A Multi-Disciplinary, Model-Driven, Distributed Science Data System Architecture

Daniel J. Crichton[1], Chris A. Mattmann[1,2], John S. Hughes[1], Sean C. Kelly[1], Andrew F. Hart[1]

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

[2]Computer Science Department
University of Southern California
Los Angeles, CA 90089 USA

Abstract. The 21st Century has transformed the world of science by breaking the physical boundaries of distributed organizations and inter-connecting them into virtual science environments, allowing for systems and systems of systems to seamlessly access and share information and resources across highly geographically distributed areas. This e-science transformation is enabling new scientific discoveries by allowing for greater collaboration as well as by enabling systems to combine and correlate disparate data sets. At the Jet Propulsion Laboratory in Pasadena, California, we have been developing science data systems for highly distributed communities in physical and life sciences that require extensive sharing of distributed services and common information models based on common architectures. The common architecture contributes a set of atomic functions, interfaces and information models that support sharing and distributed processing. Additionally, the architecture provides a blueprint for a software product line known as the Object Oriented Data Technology (OODT) framework. OODT has enabled reuse of software for science data generation, capture and management, and delivery across highly distributed organizations for planetary science, earth science and cancer research. Our experience to date shows that a well-defined architecture and set of accompanied software vastly improves our ability to develop roadmaps for and to construct virtual science environments.

Introduction

The NASA Jet Propulsion Laboratory (JPL) has researched and built data intensive systems for highly distributed scientific environments for many years [2, 4, 6, 7, 10]. Due to the dynamic and changing mission environment for both solar system and earth robotic exploration, a number of critical architectural principles have emerged, helping us to define an architecture that can evolve with exploration and technological changes. Through our work at JPL, we have defined an architectural style for data and computational grids that is focused on the capture, processing, discovery, access, and transformation of digital data objects (and their rich metadata descriptions) across highly distributed environments. The framework, called the Object Oriented Data Technology (OODT) framework [2, 10] was selected as runner up for NASA Software of the Year in 2003 and has been extensively used not only within physical science environments such as planetary [7, 8], earth [6, 11], and astrophysics [12], but also in biomedical research [4, 5].

One of the central characteristics of the architecture is the application of architectural patterns [13] consistently across very different science environments. OODT stresses up front the aspects of the architecture that are common, leaving the domain-specific aspects (where/how to reuse existing modular OODT components, and non-functional parameters of the architecture like scalability, efficiency, etc.) to be ironed out and iterated upon during system development.

Over time, informed by our growing experience designing information systems to support scientific research, we observed common architectural patterns and canonical sets of services central to the successful development of systems within the different domains. The services include:

- **Data capture** – dealing with metadata extraction, content analysis and detection (MIME-type and language detection) [15], along with validation against common metadata model e.g., ISO-11179 [16], and Dublin Core [17].
- **Data discovery** – dealing with the ability to describe resources (data, computation, identity, etc.) in a uniform fashion, and the methodologies for using those resource descriptions as a mechanism for discovery.
- **Data access** – dealing with the acquisition of data from heterogeneous stores (RDBMS'es, filesystems, etc.) using a uniform access method.
- **Data processing** – dealing with transformation (subsetting [18], interpolation, aggregation, summarization, etc.) of data once it has been accessed.
- **Data distribution** – The packaging of data and its metadata, and the plan for its eventual distribution to users downstream of the system.

These services allow for distributed, independent deployment, yet maintain the ability to work in concert with one another when needed. Building systems in this fashion allows construction of large-scale, virtual information systems that span organizational boundaries.

A second observation repeatedly impressed upon us through experience was the valuable contribution of a well-defined information architecture [1]. The information architecture formally characterizes the data that is manipulated by the system, and is critical to realizing the domain implementation. As part of designing the information architecture for any domain, we have been actively involved in developing a standard information model for the representation of information associated with data objects managed within different scientific domains. The data objects that are captured, managed and exchanged by the system are described in the information architecture by a "metadata object" which provides a set of attributes for the data object, and relationships between objects, as described in the domain information model.

