

Smarter Earth Science Data System

Thomas Huang¹

¹*Jet Propulsion Laboratory, California Institute of Technology, 1200 E. California Boulevard, Pasadena, California, USA
thomas.huang@jpl.nasa.gov*

Keywords: semantic web:ontology:reasoning:search:archive:data center

Abstract: The explosive growth in Earth observational data in the recent decade demands a better method of interoperability across heterogeneous systems. The Earth science data system community has mastered the art in storing large volume of observational data, but it is still unclear how this traditional method scale over time as we are entering the age of Big Data. Indexed search solutions such as Apache Solr (Smiley and Pugh, 2011) provides fast, scalable search via keyword or phrases without any reasoning or inference. The modern search solutions such as Googles Knowledge Graph (Singhal, 2012) and Microsoft Bing, all utilize semantic reasoning to improve its accuracy in searches. The Earth science user community is demanding for an intelligent solution to help them finding the right data for their researches. The Ontological System for Context Artifacts and Resources (OSCAR) (Huang et al., 2012), was created in response to the DARPA Adaptive Vehicle Make (AVM) programs need for an intelligent context models management system to empower its terrain simulation subsystem. The core component of OSCAR is the Environmental Context Ontology (ECO) is built using the Semantic Web for Earth and Environmental Terminology (SWEET) (Raskin and Pan, 2005). This paper presents the current data archival methodology within a NASA Earth science data centers and discuss using semantic web to improve the way we capture and serve data to our users.

1 INTRODUCTION

The Earth science community is facing a big data problem and it is getting even bigger as we deploy more high-resolution instruments that produce large datasets in rapid rate. The NASA Earth Science Data and Information System (ESDIS) Project manages the science systems of the NASA Earth Observing System Data and Information System (EOSDIS). EOSDIS is a data centric system designed for the processing and archiving from NASAs Earth Observation missions and their distribution as well as provision of specialized services to users. The major components of EOSDIS are 12 Distributed Active Archive Centers (DAACs), 14 Science Investigator-led Processing Systems (SIPs), and the EOS Clearing House (ECHO). The DAACs play an import role within EOSDIS. They are divided into by discipline with User Working Groups tailored to mission and objectives of the DAAC. Unlike the traditional data center, DAACs includes these key tasks

- Support operational ingestion and management of a suite of space-borne sensors
- Produce data products from remotely sensed and complementary in situ data sets as required

- Reprocessing data in response of improvements in the algorithms or to correct errors detected in the processing

To date, the EOSDIS data holding is approximately 7.4PB with a 5.4TB average daily growth (Huang, 2013). These are heterogeneous observational and improved science data products. They can be categorized by their science parameters, level of processing, source platforms and sensors, as well as in spatiotemporal coverage.

1.1 Open Archival Information System (OAIS) Reference Model

Most of the NASA Earth science DAACs follow The Consultative Committee for Space Data Systems (CCSDS)s Open Archival Information System (OAIS) reference model (Figure [1]) in building its data archival systems. The OAIS reference model defines the core functions of an archival system and how information should be exchanged. In particular, there are three key messages types, the Submission Information Package (SIP), the Archival Information Package (AIP), and the Dissemination Information Package (DIP).

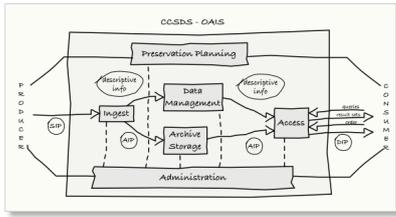


Figure 1: The CCSDS Open Archival Information System reference model

1.2 NASA's Physical Oceanographic Distributed Active Archive Center (PO.DAAC)

The NASA Physical Oceanographic Distributed Active Archive Center (PO.DAAC) is located at the Jet Propulsion Laboratory, Pasadena, California, and is responsible for archiving and distributing data relevant to the physical state of the ocean. PO.DAAC currently has over 90TB compressed data in its holding with an 10TB annual growth rate. The PO.DAAC core data management is call the Data Management and Archive System (DMAS) (Huang et al., 2009) (Figure [2]). DMAS is a highly scalable data system, designed from the ground up to be able to handle parallel ingestion of data from data providers across the globe. The entire DMAS system is able to scale down to operate on a single machine or scale up to a very large number of ingest/archive engines on an elastic cloud environment.

