Registry Services

Distributed Infrastructure Design Team

May 31, 2009

Overview

This paper provides background information on the use of registries in science data systems. The Registry Service identifies resources available within a data system. Resources in PDS include web pages, software applications, searchable databases, online collections of data products, ftp servers and web services. In the current PDS environment many of these resources are accessible only by manually traversing the node web sites. Part of the goal of PDS 2010 is to make all of these resources more accessible to internal and external users or their software agents by using standard protocols to access the resources and documenting the protocols and resources via the registry service. The current PDS 2010 design includes a Registry Service, Inventory Service and Dictionary Service, all of which are forms of registries. The Inventory Service will provide access to metadata describing data sets and data products and the Dictionary Service will register metadata descriptions. We are envisioning the Registry Service as a component of a Service Oriented Architecture (SOA) where it provides a "yellow pages" directory of all web services. It is associated with a repository where artifacts (e.g. WSDL files, schemas, business process and information models) for the IT environment within an enterprise are actually stored. Commercial SOA registry/repository systems are essentially a specialized type of content or document management system. There is a strong emphasis on resource discovery, which allows internal or external customers to determine what services are available and how to use them; and governance, which includes access control, internal quality control and resource sharing. They may also include security, messaging and event notification capabilities.

Current practice in data system registries.

This section looks at the current registry architectures being used by the PDS, Global Change Master Directory (GCMD), International Virtual Observatory Alliance (IVOA), Space Physics Archive Search and Extract (SPASE) and Deep Space Mission System (DSMS), OGC OPENGIS, Open Grid Services Architecture (OGSA) and the Consultative Committee for Space Data Services (CCSDS).

The PDS has a central registry (high level catalog or "data search" database) which is used to select data sets based on a small set of global metadata parameters (e.g. target, instrument, data type). There are several JSP forms-based interfaces to what is called the “catalog profile server” which accesses the Sybase catalog database. A full-text search is also provided using Solr and Lucene to index the catalog database and a separate file that

describes the search tools at the discipline nodes. The catalog profile server is part of a distributed PDS-D registry/repository architecture in place at some of the discipline node sites that is implemented and maintained with Java-based Object Oriented Data Technology (OODT) tools. Within PDS-D a "profile server" provides a registry search and returns a set of XML profiles that match the search parameters and include a unique product id for selected products. A "product server" provides repository access by taking a unique product id or directory id and returning a product or set of packaged products. A "query server" can direct queries to distributed profile servers and return the aggregated results. The search syntax for all server types is called XMLQUERY. The central registry is maintained by a small staff at the engineering node with inputs from the discipline nodes. Access to the distributed registry/repository at the discipline node sites is via unique search applications at each discipline node (Image Atlas, Orbital Data Explorer, Rings Multi-Mission Search, Small Bodies Data Ferret). These applications generally access local databases with direct SQL queries and are not currently available for external use. There are also custom volume browsers in place at each discipline node that provide web-based access to the repositories. Other types of PDS resources (tools, documents, phone book) are maintained separately with different custom access mechanisms.

The Global Change Master Directory (GCMD) is a central registry of earth science data and services containing over 25,000 entries. Entries are created by filling out on-line forms using the "DocBuilder" software. A substantial team (10 people) is required to maintain the data base and web site. The Directory Interchange Format (DIF) is used to document data resources. There is a slightly different Service Entry Resource Format (SERF) to document tools to manipulate data. It does not seem to provide detailed service access parameters like a WSDL file would. There is also a special Ancilliary metadata form. The GCMD documentation mentions the capability to gather entries from US and international partners via OAI-PMH harvesting, or using XSLT to convert other registry metadata formats to DIF format. We could find no description of how this is actually implemented. GCMD is seeking to utilize SOA including a UDDI service registry and RDF to incorporate ontologies as well as developing semantic web technologies in future releases. Major resource categories in GCMD include Data Analysis and Visualization, Data Management /Data Handling, Education / Outreach, Environmental Advisories, Hazards Management, Metadata Handling and Models. Science keyword parameters use a hierarchical category, topic, term scheme where category is Earth Science and topics include Agriculture, Atmosphere, Biosphere, Biological Classification, Climate Indicators, Cryosphere, Human Dimensions, Land Surface, Oceans, Paleoclimate, Solid Earth, Spectral/Engineering, Sun-Earth Interactions, Terrestrial Hydrosphere. The GCMD registry capabilities are accessed via a web site and there is little documentation of the internal implementation.