The OODT framework provides a set of core services and architectural patterns that simplify implementation of the above functions, which themselves are informed by the domain model (e.g., a cancer biomarker information model, a planetary science information model, etc). The loose coupling between each service and its associated domain model allows for the services to be easily developed to support multiple domains. Each of the OODT services can be deployed independently and then can be integrated using XML-based interfaces over a distributed, grid architecture. This service independence and insulation makes it possible to minimize the effects of organizational boundaries on accessing data repositories (either local or distributed) concurrently, compiling the results into a unified view, and making them available for analysis. The OODT framework is based on the software architectural notion of components [13]. Each component has well known interfaces that enable them to be plugged together in a distributed, yet coordinated, manner. The components themselves sit on top of off-the-shelf middleware technologies so that they can be deployed easily into an enterprise topology.

Each of our domain implementations is working to build domain-specific applications on top of the common services framework provided by OODT. For example, the NASA Planetary Data System (PDS) used a Lucene-based search engine [19] that integrated with OODT to provide millisecond-speed searching across highly distributed databases using a text-based search interface. The benefit of the framework to these projects is that it has substantially helped in both building new data systems as well as integrating existing data systems, all while controlling software development costs through software reuse and standardized interfaces.

In this chapter, we will discuss the architectural patterns and experience in implementing an e-science [20] product line. The chapter will highlight the technical, scientific, management and policy challenges associated with building and deploying multi-organizational data systems. It will compare and contrast differences between planetary, earth and biomedical research environments and discuss the importance of a well-defined architecture and the need for domain information models. It will discuss key architectural principles in the design as well as the importance of having a well-defined operational model to ensure both reliability of the system as well as quality of the data and services.

Table 1. Architectural principles derived from our experience in the domain.

Principle		Description
P1	Access and Correlation	e-science software should providing uniform methods to bring together data in distributed environments to increase the chances of discovery.
P2	Location independence	Users of e-science software should not concern themselves with the physical location of data or services.
P3	Well defined information architecture	Software changes rapidly in e-science systems. Data models and metadata attributes do not. Systems that can easily support this evolution are desired.

Applying e-Science Principles to Science

In this section, we will motivate some of the critical architectural principles derived from our experience in the e-science domain constructing systems with OODT. Each principle that we detail below is summarized for the reader's convenience in Table 1.

Collaboration is a critical aspect of scientific research. Multi-center and multi-institutional collaborations are often critical to support and validate scientific hypothesis. Yet, far too often, systems are not architected to support construction of virtual scientific environments, particularly in support of performing analysis of distributed data. It is essential that the capture, management and distribution of scientific data resulting from scientific studies and research be considered in terms of its value to sharing data. The *access and correlation of data* (P1) across distributed environments is critical to increasing the study power and validating the data from greater number of samples and contexts [5].

What we have found from our technology development of virtual scientific networks is that *location independence* (P2) has become a critical architectural tenant for the construction of modern e-science data systems. Location independence prescribes that the physical location of data and components should be transparent to those accessing them. In other words, whether data and software are local or are geographically distributed should not matter to human or application users. The implication is that the access and interpretation of the data objects should remain consistent despite multiple topologies for the system that may be in place.

As part of our work in the planetary science and cancer research communities (that we will elaborate on in Section 5 and again in Section 7 respectively) it has become apparent that a *well-defined information model* (P3) consisting of both rich data attributes implemented using well known standards (such as ISO-11179 and Dublin Core) is also an important architectural principle. The planetary science data model [7, 8] consists of a set of over 1200 data elements, including terminology such as *Target* to identify the celestial body targeted by the mission's instrument(s); *Instrument* to denote the name and type of the scientific instrument flown on the mission that records observations, and *Mission* to denote the unique name of the NASA mission for which data is being archived. One the cancer research side, we have developed a group of over 40 data elements [4, 5], including *Specimen Collected Code*, an integer value denoting the type of specimen, e.g., blood sputum, etc., collected for a patient; *Study Site Id* which denotes a numeric identifier for a participate cancer research site; and *Study Protocol Id*, a numeric identifier denoting the protocol under which data has been collected, to name a few.

Though technology changes rapidly, the above work on data models does not. In the case of the planetary model, changes have been limited over the past 20 years; an attribute was added here or there to account for some new mission, but those changes are few and far between – in all, 10s of the 1200 elements may have been modified, or added to. On the cancer research side, the same 40 data elements to describe cancer research data have been leveraged over the past eight years in the context of the National Cancer Institute's (NCI) Early Detection Research Network (EDRN) project, again, with similar experiences – some new instrument technology, or new application drives the creation of a few attributes here and there; nothing more. These examples illustrate the importance of a *well-defined information model* (P3) as a means of allowing software technology and data modeling to evolve independently of one another.