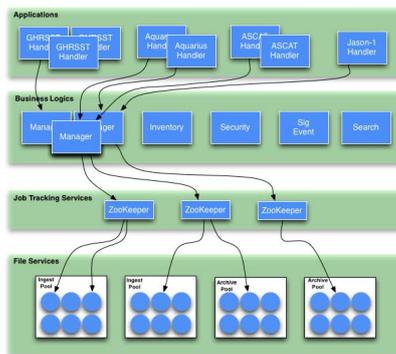


Figure 2: The PO.DAAC Data Management and Archive System architecture

A system such as DMAS could comfortably handle bursty archival workloads. The approach it has taken capturing heterogeneous oceanographic data into its system is by implementing dataset-specific handlers know as Data Handlers. The Data Handler performs several important functions (See section [5.1]) including harvesting artifact metadata and

translating the metadata according to the internal DMAS taxonomy.

Another challenge has always been helping the user finding the right datasets for their research. The current data tagging approach of manually associating keywords with each datasets has several limitations. First of all, the tag used must include industry standard terms as well as common terms, which is always incomplete. Second of all, search by keyword is a string comparison operation without getting into the meaning behind the search. Often the users will have to guess which keywords to search for in order to obtain larger result sets to work with. The current solution requires users to be trained by the machine rather than having the machine apply reasoning and inference rules to yield more accurate result sets.

2 SEMANTIC WEB FOR EARTH AND ENVIRONMENTAL TERMINOLOGY (SWEET) ONTOLOGIES

The Semantic Web for Earth and Environmental Terminology (SWEET) ontologies (<http://sweet.jpl.nasa.gov>) are widely adapted by the Earth Science community as the de facto ontologies for modeling earth environment. It was designed for Earth science data discovery through software understanding of the semantics of data artifacts. Implemented using the web ontology language (OWL), the collection of SWEET ontologies include both orthogonal concepts (space, time, Earth realms, physical quantities, etc.) and integrative science knowledge concepts (phenomena, events, etc.) The ontologies were initially developed to capture the relationships between keywords defined by the Global Change Master Directory (GCMD) (<http://gcmd.jpl.nasa.gov>). The SWEET ontologies enable scalable classification of Earth system concepts and have expanded to support space science in recent years. The current release, SWEET 2.3, is highly modular with over 6000 concepts and 200 separate ontologies. Designed as a high-level ontology (Figure[3]), SWEET promotes reuse by allowing Earth science domains to create specialized ontologies:

- **Import:** import only the ontologies required
- **Expand:** introduce new concepts and attributes specific to the application domain
- **Specialize:** extend and specialize SWEET concepts according to the application domain.

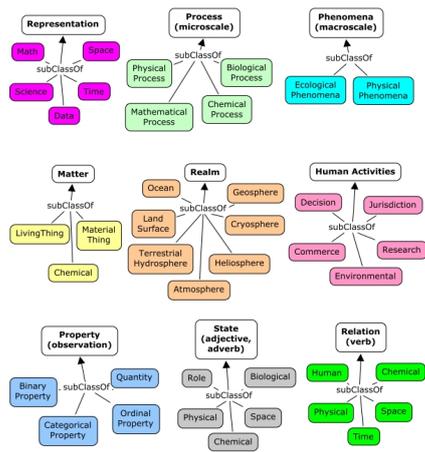


Figure 3: The SWEET Ontologies

Some of the users/applications of SWEET include Linked Environments for Atmospheric Discovery (LEAD) (Droegemeier et al., 2005), Ontology based Scoped Search Engine and Resource Aggregator for Atmospheric Science (Ramachandran et al., 2006), Semantically-Enabled Scientific Data Integration (Fox et al., 2007), Earth Science Information Partner Federation (ESIP), and Global Environmental & Earth Science Information System (GENESIS).

