Scientific Digital Libraries, Interoperability, and Ontologies

J. Steven Hughes[1] Daniel J. Crichton[1] Chris A. Mattmann[1, 2]

[1]Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109, USA
{jshughes,crichton,mattmann}@jpl.nasa.gov

[2]Computer Science Department
University of Southern California
Los Angeles, CA 90089, USA
mattmann@usc.edu

ABSTRACT
Scientific digital libraries serve complex and evolving research communities. Justifications for the development of scientific digital libraries include the desire to preserve science data and the promises of information interconnectedness, correlative science, and system interoperability. Shared ontologies are fundamental to fulfilling these promises. We present a tool framework, some informal principles, and several case studies where shared ontologies are used to guide the implementation of scientific digital libraries. The tool framework, based on an ontology modeling tool, was configured to develop, manage, and keep shared ontologies relevant within changing domains and to promote the interoperability, interconnectedness, and correlation desired by scientists.

Categories and Subject Descriptors
H.3.7 [**Digital Libraries**]: Standards, Information Modeling

General Terms
Design, Standardization, Languages, Theory

Keywords
Digital Library, Ontology, Information Model, Interoperability, Science Data, Science Metadata.

1. INTRODUCTION
Scientific Digital Libraries are the key to advancing science through scientific collaboration. The advent of the Web and languages such as XML has brought an explosion of online science data repositories and the promises of correlated data and interoperable systems. However there have been relatively few real successes since research [1] suggests that just having physical and syntactic connectivity is not adequate. To achieve seamless connectivity between repositories, not only must the semantic issues be addressed, but important assumptions must be made about the ontologies being used to address the semantic issues. These assumptions include the need for a "single shared ontology" and the need for human assistance in the development of the ontology. Without these assumptions the effort to achieve seamless connectively across pre-existing repositories is essentially "cryptography", and rapidly becomes intractable. This paper will present a tool framework and some informal principles that have been used to develop shared ontologies for several scientific digital library projects.

JCDL'09, June 15–9, 2009, Austin, TX, USA.

2. BACKGROUND
The problem of bringing together heterogeneous and distributed information systems is known as the "interoperability problem" [2]. As recent research suggests physical and syntactic interoperability without semantic interoperability does not solve the general interoperability problem. To address semantic interoperability, Uschold [1, 3] suggests the use of shared ontologies and with the following assumptions.

1. All parties should use a single language for representing their ontologies.
2. All members in a given community should use:
    a. a single shared ontology, or
    b. a single shared upper ontology, with distinct domain ontologies, or
    c. a shared interlingua ontology to map individual ontologies to and from.
3. The semantic mapping among ontologies should be human-assisted, rather than fully automated.
4. The mapping will be done between lightweight ontologies, with a limited role for automated reasoning.
5. Adequate infrastructure support will exist for community repositories of both ontologies and interontology mappings.

Uschold and Gruninger [3] also suggest the following phases for the ontological engineering process.

1. Identify the purpose and scope including specialization, intended use, scenarios, set of terms including characteristics and granularity.
2. Build the ontology.
3. Evaluation: Verification and Validation.

In the following we present a case study where a shared ontology was developing under these assumptions and following the suggested phases.

3. THE PLANETARY DATA SYSTEM

The Planetary Data System (PDS) was developed to archive and distribute scientific data from NASA planetary missions, astronomical observations, and laboratory measurements. The PDS data standards [4] were developed in the late 1980's to define the concepts and terms needed for archiving science data in the planetary science domain. Even though the data standards were innovative [4-7] for their time, ambiguity and many assumptions have crept in over almost two decades of use and have caused significant problems for PDS operations, data providers, and end-users.

In 2008 the PDS formed a team to review the data standards and create an ontology [6]. The most reasonable interpretations of the data standards were captured in an ontology modeling tool and the identified anomalies were documented for future reference. The team configured a tool framework, based on the ontology modeling tool, to manage the ontology and produce specifications for developers, documentation for end-users, and exports of the ontology content for code generation.