In the next section we will describe our work on the Object Oriented Data Technology (OODT) framework, and its architecture, and demonstrate the relationship of the two to the aforementioned architecture principles summarized in Table 1.

The Architectural Model and Framework

"Expect the unexpected" has been the driving mantra behind OODT. Years of experience building implementations of this architecture for domains as diverse as planetary and earth science and cancer biomarker research have repeatedly impressed upon us the need for a flexible, architecturally principled core platform of software and services upon which to build domain-specific extensions. Our approach has favored using a core set of loosely connected, independent components

[13] with well-defined interfaces over the more traditional monolithic system architecture. A number of observations culled from our experience have helped to influence this design decision. We can directly map these observations to the three architectural principles (recall P1-P3 from Table 1) described earlier.

The e-science domain [20] is focused on science, which in turn is focused on observation. Scientific instruments collect observations in the e-science world. For many decades, the resolution and frequency of the data returned from these instruments was minute, and disk space was expensive [21, 22]. In modern times, disk space is cheap, and instrument resolution and data capture ability is growing faster than the e-science systems that regularly must deal with the data. This situation has made it critical to develop e-science software based upon an overarching construct that was both open-ended and standards-based, to allow for necessary extension (principle P3 from Table 1). Furthermore, science is often subject to political considerations and policy factors that are subject to change. This oft-uncertain landscape amplifies the need for a system that can be quickly evolved to meet unexpected changes in the operational environment (principle P3 from Table 1).

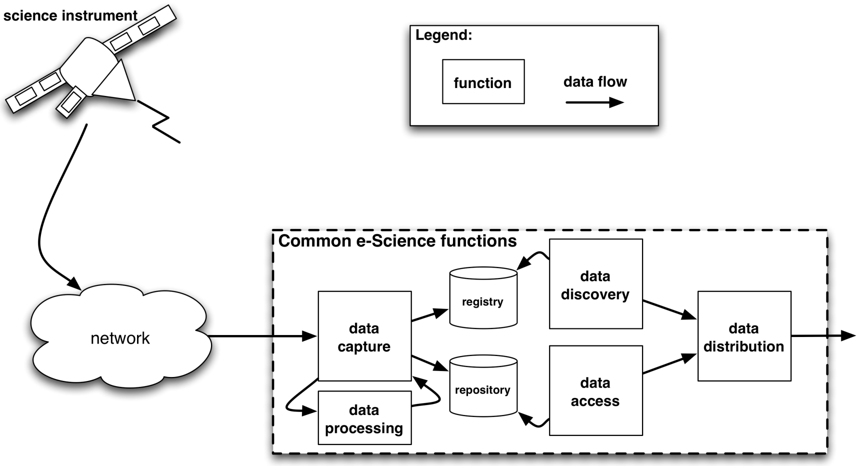


Figure 1. Common e-Science functions derived from architectural principles in Table 1. The relationship between the functions is demonstrated as data flow beginning with data capture, culminating with data distribution.

Most data-intensive scientific information systems can be deconstructed into combinations of basic concepts of data *capture*, *discovery*, *access*, *processing* and *distribution* (recall Section 1) as demonstrated in Figure 1. The figure demonstrates the canonical science data pipeline in use by many e-science projects: in the upper left portion of the figure, a scientific instrument (in this case, represented by a remote sensing instrument, but the same would apply to any type of observing sensor, e.g., a microscope, etc.) records data, and then sends it over a network to a data capture function. That data capture function then sends the recorded observations to a data processing function (either one time, or a series of times), which in turn may further process the provided data (and its descriptions, called *metadata*, or “data about data”), for example, if the data is an image, by down sampling the image, or resizing it. That processed data is then provided back to the data capture component, for persistence – the data is stored in a repository, and the metadata is stored in a registry. The data and metadata are then exposed downstream to users of the e-science system by a data discovery function (allowing search and discovery against the registry), and by a data access function (allowing the physical bits captured in the repository to be accessed). The combination of the retrieved data and metadata is then provided to a data distribution function for ultimate distribution to the community (occurring in the bottom right portion of Figure 1).