3 ENVIRONMENTAL CONTEXT ONTOLOGY (ECO)

The Environment Context Ontology (ECO), implemented using the Web Ontology Language (OWL), is a specialized ontology derived from the SWEET ontologies that comprise the knowledge engine for the DARPA AVM program. It captures the domain-specific concepts to define relationships among Earth environmental concepts. As the brain for the OSCAR archival system, the ontology also includes data management characteristics, data provenance, security, data center, and functional requirements. The environmental context portion of ECO can be divided into three top-level categories, atmospheric, aquatic, and land. The concept graphs capture the top-level concepts OSCAR needs to support.

The **Atmospheric** context models (Figure [4]) span thermodynamic and particulate influencing factors. The models provide climactic, ambient state information. The ontology organizes the environmental factors in groups distinguished by source and effect, and level of granularity. The air properties ontology encapsulates basic thermodynamic air properties such as temperature and pressure, which could be simulated under a controlled environment. At-

mospheric features include primary weather features such as thermal radiation and wind. The contaminants branch includes solids and liquids in suspension in the atmosphere, such as precipitation and particulates.

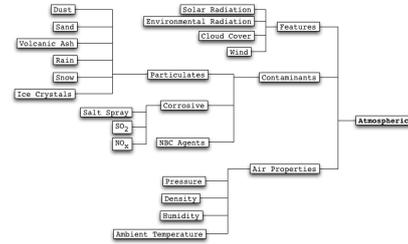


Figure 4: Atmospheric Concept Graph

The **Aquatic** context models (Figure [5]) organize amphibious realms in terms of intensive water properties, and water motion. Sea states can describe water motion in a more general fashion, and specifying the water character can provide further detail. Some of the sample data used for modeling wave height came from the Jason-1 satellite acquired through the NASA Physical Oceanographic Distributed Active Archive Center.

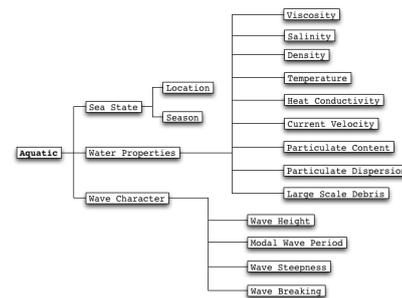


Figure 5: Aquatic Concept Graph

The **Land** context models (Figure [6]) focus on vehicle's suspension systems depending on mobility systems response, which can further be divided into continuums and obstacles. The continuum is concerned with concepts such as surface roughness and slope. The obstacles concept can be further divided into natural and man-made.

3.1 Information Linking using ECO

Data artifacts are linked (Figure[7]) through environmental context and through inference. When artifacts are archived into OSCAR, such as a context model, OSCAR can automatically apply reasoning through the ECO ontology to link the artifacts to the related functional requirements and data center or test course metadata.

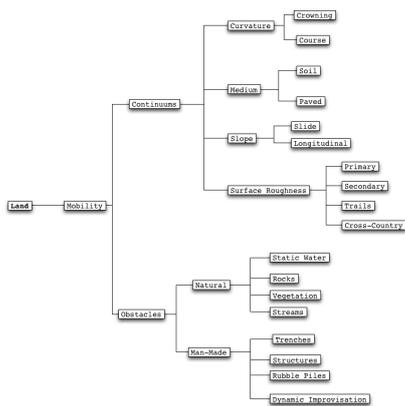


Figure 6: Land Concept Graph

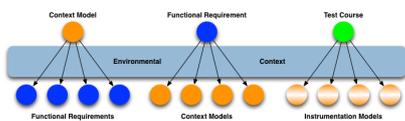


Figure 7: Linking artifacts through environmental context

There are two kinds of context models (Figure[8]), Binary Models and Instrumentation Model.

Binary Models include executable or runtime libraries. These models are implemented to execute on a specific platform and generate new model data through synthesis. These models typically take one or more Instrumentation Models as input. Environmental context is also used here to link Binary models with Instrumentation models to guide the user to obtain the correct inputs to use. When the user finds a Binary model according to a specific requirement, the list of Instrumentation model data is automatically provided.

Instrumentation Models are typically data gathered at test courses or data centers. These data might be packaged in provider-specific format such as CSV, NetCDF, XML, etc. The standard input interface to OSCAR is JSON. Some data massaging and conversion could be required before delivering data into OSCAR for archival.