The International Virtual Observatory Alliance (IVOA) has developed a federated registry system for describing resources available in the system. IVOA distinguishes between searchable registries that are intended to support client searches and publishing registries that support harvesting only. Large service providers (HESARC, STSCI) maintain searchable registries identifying their services. Smaller service providers might just register their services at one of the larger registry sites. IVOA registries tend to register services at the level of individual instrument data sets, so one site might register 50 cone

search services on different astronomical catalogs. There are about a dozen registry sites, but not all are active. The IVOA resource summary (and number of entries) is as follows: Simple Image Access (93+41+16+2), ConeSearch (70+3), VODataService (47), Simple Spectral Access (30), VORegistry (13), VOStandard, VOApplication, CEA, Space Time Coordinate, SkyNode (35), SkyService (11), TabularSkyService (13816), Authority (70), http-get (3), N/A (3), Organization (70), Other (4), Service (3), WebService (WSDL based) (5). The Registry interface specification supports an IVOA designed web services (wsdl/soap) interface, as well as the OAI-PMH harvesting interface using either a HTTP or SOAP protocol. Standard service type categories are browser-based (web page), CGI Get (URL with parameters) and SOAP-based (URL to access WSDL file). IVOA registry operations include: GetIdentity, GetResource, KeywordSearch, Search, TestGetResource, TestKeywords, TestSearch, XQuerySearch. OAI operations include: GetRecord, Identify, ListIdentifiers, ListMetadataFormats, ListRecords, ListSets, makeBadArg, makeBadVerb, makeMultipleErrors. Registry service metadata is intended to include enough detail to implement a programmatic interface to the resource. IVOA actively uses its registries to determine where to target searches with applications like DataScope. The Aladin application relies on a separate "yellow pages" capability called GLU which is synchronized weekly with the STSCI Registry.

The Space Physics Archive Search and Extract (SPASE) community has developed a registry and repository system for Space Physics metadata. The registry search interface utilizes SpaceQL which combines features of XQuery, DCOM and ADQL. SPASE resource types (and number of entries) include: Person (4,661), Observatory (826), Instrument (1,133), Catalog (1), Display Data (36), Numerical Data (1,324), Granule (184,170), Registry(1), Repository (73), Service (?). The SPASE web site provides a number of tools for building and validating registry entries. There is some code for building a local registry but I have been unable to get it to work. As far as I can tell the only access to the registry is via web pages. Only one service is registered in the current registry, the CDA ftp site. The documentation indicates that harvesting is used to populate the registry, but it is not clear on how this harvesting works.

The OPENGIS Catalog Service for Web provides a searchable registry that identifies individual server sites. Within OWS two application profiles have been developed for the catalog specification, one which uses the ISO-19115 (geographic metadata) and ISO 19119 (service metadata) standards and one which uses the OASIS ebXML Registry Information Model (ebRIM). There is also a Z39.50 protocol binding for the catalog specification that specifies the use of the geospatial metadata (GEO) and catalog interoperability protocol (CIP) profiles. The Universal Description, Discovery and Integration (UDDI) service registry was evaluated as a catalog server, but the architecture wasn't a good fit for supporting the metadata query requirements. The catalog service supports the following operations: GetCapabilities, DescribeRecords, GetRecords, GetRecordsById, GetDomain, Harvest, Transaction.

ECHO does not provide a registry service for data access. Instead it provides a collection search with metadata describing categories of data types. A query is implemented using the Catalog Service operation ExecuteQuery, which generates a result set. Other operations

include SaveQuery, SaveResultSets, GetQueryResults, ExecuteSavedQuery. Echo uses the Alternative Query Language (AQL) for the query specification. The ECHO catalog is build from periodic XML submissions by the ECHO data providers. ECHO plan to use a UDDI registry to access extended services but this capability has not been implemented yet. Access to all ECHO API's is through WSDL files which can be obtained from the ECHO web site. Echo service categories are Administration, Authentication, Catalog, Data Management, Event Notification, Extended Services, Group Management, Invocation, Order Management, Order Processing, Provider, Status, Subscription, Taxonomy and User. The ECHO metadata model is derived directly from that used by the Earth Observing System Data and Information System (EOSDIS) Core System (ECS) and conforms to the Federal Geographic Data Committee (FGDC) and Global Change Master Directory (GCMD) standards. The schema can be extended with product specific metadata. A query is implemented using the Catalog Service operation ExecuteQuery, which generates a result set. Other operations include SaveQuery, SaveResultSets, GetQueryResults, ExecuteSavedQuery. Echo uses the Alternative Query Language (AQL) for the query specification.

