FireWebService Documentation

Holger Fritze[1], Thore Fechner[1], and Frederick Loock[1]

Institute for Geoinformatics (ifgi), University of Mnster, Germany
{h.fritze|thore.fechner|f.loock}@uni-muenster.de

February 15, 2010

1 Introduction

The European Organization for the Exploitation of Meteorological Satellites (EUMETSAT[1]) has established a series of geostational satellites for monitoring metereological phenomena. The METEOSAT[2] observes the region around the Prime Meridian with a resolution of 3 x 3 km and generates an image each 15 minutes. This allows near-real time information. One available product is the Active Fire Monitoring (FIR)[3]. This products displays information on the presence of wildfires.

Wildfires are any uncontrolled fires, that occur in the countryside [3]. As they are mostly caused by human and threaten whole ecosystems [2] a fast and accurate detection plays an important role in wildfire suppression.

Among others Meteosat-7 is only one of the GEONETCast data products. Geonetcast is "'a near real time, global network of satellite-based data dissemination systems designed to distribute space-based, air-borne and in situ data, metadata and products to diverse communities"' [1]. Current partners in that network are the National Oceanic and Atmospheric Administration (NOAA), the World Meteorological Organization (WMO), the Chinese Meteorological Administration (CMA) and the already above mentioned EUMETSAT.

The main goal of this project was to develop an user-friendly, easy accessible portal application that serves as interface displaying current and past wildfires sensed by the FIR. Additionally the user should be able to get further information on the fire, including metadata like coordinates, time stamp, uncertainty classification as well as derived data as for example country or related wikipedia articles. The user should also be able to access the data through standardized web-services. The portal application can be accessed via

`http://gnc-vs01.uni-muenster.de:1337/firewebservice/`

[1] www.eumetsat.int
[2] http://en.wikipedia.org/wiki/Meteosat
[3] http://www.eumetsat.int/Home/Main/Access_to_Data/Meteosat_Meteorological_Products/Product_List/SP_1145431848902?l=en

2 Time Schedule

When evaluating the time schedule at the end of the project two important deviations could be observed:

1. The Wikipedia API was easy to use at first sight. But when including that API into the portal application it revealed that a deeper understanding was crucial and the usage of other libraries was necessary. Of course this required more time which had to be applied to the time schedule.
2. All Geoserver related tasks turned out to be least planned. Several problems and obstacles changed the maintenance plan in the way, that the workload was not on a constant level, but workload peaks characterized this task.

Evaluating the time schedule showed further that the more precise tasks are planned the more difficult it is to estimate each step. This is especially true, if you work with new or not familiar software systems.

