The Semantic Planetary Data System

PV-2005, "Ensuring Long-Term Preservation and Adding Value to Scientific and Technical Data"

21-23 November 2005, The Royal Society, Edinburgh, UK

J. Steven Hughes[(1)], Daniel Crichton[(1)], Sean Kelly[(1)], Chris Mattmann[(1))]

[(1)] Jet Propulsion Laboratory Pasadena
4800 Oak Grove Drive
Pasadena, CA 91109 USA
Email: {steve.hughes, dan.crichton
sean.kelly, chris.mattmann}@jpl.nasa.gov

INTRODUCTION

The Planetary Data System (PDS) data model was developed in the late 1980's to model the various entities and relationships of interest within the Planetary Science Community. It was developed to both prescribe the metadata to be collected for the planetary science data archive and to design the data set catalog, a high level inventory of the data holdings in the archive. The data model, implemented in a relational schema for the catalog database, supports sophisticated constraint-based searches for data sets based on their relationships to other modeled entities such as spacecraft instruments and target bodies that were involved in the collection of the data.

Since the advent of the Web, most of the information on the Web has been designed for human consumption using web technologies such as HTML, hyperlinks, and the http protocol. The Semantic Web now provides technologies to allow information to be easily read and consumed by computer software. These new technologies such as XML, the Resource Description Framework (RDF), and RDF Schema (RDFS) provide information that allows computer processing and reasoning of web information. This capability however is dependent on the existence of domain ontologies.

The Semantic PDS prototype demonstrates the use of semantic web technologies to capture, document, and manage the PDS data model and to provide intuitive facet- and text-based search for data holdings in the archive. The prototype makes use of the PDS Catalog, an inventory of over one thousand data sets and related entities. The underlying data model was ingested into an ontology tool and then exported as a Resource Description Framework (RDF) Schema (RDFS) file. The catalog data records were then written to an RDF file that conforms to the RDFS specifications. The files were then imported into a web-based semantic search engine allowing the search of PDS datasets and related entities via facet- and text-based search. This knowledge base also be made available to "semantically aware" software, allowing computers to process and reason about the information.

This paper will provide a brief overview of the PDS data model and the PDS Catalog. It will then describe the implementation of the Semantic PDS including the development of the formal ontology, the generation of RDFS/XML and RDF/XML data sets, and the building of the semantic search application.

THE PLANETARY SCIENCE DATA MODEL

The Planetary Data System (PDS) is the official science data archive for NASA's planetary science community. As such, it contains tens of terabytes of data collected from over thirty years of solar system exploration and will grow exponentially in the next few years. At its inception in the late 1980's, the PDS developed a data model, illustrated in Figure 1, that guides the capture of the information necessary to describe the data and ensure that the data remain scientifically useful for future scientists. Collected and validated using the data model, this information and the science data is submitted to peer review, archived, and distributed to the planetary science community. The data model was also used to design the data set catalog, a high level inventory of the data holdings in the archive. The data model, implemented in a relational schema for the catalog database, supports sophisticated constraint-based searches for data sets based on their relationships to other modeled entities such as spacecraft instruments and target bodies that were involved in the collection of the data.