The Deep Space Mission System (DSMS) Information System Architecture registry is a "software service that provides registration, storage, and retrieval of critical data used in the development and operation of software systems". The registry includes several sub-registries for namespaces, data elements, xml files and services. DSMS is using a commercial product called IgniteXML.

OGF/GRID/GLOBUS. The latest OGSA document specifies that web services will be used including SOAP and WSDL. They are aligned with Organization for the Advancement of Structured Information Standards (OASIS), which supports both UDDI and ebXML. They use the term directory interchangeably with registry, but the documentation does not provide any specific information about registry implementations. There is a GLUE information model for grid entities, which includes a schema for service related metadata. SAGA is the simple API for grid applications. According to Wikipedia SAGA extensions will cover the following functions: service discovery, message exchange, storage of application level information, database access and integration, checkpoint management and recovery.

CCSDS. CCSDS has prepared a white paper on registries which is based on the OASIS ebXML Version 3 Registry Model and JAXR (Java for XML Registries) APIs. The focus is on registries/repositories for schemas, content and services. They are currently doing a pilot using the freebXML package (aka Omar). Here is a summary of text describing the ebXML query capability. "Query management deals with querying the registry for registry data. A simple business-level API, the BusinessQueryManager interface provides the ability to query for the most important high-level interfaces in the information model, such as Organizations, Services, ServiceBindings, ClassificationSchemes, and Concepts. Alternatively, the DeclarativeQueryManager interface provides a more flexible, generic API, enabling the JAXR client to perform ad hoc queries using a declarative query language syntax. Currently, the only declarative syntaxes JAXR supports are SQL-92 and OASIS/ebXML Registry Filter Queries. As noted in the JAXR specification, ebXML registry providers optionally support SQL queries. If a registry provider does support SQL queries,

the JAXR ebXML provider will throw an UnsupportedCapabilityException on DeclarativeQueryManage methods."

Standards and Commercial Products.

Two leading registry schemes are Universal Description, Discovery and Integration (UDDI) and Extended Business XML Registry Information Model (eb RIM). UDDI is intended to provide an electronic "yellow pages", or basic registry service for finding business oriented services. Interest in UDDI seems to be waning in the commercial sector and some UDDI registries have been shut-down.

The ebXML (aka exRIM, ebRS) provides a generalized registry and repository capability to support just about any sort of application. It provides support for security, content management, federated SQL and XML queries, user-defined metadata (classification, associations, taxonomies). It includes event notification protocol (implemented as registered subscription objects) to notify users or other registries about events of interest (new services, usage of services, changes to service content). The ebXML LifeCycleManagement Services include SubmitObjects, UpdateObjects, ApproveObjects, DeprecateObjects, UndeprecateObjects, RemoveObjects protocols. There is one QueryManagement service called AdHocQuery protocol, which supports SQL and "Filter Query" syntaxes and allows for stored queries. The AdHocQueryResponse contains the resulting RegistryObjectList and may be heterogenous. From a DSMS study, the key benefits of ebXML registry include: "Provides standard way to manage information assets; manages user-defined organization of and relationships among content and metadata; enforces user-defined standards for content, includes capabilities for managing and governance of information asset lifecycles; provides flexible mechanisms for content discovery; manages secure access to information assets; facilitates event-based delivery of information to appropriate personnel or systems; enables integration of information assets across organizational boundaries." API Summary. The ebXML api includes get, set, add, remove for RegistryObjects, Associations, Classifications, ExternalIdentifiers, ExternalLinks, AssociatedObjects and AuditTrails and SubmittingOrganization. It also includes getName, setName, getKey, SetKey, getLifeCycleManager, getObjectType, toXML operations.

Centrasite Community is an open source Registry with built-in UDDI repository. This product is oriented to centralized enterprise governance for internal development or b2b applications. Centrasite Resource Types include BPEL; BPELObject; CustomComponent; Documentation; DTD; E-mailEvent; Emerger; FileEvent; HTML; Icon; JAR; JMSEvent; Layout; Ontology; Payload; ProjectFolder; ReportDefinition; ScheduledTask; Sequence; SOAP; Template; TypeIcon; WSDD; WSDL; XML; XSD; XSLT.

WSO2 Registry. WSO2 provides an open source registry for SOA governance, part of a larger SOA software suite including web services application server, enterprise service bus, business process server, etc.