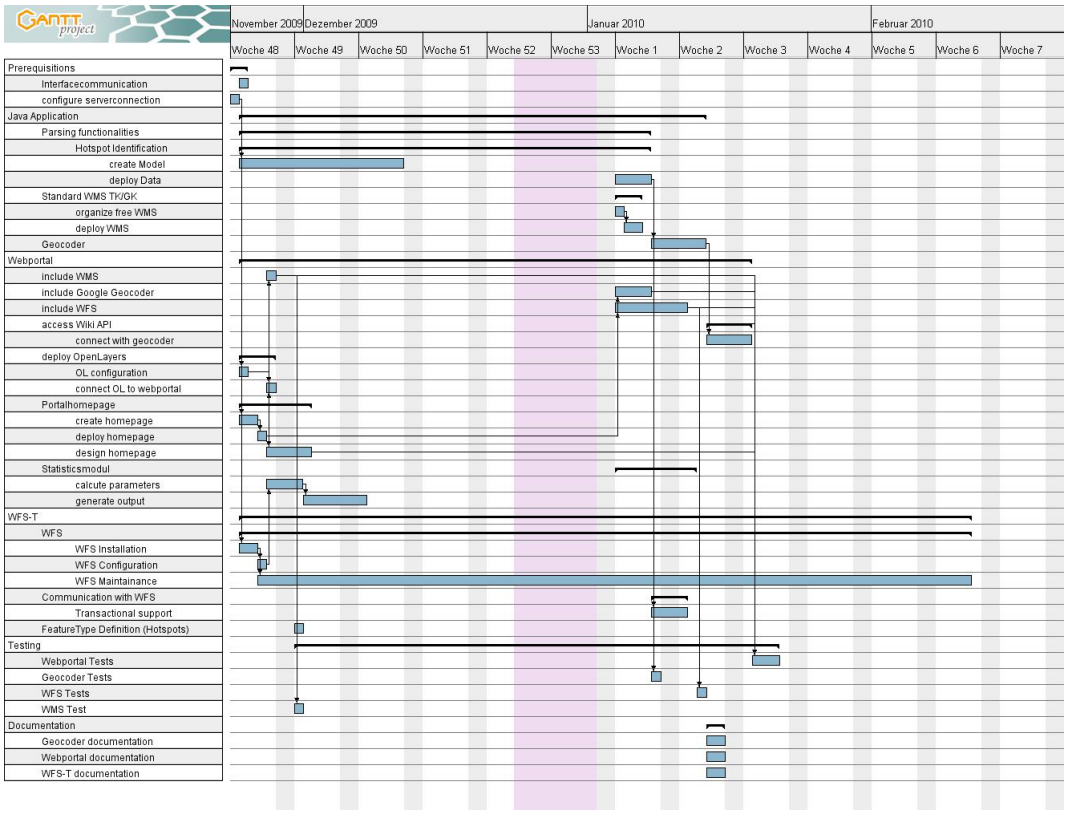

Fig. 1. Gantt Diagramm

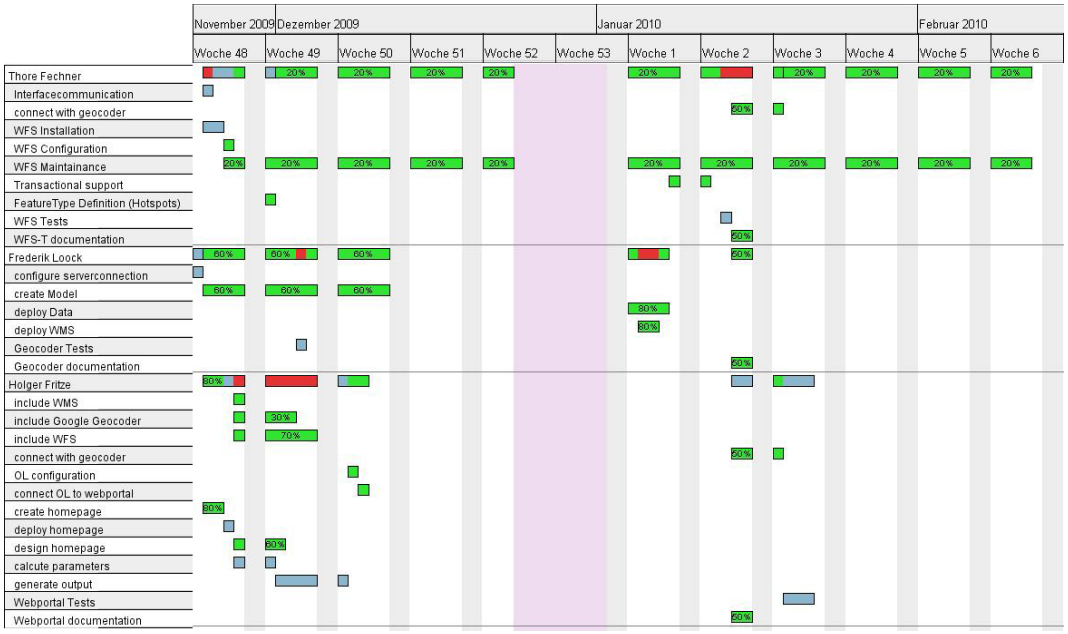

Fig. 2. Ressource Diagramm

3 Results

In the conceptualization and development phase of this project one of the focuses lied on the development of a stable and highly modular realization using standard web technologies. Additionally it was desired to use only open-source software to minimize the coast of this product and to allow further enhancement by following projects / groups.

