Geophysical Research Abstracts
Vol. 12, EGU2010-5887, 2010
EGU General Assembly 2010

GITEWS, an extensible and open integration platform for manifold sensor systems and processing components based on Sensor Web Enablement and the principles of Service Oriented Architectures

Rainer Haener, Joachim Waechter, Jens Fleischer, Stefan Herrnkind, and Herrmann Schwarting
Deutsches GeoForschungsZentrum - GFZ, Data Center, Potsdam, Germany (rainer.haener@gfz-potsdam.de, +49 331 2881703)

The German Indonesian Tsunami Early Warning System (GITEWS) is a multifaceted system consisting of various sensor types like seismometers, sea level sensors or GPS stations, and processing components, all with their own system behavior and proprietary data structure. To operate a warning chain, beginning from measurements scaling up to warning products, all components have to interact in a correct way, both syntactically and semantically. Designing the system great emphasis was laid on conformity to the Sensor Web Enablement (SWE) specification by the Open Geospatial Consortium (OGC). The technical infrastructure, the so called Tsunami Service Bus (TSB) follows the blueprint of Service Oriented Architectures (SOA). The TSB is an integration concept (SWE) where functionality (observe, task, notify, alert, and process) is grouped around business processes (Monitoring, Decision Support, Sensor Management) and packaged as interoperable services (SAS, SOS, SPS, WNS). The benefits of using a flexible architecture together with SWE lead to an open integration platform:
• accessing and controlling heterogeneous sensors in a uniform way (Functional Integration)
• assigns functionality to distinct services (Separation of Concerns)
• allows resilient relationship between systems (Loose Coupling)
• integrates services so that they can be accessed from everywhere (Location Transparency)
• enables infrastructures which integrate heterogeneous applications (Encapsulation)
• allows combination of services (Orchestration) and data exchange within business processes
Warning systems will evolve over time: New sensor types might be added, old sensors will be replaced and processing components will be improved. From a collection of few basic services it shall be possible to compose more complex functionality essential for specific warning systems. Given these requirements a flexible infrastructure is a prerequisite for sustainable systems and their architecture must be tailored for evolution. The use of well-known techniques and widely used open source software implementing industrial standards reduces the impact of service modifications allowing the evolution of a system as a whole. GITEWS implemented a solution to feed sensor raw data from any (remote) system into the infrastructure. Specific dispatchers enable plugging in sensor-type specific processing without changing the architecture. Client components don't need to be adjusted if new sensor-types or individuals are added to the system, because they access them via standardized services. One of the outstanding features of service-oriented architectures is the possibility to compose new services from existing ones. The so called orchestration, allows the definition of new warning processes which can be adapted easily to new requirements. This approach has following advantages:
• With implementing SWE it is possible to establish the "detection" and integration of sensors via the internet. Thus a system of systems combining early warning functionality at different levels of detail is feasible.
• Any institution could add both its own components as well as components from third parties if they are developed in conformance to SOA principles. In a federation an institution keeps the ownership of its data and decides which data are provided by a service and when.
• A system can be deployed at minor costs as a core for own development at any institution and thus enabling autonomous early warning- or monitoring systems.
The presentation covers both design and various instantiations (live demonstration) of the GITEWS architecture. Experiences concerning the design and complexity of SWE will be addressed in detail. A substantial amount of attention is laid on the techniques and methods of extending the architecture, adapting proprietary components to

SWE services and encoding, and their orchestration in high level workflows and processes. Furthermore the potential of the architecture concerning adaptive behavior, collaboration across boundaries and semantic interoperability will be addressed.