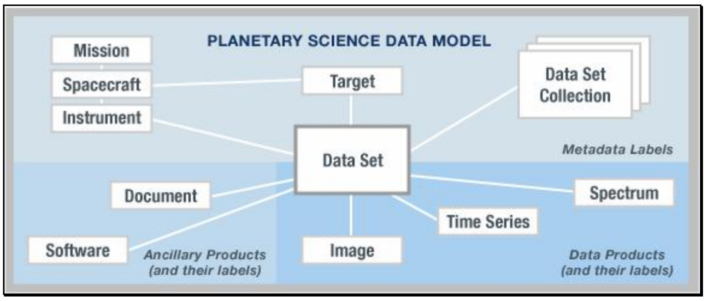
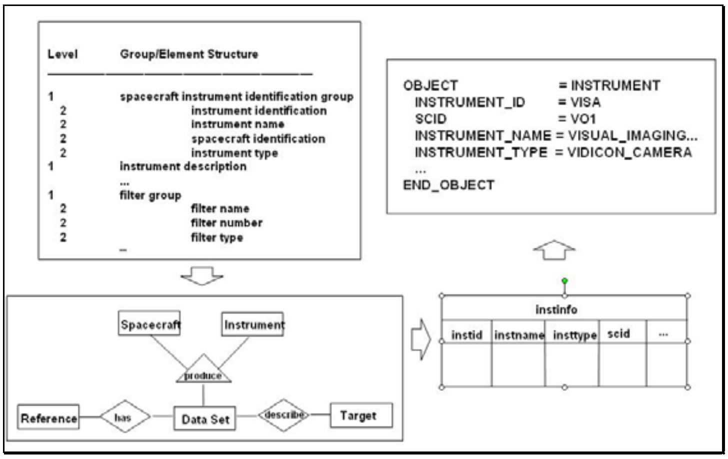


Figure 1 – The PDS Data Model

The development of the data model occurred over a period of about three years and included extensive interviews with planetary science domain experts by data management professionals. The data model was initially captured using a data dictionary and hierarchical structure diagrams, focusing on the description of planetary science entities, their attributes and relationships. The model centered on data sets (i.e. collections of data products) and a data set's relationships to other planetary science entities. Figure 2 shows the progression of the data model's development, from structure diagrams, through the Entity-Relationship model, and then implementation in a relational schema. Finally, in order to include the captured information on archive volumes as text files, the Object Description Language (ODL) was used.



```
Level       Group/Element Structure
___________________________________________
1           spacecraft instrument identification group
  2                     instrument identification
  2                     instrument name
  2                     spacecraft identification
  2                     instrument type
1           instrument description
            ...
1           filter group
  2                     filter name
  2                     filter number
  2                     filter type
            _
```

```
OBJECT                = INSTRUMENT
  INSTRUMENT_ID       = VISA
  SCID                = VO1
  INSTRUMENT_NAME = VISUAL_IMAGING...
  INSTRUMENT_TYPE  = VIDICON_CAMERA
  ...
END_OBJECT
```

instinfo

instid	instname	insttype	scid	...

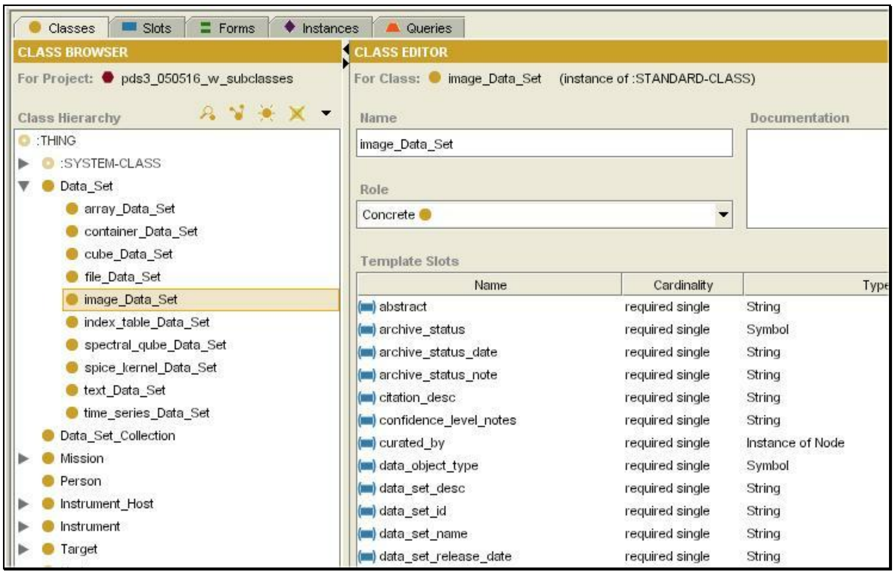
Figure 2 - Data Model Development

ONTOLOGY DEVELOPMENT

An ontology is the product of an attempt to formulate an exhaustive and rigorous conceptual schema about a domain. It is typically a hierarchical data structure containing all the relevant entities and their relationships and rules within that domain (e.g., a domain ontology). [8]