The PDS data standards define and describe the data structures, data formats, and contextual information needed to make the science data useful to current and future planetary scientists. Some of the entities in the planetary science domain had been formally defined in the PDS data standards as simple classes, for example, spacecraft images and instruments. These definitions were migrated directly to the ontology. Often however, entities were simply described in the document narrative. In these cases, the most reasonable interpretations of the descriptions were used to define classes in the ontology.

The scope of the PDS data standards is one of the broadest in the space sciences. The data standards cover several planetary science sub-domains and associated communities of planetary scientists. Each community has their own domain of discourse but simultaneously desires collaboration with the other domains. The communities also share data types ranging from images to binary tables.

To validate the resulting ontology the system's functional requirements were referenced to identify the "things" that the implemented services and processes act on to perform their functions. These "things" are often either explicitly mentioned as nouns or implied in the requirements. The resulting list of "things" is considered to be the "information modeling" response to the system's functional requirements and validates that the ontology contains the classes needed to support system services and processes.

4. INFORMATION MODELING

The initial phase of the ontology engineering process used in developing the PDS ontology was not much different than that used for data modeling. However a significant difference is the role of metadata. In a data model the metadata is typically used to describe the structure and characteristics of the data for data processing. For example, a data model for a digital image is concerned with the data type of the image's pixel and the width and height of the image.

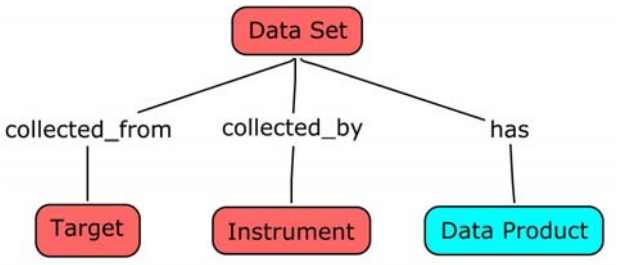

Figure 1. PDS Conceptual Model - Concept Map

In a scientific digital library however, especially those with requirements for long-term persistence and usability of the data, the metadata is on equal footing in significance to the data. For example, a digital image is essentially useless to a planetary scientist unless information about the locations of the light source, the imaging instrument, and the target body are all known in a single frame of reference. Also given that imaging observations during space flight are often non-repeatable, it is in the best interest of science to collect as much information as possible about the observation and the context within which it was performed. There are also requirements for rich classification schemes to enable searching within large volumes of data and to support correlative science. All this suggests that the metadata models should be developed with the same rigor as that applied to the data models.

A scientific digital library is a type of information system, therefore an information model is required. The information model will typically comprise several data models as well as models for physical and conceptual things in the domain. As mentioned earlier, the context within which data is collected is as important as the data itself. For example in addition to a digital image model, a model will also be needed to describe the physical instrument and the mission that is managing the project. A mission is an example of a conceptual thing.

5. INFORMATION MODELS

Information models that describe digital, physical, and conceptual entities are often complex, however these three general classes can be unified under the concept of the Open Archival Information System (OAIS) "Information Object" [8]. In general an information object is defined as comprised of a data object and its descriptive or "representation information". A "data object" can either be a digital object, a black box containing a sequence of bits, or a physical object that can be touched, for example a moon rock. The representation information contains the structural, semantic, and other information needed to understand and use the data object. The OAIS data object can be extended to add the conceptual object.

The PDS information model consists of classes that describe digital, physical, and conceptual things. In fact, even the PDS Data Object class (OAIS digital object) does not define but describes the class used to instantiate all the *actual* data in the digital library. The description indicates that the actual data is

simply a sequence of digital bits. To understand the Data Object, associations to other objects are needed to add meaning.

An information model is used throughout the development and operation of an information system. Also the users of the information model range broadly from software developers and project managers to users of the system. Few of these users understand the details of an information model, especially if it is captured in an ontology. The information model must be filtered and presented in notations that are suitable to the target audience. The Zachman Framework [9], a classification structure that is used in information technology development, defines several viewpoints and associated models that address this problem. In the following, two of these models as applied to the PDS are presented.