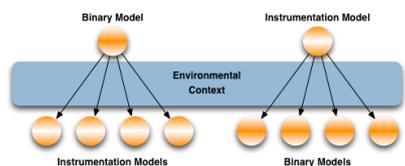


Figure 8: Cross linking between Binary Models and Instrumentation Models

4 ONTOLOGICAL SYSTEM FOR CONTEXT ARTIFACTS AND RESOURCES (OSCAR)

The Ontological System for Context Artifacts and Resources is a semantic web-based data system. It is an intelligent data system that links data artifacts according to environmental context. The diagram (Figure[9]) captures the high level system interfaces.

Model Builders: The model builders are special users of the system. They create and register models into OSCAR. They also serve as the data curator as they have the responsibility to maintain the artifacts they delivered to OSCAR.

Model Users: The model users typically use the OSCARs semantic search capability to discover the artifacts they need. The data models can be downloaded in various package formats to support the users local operating environment.

Simulation Environment: The simulation environment is the vehicle testbench. It is an environment to synthesize the vehicle being designed. This automated environment interacts with OSCAR through its RESTful interface. It queries and downloads models automatically. Typically the simulation environment searches OSCAR according to the predefined functional requirements.

Remote Data: These are data providers or data centers. Data gathered and/or distributed through these centers are ingesting into OSCAR for long-term archive and distribution. Often data produced at these centers need to be converted to JSON prior to ingestion.

Context Model Repository: The context model repository really consists of two major components, a database for metadata and a file system for file storage. As a semantic web-based system the database used is an RDF graph database. OSCAR is designed to support remote file storage, that is, the RDF graph database store the URL location of the data files. The URL can be local or remote.

4.1 OSCAR System Architecture

The Ontological System for Context Artifacts and Resources (OSCAR) is designed from the ground up in accordance to the RESTful service architecture. It promotes abstraction and separation of concerns, by considering objects within the system as resources and each object has a set of operations. The key components of OSCAR includes

- Framework for handling RDF/OWL data including reasoning.

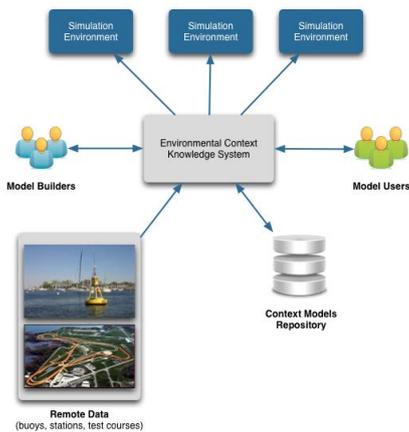


Figure 9: OSCAR Top-Level System Architecture

- Repository for reliable storage and retrieval of data.
- Fast indexed search.
- Security architecture to support authentication, authorization, and communication.

JavaTM is used for the implementation of OSCAR (Figure[10]), which maximizes OSCAR's portability. Over the years, Free and Open Source Software (FOSS) has proven itself to be the winning card for developing any software system. Examples of successful FOSS projects include Linux, Apache Software Foundation, Eclipse Foundation, Android Open Source Project, Perl, PHP, Python, etc. OSCAR is designed and developed to leverage from industry-standard open source components, which frees our sponsor from any long term licensing and vendor lock-in.

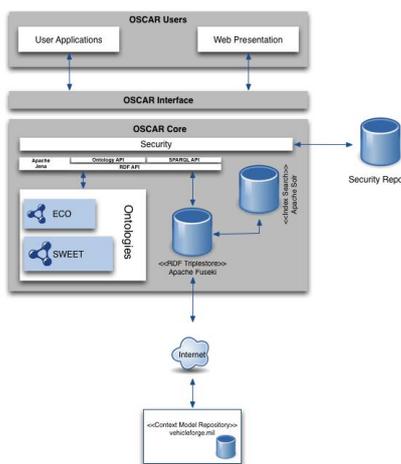


Figure 10: OSCAR System Architecture

OSCAR is designed with the following goals:

- A semantic web-based system
- Leverage successful JPL data service technologies
- RESTful service architecture
- Using SWEET ontologies as the upper ontology and specialized to support terrain modeling.
- Archival and distribution - Create, Read, Update, and Delete (CRUD) context model artifacts
- Semantic search and discovery. With integration with its reasoning subsystem and fast indexed search capability, OSCAR can deliver more accurate search results to the users.
- Auto linking between artifacts and resources
- Pluggable backend RDF graph database
- Pluggable security model. The system currently supports authentication and authorization through LDAP or custom internal database.

