
A Service Oriented Architecture for Highly Distributed and Data-Intensive Geospatial Grid Software Systems

Chris A. Mattmann¹, Robert Raskin¹, Daniel J. Crichton¹

¹NASA Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109, USA

The volume of geospatially referenced scientific data is growing by orders of magnitude worldwide due to factors such as: (1) growing sensitivity and specificity of scientific instruments acquiring higher resolution data from space; (2) availability of output from large simulation models, such as those that predict environmental conditions; (3) decreasing costs of disk storage and network bandwidth to warehouse data that are captured; and (4) the increase of pay-for-play “commodity” computing environments, such as Amazon’s EC2 compute cloud and S3 storage system. Geospatial information *services* are in demand to geolocate these large data volumes and integrate them with other geo-information services, [4].

Today’s scientists and decision makers must regularly operate in such high volume, computer-intensive data environments, and the (potential) multi-national dissemination of resultant data has spawned the rapid growth of “virtual organizations” and associated data grids. Virtual organizations are heterogeneous, distributed, cross-institutional hardware and software networks sharing organizational compute power and data storage resources in support of solving hugely complex scientific problems. Virtual organizations are enabled by the *grid* software architecture [1] and corresponding grid software technologies that implement the architecture.

At NASA’s Jet Propulsion Laboratory, we have constructed a suite of grid software services and a service-oriented architecture called Object Oriented Data Technology (OODT) [2]. We will restrict our focus in this work to OODT’s Catalog and Archive Service (CAS). The CAS allows for ingestion of data and metadata into underlying grid catalogs and data repositories. The CAS is itself a service oriented architecture, leveraging four core grid services: (1) file management, providing data cataloging, query retrieval, and data transfer; (2) grid workflow [5], providing orchestration of science programs and business tasks, modeling their data and control flow dependencies; (3) resource management, responsible for allocation of workflow tasks to underlying grid computing resources; and (4) crawling functionality for automatic data product ingestion.

We are involved in the construction of a suite of geospatial information services for the upcoming Orbiting Carbon Observatory (OCO) NASA mission (to launch in 2009) and other satellite observation studies of the Earth. One such service is the ground-based Fourier Transform Spectrometry (FTS) service. The FTS service allows the OCO science team to validate measurements of atmospheric CO₂ retrieved from ground-based FTS instruments against other measures of atmospheric content at the same spatial location