A conceptual model defines the community model of data from a manager's point of view and is concerned with the language of the community. For the PDS the ontology content was filtered and exported to produce a concept map. The individual concepts, or an ontology class with its attributes removed, are presented in the concept map as simple shapes with connecting lines representing named associations. Figure 1 provides a portion of the PDS concept map and illustrates several PDS concepts such as data set, instrument, and product.

A logical model defines the system model from a designer's point of view and is concerned with entity classes, attributes, and relationships. The logical model describes the things in the domain in rigorous terms. Figure 2 provides a portion of the logical model for two PDS classes, data set and instrument.

Other models in the Zachman hierarchy include the contextual model that provides a high level strategic view and other more detailed views associated with specific implementation choices. For example, the implementation of a model into a relational database system requires the logical model to be mapped to a relational physical model.

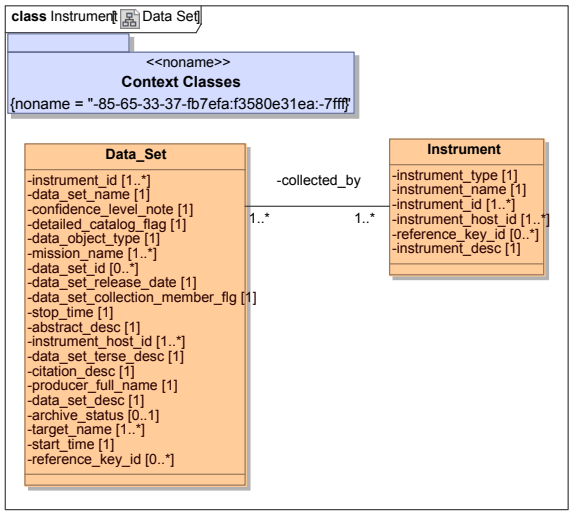

Figure 2. PDS Logical Model – UML Class Diagram

6. Principles

Wache [2] cites a "striking lack of sophisticated methodologies supporting the development and use of ontologies." The following principles used during development of the PDS ontology point to some guidelines for shared ontology development. Ongoing work is expected to help mature these guidelines.

6.1 Model Independence

The model should remain independent of its implementation. During the development of the PDS ontology, it was assumed that the ontology would remain independent of the target languages into which it was to be expressed. For example, XML is currently a popular language for implementation. However, a model for a domain that was not naturally hierarchical would be skewed if the hierarchical nature of XML/Schema were considered as a constraint during the modeling process.

Once an ontology is captured it can be filtered and mapped to less expressive languages. The special treatments needed to "shoe horn" an ontology into a target language are then located within the local implementation and subsequently are not propagated to other implementations or versions of the ontology. The PDS experience is that many requests for change are often negatively impacted by the limitations and quirks of the implementation language. The change process is therefore easier managed by considering the model and its implementation separately.

6.2 Model Driven

Developing scientific digital libraries for diverse and complex science domains such as the PDS poses two special challenges. First because the PDS supports a research community, the information model must keep pace with advancements in the science domain and periodic changes in geographical and political boundaries. Second, the technology used to implement the underlying information system will change at a different speed, typically faster, than the science domain.

As recommended earlier the ontology must remain independent of the implementation technology. A changing environment suggests that not only should the ontology guide the implementation of the information system but as much as possible it should drive the implementation so that a change in the ontology results in a change in the implementation. The PDS is planning to use the ontology to drive the implementation of its next generation system.

6.3 Semantic Richness

The ontology modeling language should be semantically richer than the other languages in the framework. This is suggested by the model independence principle since the ontology contents will typically be filtered and exported to less expressive languages. The contents of an existing ontology should also be migrated to richer ontology languages as needed.