Since the strong background in the geospatial and open source domain these goals could be achieved using the standards developed by the OpenGeospatial Consortium and various other organizations. The developed FireWebService is entirely based on open standards and software which are publicly available. The entire service is deployed on a Debian virtual server (Xenon 2.66 GHz / 512 MB Ram / 7 GB hdd) and due to its modular conceptualization it offers various standardized interfaces to access the processed data. The main building blocks of the service chain of the FireWebService are a small Java Application, which handles the parsing of the MODIS Fire Data, a WFS-T (Geoserver), a PostGIS database installed upon PostgresSQL, an Apache and Cronjobs which handle the scheduling of the service.

The results are published to the web via a portal homepage (see Figure 3) which allows an easy access to the data for presentation purposes. If the data is needed for further analysis it is possible to access it directly through the PostGIS database or the interface of the WFS.

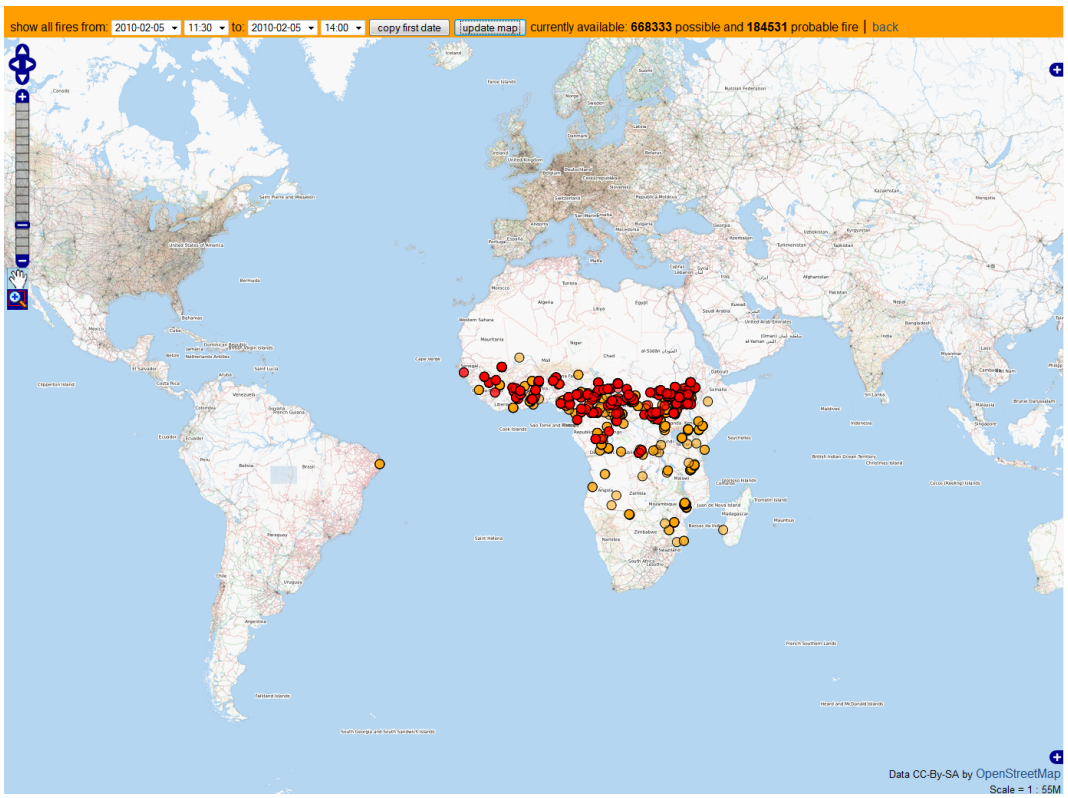

Fig. 3. Screenshot of the Portal Application map

4 Evaluation

The developed service shows how rather complex and sophisticated software architectures can be developed and deployed if proper standards are available and are used in the right way. The development of the FireWebService was only possible in the short amount of time (3 month) because those standards and implementations were available publicly and are widely used by large communities.