The development of the PDS ontology was relatively easy since the PDS catalog and schema contained the essential elements and defined the Planetary Science domain object classes, their attributes, and relationships. Stanford's ontology tool, Protégé was used to capture the object classes and their attributes from both the relational schema and the PDS data dictionary. Object relationships were then captured by analyzing foreign key constraints and SQL joins written for catalog applications. Since some modeling information is typically lost when implementing a relational schema, the initial interviews and structure chart documentation was also reviewed to refine the ontology. Figure 3 shows a portion of the resulting ontology, focusing on the Image Data Set class, its attributes (slots), and relationships. Useful modeling information such as subclass relationships, relationship cardinality, and whether an inverse relation exists are easy to see. To be consistent with Semantic Web trends the ontology was modified to include information architecture concepts from the Object-Oriented Data Technology (OODT) project. [1, 2, 4] These include broad-scope profile attributes and their relationships to support interoperability across domains. Finally, example instances of the ontology classes were ingested into the Protégé tool to validate the ontology. It should be noted that the entire PDS Catalog could be ingested into the Protégé tool, resulting in a PDS knowledge base. This would provide yet another alternative to the PDS catalog as a source for archive information.

.



● Classes | Slots | Forms | Instances | Queries

CLASS BROWSER

For Project: ● pds3_050516_w_subclasses

Class Hierarchy

- :THING
- ▶ :SYSTEM-CLASS
- ▼ Data_Set
  - array_Data_Set
  - container_Data_Set
  - cube_Data_Set
  - file_Data_Set
  - image_Data_Set
  - index_table_Data_Set
  - spectral_qube_Data_Set
  - spice_kernel_Data_Set
  - text_Data_Set
  - time_series_Data_Set
- Data_Set_Collection
- ▶ Mission
- Person
- ▶ Instrument_Host
- ▶ Instrument
- ▶ Target

CLASS EDITOR

For Class: ● image_Data_Set (instance of :STANDARD-CLASS)

Name: image_Data_Set

Documentation

Role: Concrete ●

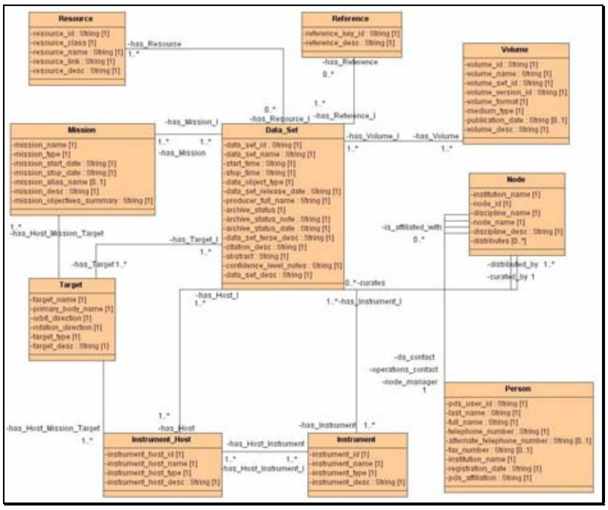
Template Slots

Name	Cardinality	Type
abstract	required single	String
archive_status	required single	Symbol
archive_status_date	required single	String
archive_status_note	required single	String
citation_desc	required single	String
confidence_level_notes	required single	String
curated_by	required single	Instance of Node
data_object_type	required single	Symbol
data_set_desc	required single	String
data_set_id	required single	String
data_set_name	required single	String
data_set_release_date	required single	String

Figure 3 - Data Set Class

ONTOLOGY REPRESENTATIONS

The capture of the data model as an ontology has resulted in a more formal and richer specification of the planetary science domain model. It reveals both known and unknown weaknesses in the model and provides alternate methods for analyzing and documenting the model. For example the Protégé tool provides several plug-ins for producing class diagrams and UML graphical representations as illustrated in Figure 4. Also since essentially all aspects of the data model have been captured in the ontology, the ontology becomes the master copy from which all other views of the data model can be extracted. For example, a relational schema can be built from the ontology.