6.4 Class Unification

The dichotomy in a scientific digital library between descriptions of "actual" data and descriptions of physical and conceptual

things in the domain can be unified under the OAIS Information Object. First, the *actual* data is defined as a set of digital objects, each a sequence of bits about which almost nothing is known and which was instantiated by some software application using a data structure. Since under the OAIS reference model the description of the data structure is considered representation information, it must be assumed that the data object was instantiated by a class that simply defined a sequence of bits. The model for the actual data then consists of information objects that are comprised of the class that describes the simple digital object and a class that describes the data structure. Information objects for physical and conceptual objects in the domain are comprised of representation information but typically no data object, at least not physically in the library.

6.5 Manage Change

Metadata management is a highly difficult problem that requires time and both domain and information modeling experts. For even the smallest domains the development of an information model can take years to complete since getting consensus on what something is, can be extremely arduous, especially in the science domains. Experience suggests that implementation independent ontologies make the development and management of information models much more efficient. The resulting model is subsequently used to support the collection, validation, and use of both the metadata and the data.

Once completed and in use, the ontology is simply a snapshot in time and will need to evolve with the domain. Significant effort will again be required to gain consensus on new concepts and integrate the results however the management of this change is again much more efficient in an implementation independent ontology.

6.6 Requirements

Four high level requirements for the information model were written for the information model development tasks. They are paraphrased below.

- The model shall consist of formal definitions for the relevant things in the science domain.
- The model shall provide the necessary specifications for data producers to design and generate data collections.
- The model shall provide the necessary specifications for software developers to write or generate compliant software.
- The model shall support interoperability between the data repositories in the community.

As mentioned earlier, the system's functional requirements were used to determine the things to be included in the model, in other words the descriptions of the things that are needed to support system services and tools. The resulting information model then acts as a set of requirements for data management software.

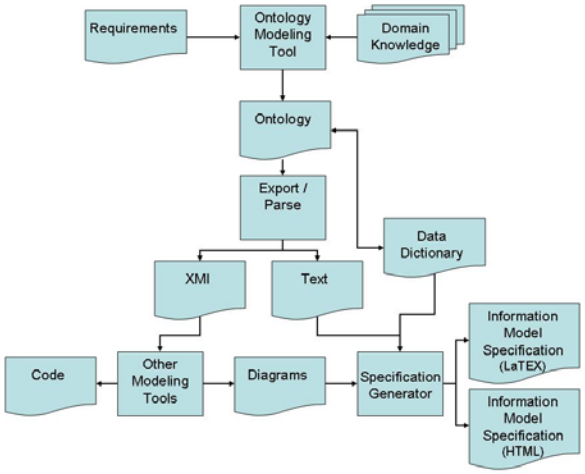

Figure 3. Tool Framework and Process Flow

7. TOOL FRAMEWORK

The tool framework is based on the Protégé ontology modeling tool [10]. The CMapTools Knowledge Modeling Kit [11] is used to generate visual depictions of conceptual models. The MagicDraw modeling tool [12] is used to generate UML models, class diagrams and code for defining and accessing Java classes. A Java application was written to generate specification documents that present several views of the information model using class definition tables, UML class diagrams, and concept maps. The specification also includes the data dictionary that describes the attributes in detail. The ontology content is both exported to XMI for use by the modeling tools and parsed by the Java application for the generation of the specification document. HTML and LaTeX versions of the specification document are generated. This framework and the process flow is illustrated in Figure 3.

Besides the PDS, the tool framework is being used to support information modeling tasks for several other projects including the International Planetary Data Alliance (IPDA), the Early Detection Research Network (EDRN) Knowledge Environment (EKE) [13], and the Consultative Committee for Space Data Systems (CCSDS) Registry Reference Model specification work.

The IPDA is a close association of international partners with the aim of improving the quality of planetary science data and services to the end users from space based instrumentation. In particular it seeks to improve interoperability between planetary science archives by developing common data and technology architectures. The IPDA adopted [14] the PDS information model as the de-facto data standard for the planetary science community and is now developing the technology architecture including a set of standard protocols.

