Geophysical Research Abstracts
Vol. 12, EGU2010-12027, 2010
EGU General Assembly 2010

An OGC/SWE Conformant Client to Manage Geospatial Data on the GRID

Antonio Esteves, Marco Caldas, Antonio Pina, and Alberto Proença

University of Minho, Engineering School, Department of Informatics, Braga, Portugal (e-Mail: {esteves, marcocaldas, pina, aproenca}@di.uminho.pt / Fax: +351253604471)

CROSS-Fire is a Portuguese NGI funded project that aims to develop a grid-based risk management decision support system for the Civil Protection. It uses forest fires as the main case study and FireStation (FS) as an application that simulates fire spread.

To provide FS with dynamic geospatial and meteorological data needed for computing the wind field (wind direction and speed) and Fire Weather Index (precipitation, temperature, and humidity) we are currently developing a Web-interface for Remote Weather Stations (RWS).

The implementation is based on a SOS server from 52North that manages the access to a database that conforms to SOS from OGC/SWE initiative. The Web-interface is being tested with a Davis VantagePro2 Weather Station (WS), which has a base module and is connected wirelessly to a sensor suit that includes: rain collector, temperature, humidity, anemometer, barometer, and wind sensors.

A simple SOS client retrieves the data from the RWS which is connect to a FTP server, performs the necessary processing and uses an insertObservation client operation to insert data in the SOS database. The algorithm includes (i) parsing the configuration file, (ii) parsing the meteorological data file retrieved from the FTP server, and (iii) parsing the answer from the SOS server and all the required SOS operations.

All the functionality of the SOS client is a Web Processing Service (WPS) algorithm that runs as a core service of the CROSS-Fire. The services provided by the WPS allows the access and management of most of the interaction among all the components of the architecture, include the GRID infra-structure, the SDI infra-structure and a web portal.

The algorithm implements a GetObservation class that provides the methods: getStations and getData. These methods allows WPS (i) to request, from the SOS, the list of WSs available on a certain spatial window, together with details about the sensors included on each station, and (ii) to retrieve observations from sensors belonging to a specific station included on the list returned by the getStations request.

To execute a getStations request, the WPS must specify the coordinates of the desired spatial window and the used Coordinate Reference System (CRS), using the EPSG standard notation. To execute a getData request, the WPS must indicate (i) the station(s) to be accessed, (ii) the initial and final dates that delimitate the request, and (iii) the sensors, belonging to the specified station(s), from where observations are requested. The algorithm enquires the SOS, waits for its answer and forwards the answer to the WPS.

To implement the GetObservation class, some other classes were developed: CreateDoc, GetHTTP, and ObservationOffering. CreateDoc is responsible for structuring the XML document containing the SOS request. GetHTTP deals with the HTTP communication between the SOS client and the RWS. Finally, ObservationOffering is just used to store temporarily the fields returned by a call of getStations method.

In the future, we plan to integrate the meteorological data from RWSs with other types of spatial data. We

consider to use the MODIS instrument, in the case of accessing satellite imaging. The data we are interested in is mainly land coverage (vegetation) and burned areas to validate the maps of vegetation used by FS and the results of fire spread simulations. The information will be provided as coverages by a WCPS service, an extension of the WCS.