Resource
-resource_id : String [1]
-resource_class [1]
-resource_name : String [1]
-resource_link : String [1]
-resource_desc : String [1]
-has_Resource
1..*
Reference
-reference_key_id : String [1]
-reference_desc : String [1]
-has_Reference
0..*
Volume
-volume_id : String [1]
-volume_name : String [1]
-volume_set_id : String [1]
-volume_version_id : String [1]
-volume_format [1]
-medium_type [1]
-publication_date : String [0..1]
-volume_desc : String [1]
0..*
-has_Resource_I
1..*
-has_Reference_I
-has_Mission_I
Mission
-mission_name [1]
-mission_type [1]
-mission_start_date : String [1]
-mission_stop_date : String [1]
-mission_alias_name [0..1]
-mission_desc : String [1]
-mission_objectives_summary : String [1]
1..*
1..*
-has_Mission
Data_Set
-data_set_id : String [1]
-data_set_name : String [1]
-start_time : String [1]
-stop_time : String [1]
-data_object_type [1]
-data_set_release_date : String [1]
-producer_full_name : String [1]
-archive_status [1]
-archive_status_note : String [1]
-archive_status_date : String [1]
-data_set_terse_desc : String [1]
-citation_desc : String [1]
-abstract : String [1]
-confidence_level_notes : String [1]
-data_set_desc : String [1]
-has_Volume_I
-has_Volume
1..*
1..*
Node
-institution_name [1]
-node_id [1]
-discipline_name [1]
-node_name [1]
-discipline_desc : String [1]
-distributes [0..*]
-is_affiliated_with
0..*
1..*
-has_Host_Mission_Target
-has_Target_I
1..*
-has_Target 1..*
-distributed_by 1..*
-curated_by 1
0..*-curates
Target
-target_name [1]
-primary_body_name [1]
-orbit_direction [1]
-rotation_direction [1]
-target_type [1]
-target_desc : String [1]
-has_Host_I
1..*
1..*-has_Instrument_I
-da_contact
-operations_contact
-node_manager
1
Person
-pds_user_id : String [1]
-last_name : String [1]
-full_name : String [1]
-telephone_number : String [1]
-alternate_telephone_number : String [0..1]
-fax_number : String [0..1]
-institution_name [1]
-registration_date : String [1]
-pds_affiliation : String [1]
1..*
-has_Host
-has_Host_Mission_Target
1..*
Instrument_Host
-instrument_host_id [1]
-instrument_host_name [1]
-instrument_host_type [1]
-instrument_host_desc : String [1]
-has_Host_Instrument
1..*
1..*
-has_Host_Instrument_I
-has_Instrument
1..*
Instrument
-instrument_id [1]
-instrument_name [1]
-instrument_type [1]
-instrument_desc : String [1]

Figure 4 - UML Class Diagram

RESOURCE DESCRIPTION FRAMEWORK (RDF)

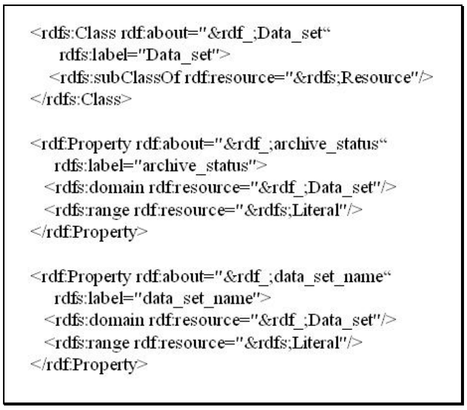
“The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation." [10] In contrast, early Web development focused on people collecting information and using HTML to present the information in an organized manner for human consumption. The resulting web pages could be easily navigated by people however computers had little understanding other than how to display the information based on HTML tag semantics and how to dereference hyperlinks. For example, a computer understands that an <H2> header should be displayed differently than an <H1> header but does not understand that when displayed the user observes a clear hierarchical relationship between the headers. Similarly, well designed hyperlinks can illustrate vivid semantic relationships to the user while the computer is limited to understanding simple links between information items.