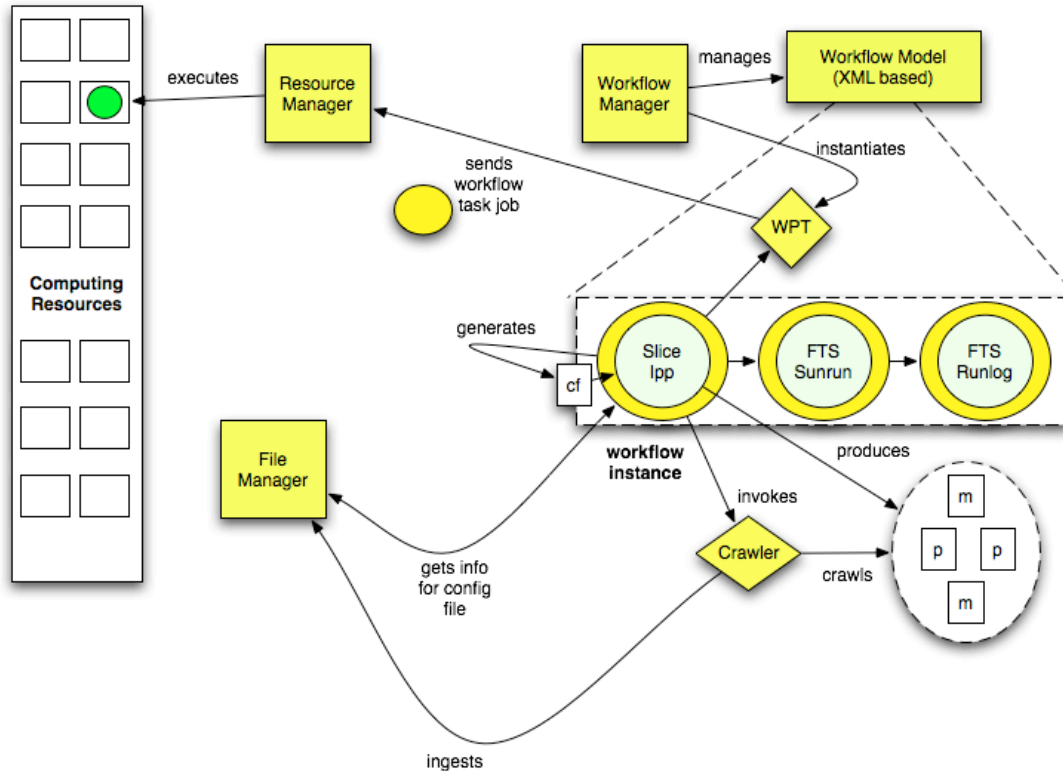


Figure 1. The OODT Science Data Processing Architecture

and time [3]. The FTS service represents a canonical geospatial service (as defined in ESRI's white paper [4]), encapsulating elements from *geo catalog services* that identify time and location-based raw instrument data and derived data products to stage to an FTS task, as well as *geo processing services* that identify and subset appropriate FTS spectrum for use in validation according to their location and data time.

The flow of the FTS service is demonstrated in middle right portion of Figure 1. We have directly leveraged our OODT CAS architecture and services in support of the software implementation of the FTS service. In Figure 1 *WPT* stands for *Workflow Processor Thread*, a CAS entity that orchestrates the execution of the 3 processing steps in the FTS service. *Cf* is a configuration file providing input to the processing program, generated by information agglomerated from the CAS services. *P* stands for output data *product*, and *M* for output *metadata*. The FTS information service collects raw instrument data (called Saveset Directories) from the OCO ground data system instrument catalog and runs them through the *SliceIpp* processing program to produce geolocated FTS spectrum, then generates ancillary products called FTS sunrun and FTS runlogs, which are produced by the corresponding *Sunrun* and *Runlog* processors.

Our future work with OODT's CAS services includes leveraging them to construct additional reusable geospatial and science data processing services and to make them available to the user communities. We are creating a Virtual Oceanographic Data Center that will leverage the OODT CAS technology and OODT's information integration grid services. This technology will enable provide transparent, easy-to-use access to a number of critical oceanographic data catalogs, including the EOS Clearinghouse (ECHO), as

well as the National Virtual Oceanographic Data System (NVODS). Closely coupled to this architecture is an ontology for representing geospatial and oceanographic concepts. The ontology provides systemwide semantic agreement on the meaning of the geospatial services offered. In addition to our core OODT services, we are investigating the integration of other geospatial services into our OODT architecture including OPeNDAP for data dissemination, as well as WMS services for dynamically selecting spatially referenced data.

References

1. C. Kesselman et al. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *J. Supercomputing Applications*, 2001.
2. C. Mattmann, D. Crichton, et al. A Software Architecture-based Framework for Highly Distributed and Data-Intensive Scientific Applications. *Proc. of ICSE*, 2006.
3. Fourier Transform Spectrometer (FTS) Delivery to Australia, <http://oco.jpl.nasa.gov/news/index.cfm?FuseAction=ShowNews&NewsID=6>, August 18, 2005.
4. Geospatial Service-Oriented Architecture (SOA), ESRI Whitepaper, available from <http://www.esri.com/library/whitepapers/pdfs/geospatial-soa.pdf>.
5. J. Yu and R. Buyya. A Taxonomy of Workflow Management Systems for Grid Computing. *J. Grid Computing*. 3(3-4): 171-200, 2005.