The EDRN is a research network of collaborating scientists from over 40 institutions focused on identifying and validating cancer biomarkers (biological indicators of cancer) at their earliest stages. The EDRN Knowledge Environment (EKE) serves as an online, distributed resource of data and information that helps improve scientific research by enabling real-time access to cancer-research information that crosses institutional boundaries at a national level. The EDRN core ontology [15] defines this data and information.

The CCSDS is an organization of Space Agencies and produces recommendations and standards mainly for ground systems and their interface to space systems. The CCSDS Registry Reference Architecture includes an information model for a general purpose registry. This information model is being managed by the tool framework.

8. DATA DICTIONARY

Ontology modeling tools provide the means to capture a domain model in terms of classes and relationships. However a scientific digital library such as the PDS needs a rich set of data about class attributes and their values. For example the definition of an image pixel must include the data type of the value, the value's minimum and maximum bounds, whether the value is signed, and the order of the bytes. In addition during the design of a new image, data engineers will want to know what pixel definitions have been previously used that are similar in concept, who defined them, and who is allowed to make changes. A data dictionary is design to capture this type of information.

ISO/IEC 11179 2003 [16] is a specification for metadata registries and includes a comprehensive model for defining data elements. Figure 4 shows the basic model for describing a data element. The model separates the data element proper from its set of valid values. It also separates the conceptual view from the physical view, resulting in four distinct aspects of a data element. For example, the PDS data element, calibration_lamp_state_flag indicates whether the lamp used for onboard camera calibration is turned on or off. In the model, this data element is partitioned into 1a) the concept of a binary calibration lamp state indicator, 1b) the named data element "calibration_lamp_state_flag", 2a) the concept of a binary value, and 2b) the specific tokens used to indicate binary states, for example "on" and "off". The specification also allows for the classification and administration of each component of a data element. For example, the data element concept will have a version, last changed date, registration authority, submitter, and steward.

A data dictionary model that conforms to the ISO/IEC 11179 specification has been modeled and populated using the ontology modeling tool. It will be integrated into the tool framework.

9. RELATED WORK

Wache et. al. [2] summarize that reasonable results have been achieved on the technical side of using ontologies for intelligent information integration. The typical information integration system uses ontologies to explicate the contents of an information source, mainly by describing the intended meaning of table and datafield names. For this purpose, each information source is supplemented by an ontology which resembles and extends the structure of the information source. Noy [17] states that many issues that ontology researchers in semantic integration grapple with are very similar to the issues that database and information-integration researchers have been addressing. Some of the approaches are also similar although the ontology community relies more heavily on the higher expressive power of ontology languages and on reasoning techniques. Knublauch [18] also examine the benefit of using ontologies to support information modeling. Finally Singh et al. [19] suggest the need for ontologies to support the development of metadata catalogs for the sophisticated data-intensive applications resulting from advances in computational, storage and network technologies and data grid infrastructures.

10. ACKNOWLEDGMENTS

The authors wish to acknowledge the PDS Data Modeling for developing the original PDS data model and the PDS Technical staff who have performed heroically in attempting to keep the PDS data standards viable in the continually evolving planetary science domain. This work was supported by the Jet Propulsion Laboratory, managed by the California Institute of Technology under a contract with the National Aeronautics and Space Administration.

11. REFERENCES

[1] M. Uschold and G. M., "Ontologies and Semantics for Seamless Connectivity," *SIGMOD Record*, vol. 33, 2004.

[2] H. Wache, et al., "Ontology-Based Integration of Information — A Survey of Existing Approaches," In Proc. *IJCAI-01 Workshop:Ontologies and Information Sharing*, 2001.

[3] M. Uschold and M. Gruniger, "Ontologies: Principles, methods and applications," *Knowledge Engineering Review*, vol. 11, pp. 93-155, 1996.

[4] J. S. Hughes and S. K. McMahon, "The Planetary Data System. A Case Study in the Development and Management of Meta-Data for a Scientific Digital Library.," In Proc.