The Resource Description Framework (RDF) is a language for representing information about resources in the World Wide Web. [9] RDF is based on the idea of identifying things using Web identifiers (called Uniform Resource Identifiers, or URIs), and describing resources in terms of simple properties and property values. This enables RDF to represent simple statements about resources as a graph of nodes and arcs representing the resources, and their properties and values.

The RDF specification provides an XML-based syntax (called RDF/XML) for recording and exchanging RDF graphs that can be processed by a computer. Referring to any identifiable thing, URI’s can access things that are accessible on the Web but importantly do not have to be. For example, within the planetary science domain, URI’s can identify image and spectrum data products that are available online as well as more conceptual things such as spacecraft or instruments that are simply described. In addition, RDF properties themselves have URIs to precisely identify the relationships that exist between the linked items. So RDF/XML provides a means for allowing a machine to process ontological information about the relationship between an instrument and an image data set.

RDF properties may be thought of as attributes of resources and in this sense correspond to traditional attribute-value pairs. RDF properties also represent relationships between resources. RDF however, provides no mechanisms for describing these properties, nor does it provide any mechanisms for describing the relationships between these properties and other resources. That is the role of the RDF vocabulary description language, RDF Schema. [11] RDF Schema defines classes and properties that may be used to describe classes, properties and other resources. The RDF Schema also has an XML-based syntax, RDFS/XML.

The Protégé tool allows the export of its contents to RDFS/XML and RDF/XML. Class definitions, their attributes, and relationships are written to a file in RDFS/XML format. Any object instances that have been ingested are written to a file in RDF/XML format. For this prototype, the PDS ontology was refined to focus on the subset of classes represented in the PDS Data Set View interface. These class definitions, attributes, and their relationships were exported to a file in RDFS/XML format as illustrated in Figure 5. As shown, the Data_set class is defined as a subClassOf a Resource and it has the properties archive_status and data_set_name. It should be noted that for this prototype, much of the relationship information is not expressed in the RDF schema, even though it was modeled in the ontology. The goal of this prototype was to demonstrate simple facet search which is accomplished by using relational foreign keys. Future work will include and use the relationships modeled in the ontology.



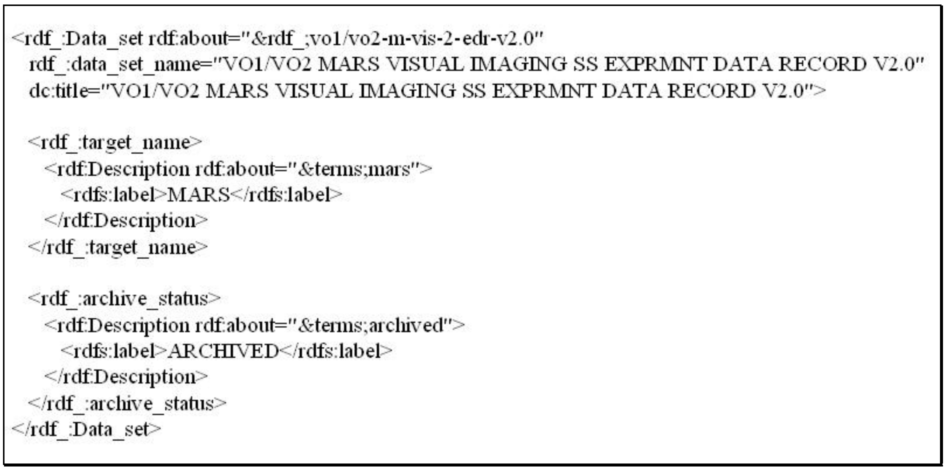
```
<rdfs:Class rdf:about="&rdf_;Data_set“
    rdfs:label="Data_set">
  <rdfs:subClassOf rdf:resource="&rdfs;Resource"/>
</rdfs:Class>

<rdf:Property rdf:about="&rdf_;archive_status“
   rdfs:label="archive_status">
  <rdfs:domain rdf:resource="&rdf_;Data_set"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>

<rdf:Property rdf:about="&rdf_;data_set_name“
   rdfs:label="data_set_name">
  <rdfs:domain rdf:resource="&rdf_;Data_set"/>
  <rdfs:range rdf:resource="&rdfs;Literal"/>
</rdf:Property>
```

