web mapping applications, such as dynamic clustering, self-updating symbols, charts and animated icons. Depending on the user's role and task, the visualization strategy has to be adapted. Responsive design is another important issue to render the web mapping applications suitable for use on different devices (smartphones, tablets and desktop PCs). Our live mapping dashboard shows one possible solution for a simple, innovative, user-centric way to communicate data streams.

Acknowledgment

This research work is part of the EU FP7 project FOCUS.

References

- 52°North (2014). Sensor Observation Service. Available online at <http://52north.org/communities/sensorweb/sos/> [accessed 2016-01-27]
- Atzl, C. (2015). How to Design Web Maps that Users Like? In S. Hennig (Ed.), *Online-Karten im Fokus: Praxisorientierte Entwicklung und Umsetzung* (pp. 33-51). Köthen, Germany: Wichmann
- Bhide, M., Deolasee, P., Katkar, A., Panchbudhe, A., Ramamritham, K., & Prashant, S. (2002). Adaptive Push-Pull: Disseminating Dynamic Web Data. *IEEE Transactions on Computers*, 51/6, pp. 652-668
- Bramer, M., & Sunderman, R.J. (2015). Real-Time GIS: Leveraging Stream Services. Available online at http://proceedings.esri.com/library/userconf/proc15/tech-workshops/tw_551-79.pdf [accessed 2016-01-28]
- Froese, M., Tory, M., Evans, G., & Shrikhande, K. (2013). Evaluation of Static and Dynamic Visualization Training Approaches for Users with Different Spatial Abilities. *IEEE Transactions on Visualization and Computer Graphics*, 19/12, pp. 2810-2817
- GeoServer (2014). What is GeoServer? Available online at <http://geoserver.org/about/> [accessed 2016-01-27]
- Gorton, K. (2014). Building Real Time Web Applications with GeoEvent Processor. Available online at http://proceedings.esri.com/library/userconf/devsummit-dc14/papers/dsdc_09.pdf [accessed 2016-01-26]
- Koop (2016). Transform, Query, and Download Geospatial Data on the Web. Available online at <https://github.com/koopjs/koop> [accessed 2016-01-27]
- Leaflet (2015). An Open-Source Javascript Library for Mobile-Friendly Interactive Maps. Available online at <http://leafletjs.com/> [accessed 2016-01-27]
- Reclus, F., & Drouard, K. (2009). Geofencing for Fleet & Freight Management. *Proceedings of 9th International Conference on Intelligent Transport Systems Telecommunications (ITST)*, 20-22 October 2009, pp. 353-356