The capability delivered to DARPA is an operational semantic web application (Figure [11]). A model builder only needs to provide the minimal required information specific to their model, and OSCAR will automatically link the model with requirements and data center information. It provides a semantic search capability with concept inference.

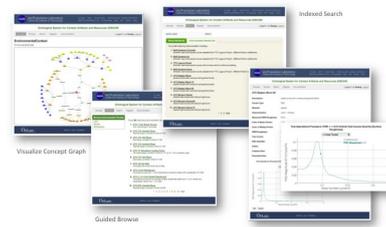


Figure 11: OSCAR Screenshots

5 EARTH SCIENCE DATA SYSTEMS

The modern Earth science data center provides the following key functions:

- **Data Capture:** This involves finding and bringing new data into the data system. The process often involves data validation, metadata extraction, and metadata translation. Validation involves checking for possible corruptions, data packaging, and data compliance. Metadata extraction involves gathering data on the data. It is a multifaceted operation since providers could embed

artifact metadata in many locations such as the file name, file headers, or the location of the file. Some metadata extraction are done by inference or derived from existing metadata. Finally, metadata translation involves mapping the provider metadata model into the data systems internal metadata model.

- **Data Catalog:** This involves longer-term persistence of metadata, linking the artifact with existing collections, and checking and apply dataset-specific storage policies. Various data management policies also involved such as storage management, stewardship policies, and security policies.
- **Data Search and Discovery:** An efficient and adequate search capability is as important as efficiently capturing the data artifacts. Search has become an important part of our daily computing lives. We use our favorite search engine to find our restaurants, movies, articles, and news, so we can conduct our business. Google and Bing are considered the "super-nodes" of information. Without an efficient and reliable search capability, a data center is of very little use to the general user.
- **Data Processing:** This usually involves one or more datasets with the goal of improving or enhancing captured artifacts. With NASA's EOS-DIS, data products are processed at various levels ranging from Level 0 to Level 4. Level 0 products are raw data at full instrument resolution. With higher levels, data are converted into more useful parameters and formats.
- **Data Visualization:** Data visualization is an important service to the users. It serves as an aid for data selection. It provides decision support and hazard management when near real-time data is involved. It is an instrument for conducting scientific researches. Visualization could be as simple as creating a quick look of a given piece of data artifact, or it could be as complex as a temporal-spatial time-lapse display of mash up of data artifacts.

As we are moving into the era of Big Data, the fundamentals of these key functions do not change. However, data centers are challenged to review their entire system architecture. It challenges them to think bigger. How will their handling of these key functions scale with bigger data volume and more complex data queries and operations? Can their system still able to deliver the kind of high quality services they are offering now? Our user community expectation is set by social media like Facebook and Twitter, and person-

alized web content experiences is the new norm. Data centers are not just challenged by the volume of data; they are challenged by their users expectations.

As an intelligent data archival system, OSCAR provided a reference-architecture for the next generation of Earth science data system. In particular, it provided reference to how semantic web should be applied. Fundamental to all data centers are three key operations: capture data, store data, and make data available. With the popularity of public cloud computing on the rise, moving onto the cloud could be one of the channels in acquiring elastic storage and reliability. The two key operations directly affected by Big Data include accuracy in data capturing and accuracy in searches.

5.1 Multi-Dialects Data Capturing

A data center for Earth science deals with unique challenges in data capturing. While the push for metadata standards such as ISO-19115 and the NetCDF Climate and Forecast (CF) metadata convention are influencing how some new data products are being defined, the reality of multi-dialects in metadata specification is still very much a reality in Earth science data artifacts. Internally all data centers have "standards" metadata architecture. It may not be any one of the existing standards, but rather a composition of all the current widely used standards as well as some local requirements. Taking PO.DAAC for example, its rich data model was shaped by the metadata standards it has to support, ISO-19115, The Federation Geographic Data Committee (FGDC), NetCDF Climate and Forecast (CF), and the EOS Clearing House (ECHO). The current process in bringing new datasets into PO.DAAC is to first review the datasets metadata model and implement a handler that translates the external data model into the internal data model. At PO.DAAC, this process has been streamlined in order to quickly bring in a new dataset. PO.DAAC has a reusable framework for constructing what it calls a data handler (Figure[12]). The product-specific handling logics include metadata harvesting and translation, data packing specification, and interface control.