Figure 5 - RDFS/XML for a PDS Data Set

SEMANTIC SEARCH

Several research efforts are using RDF/XML knowledge bases to provide semantic search capabilities. The SIMILE Project deals with applying semantic web technologies to digital libraries and providing the capability to browse and search arbitrary RDF datasets. It also supports different user interface scenarios useful to end-users, digital librarians, and metadata analysts. [12] The Simile/Longwell suite includes web-based RDF browsers that allows the user to browse and search arbitrarily complex RDF datasets using different styles including an end-user friendly view (where all the complexity of RDF is hidden) an RDF-aware view (where all the details are shown). For this prototype, the Simile/Longwell suite was chosen to provide facet- and text-based search. The text-based search is provided by Lucene. [13]

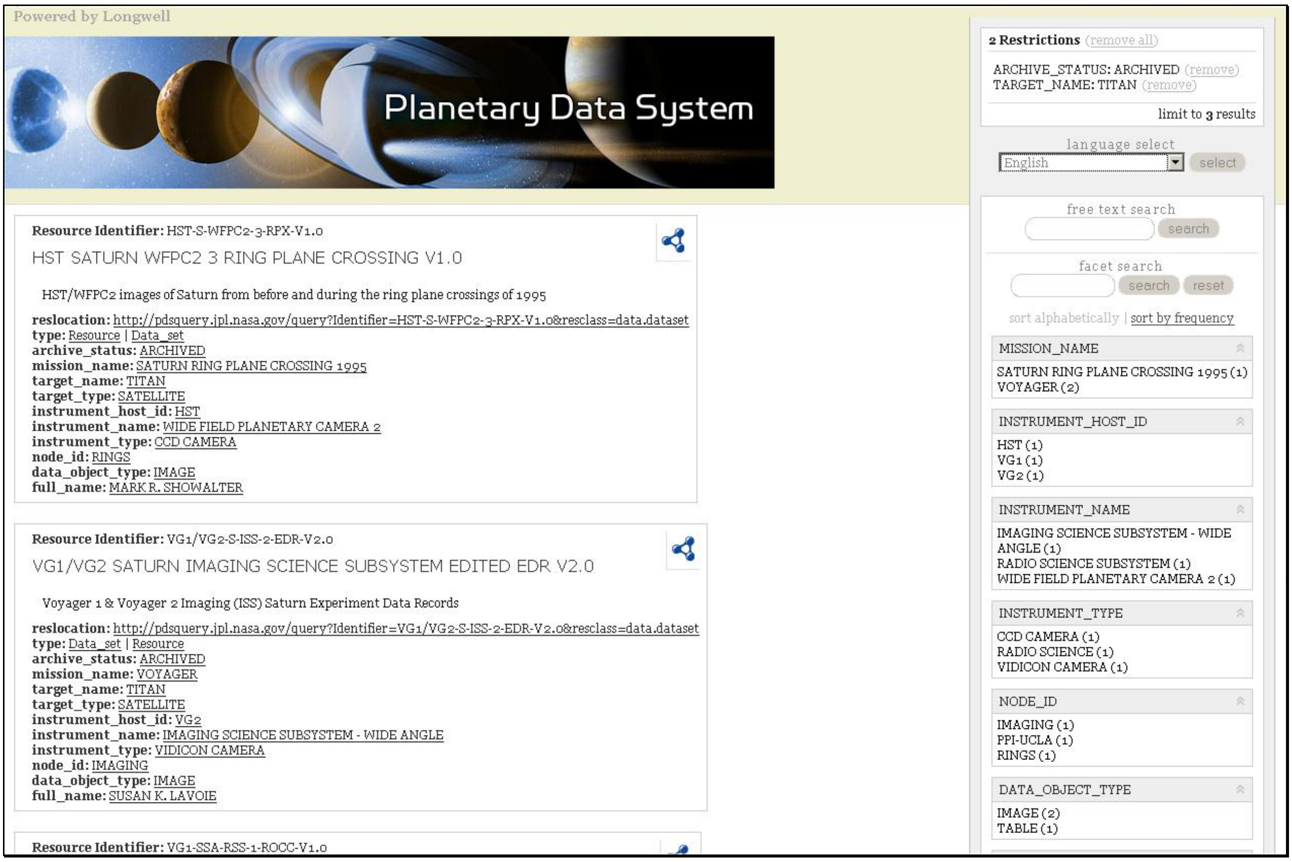
As previously mentioned, the PDS ontology was exported from Protégé into a RDFS/XML data set. The RDF/XML data set containing the data set, spacecraft, and instrument descriptions was created from the PDS Catalog database using a Java extract program. Figure 5 illustrates the a portion the RDF/XML describing a Viking data set and shows the data set name, the target body, and the status of the data set. Again notice that the relationship between the data set and target classes is represented using relational foreign keys.



```
<rdf_:Data_set rdf:about="&rdf_;vo1/vo2-m-vis-2-edr-v2.0"
  rdf_:data_set_name="VO1/VO2 MARS VISUAL IMAGING SS EXPRMNT DATA RECORD V2.0"
  dc:title="VO1/VO2 MARS VISUAL IMAGING SS EXPRMNT DATA RECORD V2.0">

  <rdf_:target_name>
    <rdf:Description rdf:about="&terms;mars">
      <rdfs:label>MARS</rdfs:label>
    </rdf:Description>
  </rdf_:target_name>

  <rdf_:archive_status>
    <rdf:Description rdf:about="&terms;archived">
      <rdfs:label>ARCHIVED</rdfs:label>
    </rdf:Description>
  </rdf_:archive_status>
</rdf_:Data_set>
```