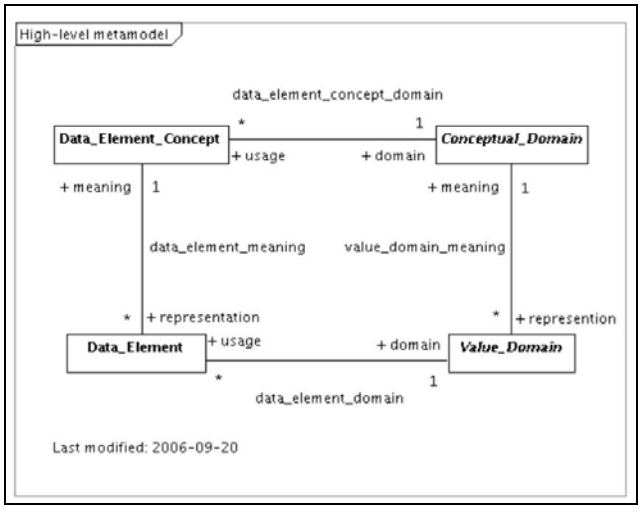

Figure 4. ISO/IEC 11179 - Data Element Model

ECDL, 1998.

[5] J. S. Hughes, et al., "A Planetary Data System for the 2006 Mars Reconnaisance Orbiter Era and Beyond," In Proc. *2nd ESA Symposium on Ensuring the Long Term Preservation and Adding Value to Scientific and Technical Data (PV-2004)*, Frascati, Italy, 2004.

[6] J. S. Hughes, et al., "An Ontology-Based Archive Information Model for the Planetary Science Community," In Proc. *Spaceops*, Heidelberg, Germany, 2008.

[7] S. Hughes, et al., "The Semantic Planetary Data System," In Proc. *3rd Symposium on Ensuring Long-term Preservation and Adding Value to Scientific and Technical Data*, The Royal Society, Edinburgh, UK, 2005.

[8] "Reference Model for an Open Archival Information System (OAIS)," CCSDS 650.0-B-1, 2002.

[9] J. A. Zachman, "A framework for information systems architecture," *IBM Syst. J.*, vol. 26, pp. 276-292, 1987.

[10] H. Eriksson and M. Musen, "Metatools for Knowledge Acquisition," *IEEE Softw.*, vol. 10, pp. 23-29, 1993.

[11] A. Cañas, et al., "Managing, Mapping, and Manipulating Conceptual Knowledge," In Proc. *AAAI-99 Workshop on Exploring Synergies of Knowledge Management and Case-Based Reasoning*, 1999.

[12] NoMagic, "Magic Draw, http://www.magicdraw.com/. ," 2009.

[13] D. Crichton, et al., "A Distributed Information Services Architecture to Support Biomarker Discovery in Early Detection of Cancer," In Proc. *2nd IEEE International Conference on e-Science and Grid Computing*, Amsterdam, the Netherlands, 2006.

[14] J. S. Hughes, et al., "Preliminary Definition of the Core Archive Data Standards of the International Planetary Data Alliance (IPDA)," In Proc. *PV 2007*, 2007.

[15] J. S. Hughes, et al., "An Information Model for Biomarker Research," In Proc. *5th EDRN Scientific Workshop*, Bethesda, MD, 2008.

[16] ISO/IEC, "ISO/IEC 11179: Information Technology -- Metadata registries (MDR), http://metadata-standards.org/11179/," 2008.

[17] N. Noy, "Semantic Integration: A Survey of Ontology Based Approaches," *SIGMOD Record*, vol. 33, pp. 65-70, 2004.

[18] H. Knublauch, "Ontology-Driven Software Development in the Context of the Semantic Web: An Example Scenario with Protege/OWL," in *International Workshop on the Model-Driven Semantic Web*. Monterey, CA, 2004.

[19] G. Singh, et al., "A Metadata Catalog Service for Data Intensive Applications," in *Proceedings of the 2003 ACM/IEEE conference on Supercomputing*: IEEE Computer Society, 2003, pp. 33.