The framework has evolved over the year and many of the built-in logics can be parameterized through configuration parameters and environment variables. The portion that still requires much manual labor is implementing the physical mapping between provider metadata attribute(s) to PO.DAACs internal metadata attribute(s). In the W3C standard Web Ontology Language (OWL), the class axioms `owl:equivalentClass`, `owl:subclassOf`, and `owl:disjointWith` are axioms used to define spe-

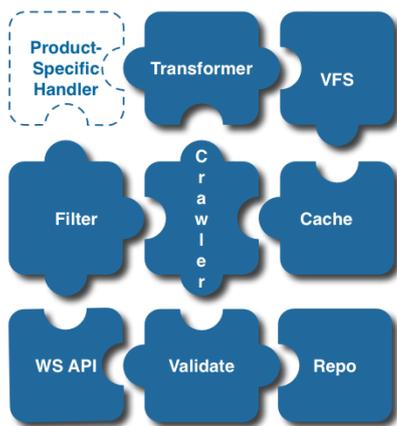


Figure 12: PO.DAAC DMAS Data Handler Framework

cialized relationships between concepts. Through ontology defined in OWL, computer software can look beyond the syntactic matches with a systematic approach to bridge between different concepts. Figure[13] shows an example of extending from the SWEET ontologies and define relationship between concepts. It is an example of specializing SWEET and bridging domain-specific concepts using equivalent class definitions. In the PO.DAACs data handler framework, the reasoning module should be added as the bridge from an external data model to the internal PO.DAAC data model. This reasoning module should be a component of the PO.DAAC data handler with an ontology implemented in OWL. When a new dataset is ready to bring into the data center, the data engineer only needs to update the data handler ontology in order for the data handler to understand the new dataset.

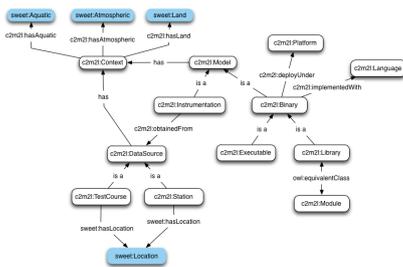


Figure 13: Extending from SWEET and linking similar concepts

5.2 Big Data Requires Intelligent Searches

Search is one of the most common operations we perform on a regular basis. Before the days of computer, we conduct our researches by searching through card

catalogs at a library. Often we had to work with several catalogs including books, publications, and medias. We search for a restaurant through phonebooks and when we arrive at the restaurant we search through their menu. In the era of relational database (RDB) (Codd, 1970), searches are performed by using the Structured Query Language (SQL). The relational part of RDB is defined by the foreign keys between database tables. Database normalization (Date, 2003) is a process for organizing tables in order to minimize redundancies and dependencies. The higher the Normal Form (NF), the smaller the database tables, which causes slower more complex database searches. While SQL is designed to be a flexible and intuitive query language, the maintenance of complex join queries is a nontrivial task.

Inverted index (J. Zobel, 1998) is an indexing data structure by mapping words to location in a database or documents. The conventional on-the-fly document parsing has proven to be very inefficient, since the time required depend on the number of the documents and the size of each document to be parsed. Creating a mapping of words to the documents enables fast full text searches. The popular Apache Solr enterprise search engine is deployed in various Earth science data centers and services. PO.DAACs metadata discovery service (Huang et al., 2011), an implementation of OpenSearch (Ogbuji, 2007) supports various metadata standard translation, is backed by Apache Solr, the entire PO.DAAC catalog is being incrementally indexed throughout the day.