Figure 6 - RDF for Viking Imaging Data Set

The build of the Longwell semantic search application is accomplished by specifying the RDFS/XML file as an ontology, the RDF/XML file as a data set, and the object attributes and values to be used as “facets” in a set of configurations files. The build process produces a .war file for deployment as web application. Figure 7 illustrates the resulting user interface where users restrict searches using an arbitrary combination of text input and facet selections. The query results are displayed as specified in the application build configuration files and is typically a subset of the information available from the RDF data sets. The complete RDF for each result can be viewed by clicking to the Knowle RDF navigator via the blue triangle. In the figure, two restrictions, archive_status=ARCHIVED and target_name=TITAN have resulted in three data set objects. The source RDF data set includes 1066 data sets and thousands of targets.



Resource Identifier: HST-S-WFPC2-3-RPX-V1.0

HST SATURN WFPC2 3 RING PLANE CROSSING V1.0

HST/WFPC2 images of Saturn from before and during the ring plane crossings of 1995

reslocation: http://pdsquery.jpl.nasa.gov/query?Identifier=HST-S-WFPC2-3-RPX-V1.0&resclass=data.dataset
type: Resource | Data_set
archive_status: ARCHIVED
mission_name: SATURN RING PLANE CROSSING 1995
target_name: TITAN
target_type: SATELLITE
instrument_host_id: HST
instrument_name: WIDE FIELD PLANETARY CAMERA 2
instrument_type: CCD CAMERA
node_id: RINGS
data_object_type: IMAGE
full_name: MARK R. SHOWALTER

Resource Identifier: VG1/VG2-S-ISS-2-EDR-V2.0

VG1/VG2 SATURN IMAGING SCIENCE SUBSYSTEM EDITED EDR V2.0

Voyager 1 & Voyager 2 Imaging (ISS) Saturn Experiment Data Records

reslocation: http://pdsquery.jpl.nasa.gov/query?Identifier=VG1/VG2-S-ISS-2-EDR-V2.0&resclass=data.dataset
type: Data_set | Resource
archive_status: ARCHIVED
mission_name: VOYAGER
target_name: TITAN
target_type: SATELLITE
instrument_host_id: VG2
instrument_name: IMAGING SCIENCE SUBSYSTEM - WIDE ANGLE
instrument_type: VIDICON CAMERA
node_id: IMAGING
data_object_type: IMAGE
full_name: SUSAN K. LAVOIE