4.1 Teamwork / Organization

Organizing a team of software developers with different backgrounds, competences and time constraints is a task on its own. If the motivation is high everything went rather smooth, but as soon as major obstacles arose or certain unpleasant design decisions had to be made the development process can got stuck seriously. Teamwork and competence as well as mediator skills are required to achieve the common goal. Everybody expected that there would be some disagreements over certain aspects or that internal deadlines could be stretched a bit — what was not expected that there would be more then some. As mentioned in section 2 obstacles effect time schedules sometimes quite

tremendously Hence time buffers are essential for such unexpected problems. As rule of thumb 1/3 of the time amount available in total is a reasonable buffer!

Inspiring people for an idea is easy, maintaining that spirit is hard work. Still – if the goal is reached – the reward is worth it and everybody of us learned more then a bit about serious teamwork and motivation.

4.2 Technical Side

The developed architecture is highly flexible. At the moment it is deployed on a single server but it could be distributed easily. This is due to the fact that several independent services are used and tied together, each of them are capable to run on its own. The FireWebService is not a closed black box process, it is a transparent collection of services which gives the opportunity to access it a several levels according to the needs of the user. In addition it is possible to benefit from newly developed technologies, tie them in and enhance the overall product (e.g. a WebNotificationService).

4.3 Goals

It was possible to process the disseminated fire data and enhance it considerably. In its raw form the data is accessible via a plain ASCI file which is not really convenient in comparison to the massive amount of data points which are published.

The developed FireWebService processes this raw data and publishes it in a standardized way to the Internet. But it actually does much more then only "publish" the data. It gives a user the opportunity to analyze the data in several ways. Spatial, Logical and Comparison Operations along with the possibility to update existing datasets, insert new or delete old one build a strong framework for a lot of other applications. It is not limited to only show the process data it actually enhances it with community driven knowledge (Wikipedia Layer).

Most products are limited to serve one purpose and they do not allow to come up with new creative usages. In the case of the FireWebService the users are encouraged to take the published date and process it further and to come up with new ways to interpret or use it. The biggest value of this deployment is not the published data, it is the opportunity to do even more with it.

4.4 Conclusion

This project gave many lessons we had to learned during its realization. Teamwork, self organization, emphasizing ones skills, collaborative development, design patterns, documentation, Internet security and debugging to name only a

few of them. In total this was an inspiring but also very challenging project. It showed clearly that developing, programming and maintaining a project are totally different topics.

It takes a lot of effort to reach ambitious goals and even more if they should be presented to an audience in a appropriate way. It also requires much work to make a project stable and enduring. Finally this project showed us how we could improve an existing product and how could add value to it.

5 Future Work

This project was only a small sample of possibilities current technologies offer. During the last couple months a framework and an infrastructure for processing and publishing fire data were created and no further usage would be kind of waist of the invested work.

On the data side we have information only for the region around the prime meridian. Extending this area to a world wide coverage would require to tie in additional data sources. One example are given by the GEONETcast MODIS datasets.

Another topic this project has not touched at all so far is about analyzing on the data. Currently the user gets information about the existence and its fire certainty within a grid cell of an extent of 3 x 3 km, but at this point in time it is not possible to get know easily for example how big a fire is and how long it is already burning.

A next and technically more challenging idea that came up during the development of this project is to enable mobile devices like Android or Blackberry smartphones as well as the iPhone to use the portal application appropriately. Currently all this devices can access the homepage and display the map and the fires. But so far there is no intuitive interaction possible, which would include panning and zooming with your finger and no dependence on control keys. For more user-friendliness OpenLayers requires new interaction methods including multitouche.

The last point is to create a blog for this project, where all our experiences including strengths and weaknesses of each piece could be published as well as newly recognized bugs and developed work-arounds. Additionally this project or its components might be worth to publish them in the OWS catalogue.

References

1. GEO. Geonetcast. `http://www.earthobservations.org/GEONETCast.shtml`, 2010. [Online; accessed 14-February-2010].
2. Stephen J. Pyne. *Introduction to wildland fire: fire management in the United States / Stephen J. Pyne.* Wiley, New York :, 1984.
3. Wikipedia. Wildfire - wikipedia, the free encyclopedia. `http://en.wikipedia.org/wiki/Wildfire`, 2010. [Online; accessed 09-February-2010].