The keywords indexed search approach can provide fast access to documents that are tagged or contain the matching keywords. While Solr does support some fuzziness and is also equipped with a built-in dictionary, the solution does not get into the meaning behind the keywords or any inference that could be derived. Another limitation is that it does not address any relationship between various pieces of artifacts. OSCAR demonstrated it is possible to create a sophisticated linked data system through ontology. In Earth science, observational data could be inferred through temporospatial, cause and effects, and natural phenomenon. Science users are already demanding better approach on helping them find the right datasets for their research. Searches in Earth science data systems must be more intelligent and organic. Not only does a search need to be able to coordinate between artifacts, it must also take into account user behaviors. User behaviors include frequent searched patterns, popular download metrics, online discussions, and user feedbacks.

6 CONCLUSIONS

The next generation of science data system (SDS.NEXT) must include knowledge. It must be able to quickly capture data artifacts from all data sources. It must be able intelligently identify the collection of datasets that are relevant to the intent of the users. It must be able to dynamically and intelligently link data artifacts together to deliver a complete picture of the data to the user. It must also be able to identify any relevant datasets through inference. Googles Knowledge Graph is a good example of what semantic search (Li and Yang, 2008) can deliver. As we are progressing toward the age of Big Data, it is time for Earth science data centers to consider adding knowledge and reasoning to their current pipelines.

ACKNOWLEDGEMENTS

This work was conducted at the Jet Propulsion Laboratory, California Institute of Technology under contract to the National Aeronautics and Space Administration.

REFERENCES

- Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13 (6):377–387.
- Date, C. J. (2003). *An Introduction to Database Systems*. Addison-Wesley, 8 edition.
- Droegemeier, K. K., Chandrasekar, V., R. Clark, D. G., Graves, S., Joseph, E., Ramamurthy, M., Wilhelmson, R., Brewster, K., Domenico, B., Leyton, T., Morris, V. R., Murray, D., Plale, B., Ramachandran, R., Reed, D., Rushing, J., Weber, D., Wilson, A., Xue, M., and Yalda, S. (2005). Linked environments for atmospheric discovery (lead): Architecture, technology road map and deployment strategy. In *21st International Conference on Interactive Information Processing Systems (IIPS) for Meteorology, Oceanography, and Hydrology*. Association of Computing Machinery.
- Fox, P., McGuinness, D., Raskin, R., and Sinha, K. (2007). Semantically-enabled scientific data integration. In *American Geophysical Union, Fall Meeting*, San Francisco, CA. American Geophysical Union.
- Huang, T. (2013). Transforming science data systems. In *Ground System Architectures Workshop 2013*. The Aerospace Corporation.
- Huang, T., Chung, T. N., Gangl, M. E., and Armstrong, E. M. (2011). Metadata-centric discovery service. In *American Geophysical Union, Fall Meeting*, San Francisco, CA. American Geophysical Union.
- Huang, T., Hardman, S., Bingham, A. W., Takagi, A., Chau, Q. T., and Gangl, M. E. (2009). Building the next generation data management and archive system. In *American Geophysical Union, Fall Meeting*, San Francisco, CA. American Geophysical Union.
- Huang, T., Quach, N. T., and Mukherjee, R. M. (2012). Ontological system for context artifacts and resources. In *American Geophysical Union, Fall Meeting*, San Francisco, CA. American Geophysical Union.
- J. Zobel, M. Alistair, K. R. (1998). Inverted files versus signature files for text indexing. *ACM Transactions on Database Systems (TODS)*, 23 (4):453–490.
- Li, W. and Yang, C. P. (2008). A semantic search engine for spatial web portals. In *Geoscience and Remote Sensing Symposium, IGARSS*, volume 2, pages 1278–1281. IEEE International.
- Ogbuji, U. (2007). Introducing opensearch. <http://www.xml.com/pub/a/2007/07/20/introducing-opensearch.html>.
- Ramachandran, R., Movva, S., Li, X., Cherukuri, P., and Graves, S. (2006). Noesis: Ontology based scoped search engine and resource aggregator for atmospheric science. In *American Geophysical Union, Fall Meeting*, San Francisco, CA. American Geophysical Union.
- Raskin, R. G. and Pan, M. J. (2005). Knowledge representation in the semantic web for earth and environmental terminology (sweet). *Computers & Geosciences*, 31:1119–1125.
- Singhal, A. (2012). Introducing the knowledge graph: Things, not strings. <http://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html>.
- Smiley, D. and Pugh, E. (2011). *Apache Solr 3 Enterprise Search Server*. Packt Publishing.