Resource Identifier: VG1-SSA-RSS-1-ROCC-V1.0

2 Restrictions (remove all)

ARCHIVE_STATUS: ARCHIVED (remove)
TARGET_NAME: TITAN (remove)

limit to **3** results

MISSION_NAME
SATURN RING PLANE CROSSING 1995 (1)
VOYAGER (2)

INSTRUMENT_HOST_ID
HST (1)
VG1 (1)
VG2 (1)

INSTRUMENT_NAME
IMAGING SCIENCE SUBSYSTEM - WIDE ANGLE (1)
RADIO SCIENCE SUBSYSTEM (1)
WIDE FIELD PLANETARY CAMERA 2 (1)

INSTRUMENT_TYPE
CCD CAMERA (1)
RADIO SCIENCE (1)
VIDICON CAMERA (1)

NODE_ID
IMAGING (1)
PPI-UCLA (1)
RINGS (1)

DATA_OBJECT_TYPE
IMAGE (2)
TABLE (1)

Figure 7 – The Semantic PDS Search Interface

CONCLUSION

The Planetary Data System archives data for the planetary science community. Although the total data volume in the archive is not large relative to other science domains such as Earth Science, the planetary science domain is very complex, involving dynamic contexts within which the data is collected - orbiting target bodies, moving instrument platforms, and a plethora of reference frames. The early development of the PDS data model enabled the creation of a data archive consistent in its structure, meaning, and organization as well as rich in descriptive information. The advent of semantic web technologies provides a means for capitalizing on this knowledge base and thereby making the planetary science archive available to a wider range of customers in increasingly more intuitive and sophisticated ways.

Semantic Web technologies also suggest the means to support correlative science across science disciplines, missions, and instruments since they were designed to support inter-operability among digital assets. The Simile/Longwell suite for example allows a single knowledge base to be built using multiple and diverse ontologies and data sets. Large scale data system interoperability can be now be envisioned where “semantically aware” software agents reason about and process distributed science data repositories.

The Semantic PDS prototype demonstrates the ability to quickly develop facet- and text-based searches by leveraging existing domain catalogs. Even though designed for relational database technology, the resulting prototype demonstrates quick development, easy deployment, and functionality surpassing that available in traditional form-based database interfaces. The prototype also suggests the potential use of sematically aware software agents to assist scientists in gleaning existing space science archives.

REFERENCES

[1] Crichton D, Hughes JS, Hyon J, Kelly S., "Science Search and Retreval using XML". Proceedings of the 2nd National Conference on Scientific and Technical Data, National Academy of Science, Washington D.C, 2000.

[2] Kelly S, Crichton D, Hughes JS., "Deploying Object Oriented Data Technology to the Planetary Data System", Proceedings of the 34th Lunar and Planetary Science Conference 1607, 2003.

[4] Consultative Committee on Space Data Systems, "Space Information Architecture", White Paper, Information Architecture Working Group. February 2004, in press.

[6] ISO/IEC 1999. Framework for the Specification and Standardization of Data Elements 11179-1, Specification and Standardization of Data Elements 11179. International Organization For Standardization.

[8] Wikipedia, The Free Encyclopedia, "Ontology (computer science)", http://en.wikipedia.org/wiki/Ontology_%28computer_science%29, August 2005.

[9] RDF Primer, W3C Recommendation, http://www.w3.org/TR/rdf-primer/, 10 February 2004.

[10] The Semantic Web, Scientific American, Tim Berners-Lee, James Hendler, Ora Lassila, May 2001.

[11] RDF Vocabulary Description Language 1.0: RDF Schema W3C Recommendation 10 February 2004.

[12] The SIMILE Project, http://simile.mit.edu/longwell/index.html.

[13] Apache Lucene, The Apache Software Foundation, http://lucene.apache.org/java/docs/index.html