By modeling these core concepts as a collection of loosely connected components we have found that we can selectively utilize and re-arrange them to create a variety of scientific environments uniquely suited to the needs of specific projects, independent of the project domain. In other words, some projects will have a strong focus on e.g., data ingestion and data distribution, but not so much on that of data processing (planetary science is an example, as well as cancer research). However, on the other hand, other projects (and even science domains) will focus entirely on that of data ingestion, and data processing, omitting a strong focus on data distribution, or on data discovery). With the base components in place, domain-specific intelligence can be layered on top to provide customization and tuning to the environment (principles P1 and P3 from Table 1).

Finally, the scope of the challenges being addressed across scientific disciplines today has driven a trend towards increased collaboration and partnership among researchers, often crossing organizational and institutional boundaries (principle P2 from Table 1). This new reality has placed a premium on the perception of location-independence of data from the perspective of access and processing (principle P2 from Table 1). As will be evident from the following sections, we have found that the federated component model provides a powerful mechanism for connecting distributed data holdings into virtual scientific environments in which the physical location of data is largely transparent to system users.

Particularly for multi-institution implementations of large-scale data processing systems, the use of open, standards-based protocols for communication between distributed components of the system architecture is critical (principles P1, P2 and P3 from Table 1). Effective data capture, manipulation, storage, and dissemination are all predicated upon the existence of a shared protocol for communicating representations of data between components. Our approach has favored open standards like XML-RPC [23], Resource Description Framework (RDF) [24], and Representational Entity State Transfer (REST) [25] to improve its integration potential into as diverse an ecosystem as possible.

So far, we have restricted much of our focus to detailing the common software functions that are part of our reusable e-science architecture and framework. In the ensuing section, we will hone in on the information architecture, and discuss OODT's focus on reusable models and patterns for representing data in an e-science environment.

An Information Centric Approach

Data intensive systems in the e-science era must not only meet the expectations of a new generation of internet savvy scientists but as distributed scientific data repositories, they will also be expected to support science in ways not conceived of when the system were originally designed. To meet these expectations there must be an unambiguous specification of the data objects the systems manage and the context within which they exist in the targeted domain. These specifications must contain a broad range of modeling information, from classical data models that define the structure of the data objects to descriptions of the science context in which the data objects exists. In addition, to enable the potentials of the semantic web [26], the specification must also define a rich set of relationships between the data objects in the domain to allow machine reasoning. Finally to support system interoperability at the data level, shared models must be developed by science domain experts to provide a common domain of discourse for both scientists and machines.

The information model is a key component of an e-science system. Lee [9] has defined an information model as a representation of concepts, relationships, constraints, rules, and operations to specify data semantics for a chosen domain of discourse.

In the Object Oriented Data Technology (OODT) reference architecture, and framework, an information model is thought of as a network of data models where each data model deals with one or more aspects of the system. For example, the Planetary Data System (PDS) information model has data models for each of the four fundamental data structures used to store digital objects, such as images of the planet Mars. Other data models exist for the science interpretation of the images, the time and geometry data needed to register the image on the planet's surface, and the descriptive information about the planets including identification attributes and web-resource links for publications and authoritative information sources. Another data model prescribes a structure for packaging data objects into products that are registered, searched, located and retrieved. Finally the information model as a whole puts the products into their science context by defining the associations between products and adding taxonomical information such as asserting that Mars is a Terrestrial planet in the solar system.

Ontology modeling tools, used to model the domain, are leveraged often in OODT. The tools help to explicitly record each "thing" in the domain as a class. For example, data product, target, and investigation are all modeled as classes in the PDS ontology as shown in Figure 2. Figure 2 illustrates a few of the higher-level classes and their relationships that have been defined in the PDS information model. More specific things such as "planet" exist as subclasses. A preliminary list of things to be modeled can often be identified in the functional requirements of an information system. The resulting information classes are then operated on by the system's functions and services, aiding in addressing architectural principle P3 from Table 1.

Functional requirements for the e-science domain typically include those mentioned earlier in Sections 1 and 3 (recall: data *capture*, *processing*, *discovery*, *access* and *distribution*). These requirements suggest class attributes. For example basic management of an object, such as object capture, suggests the need for a unique immutable identifier, a title for display purposes, a version identifier, and some type of description. An object status attributes is suggested by life cycle management functions.

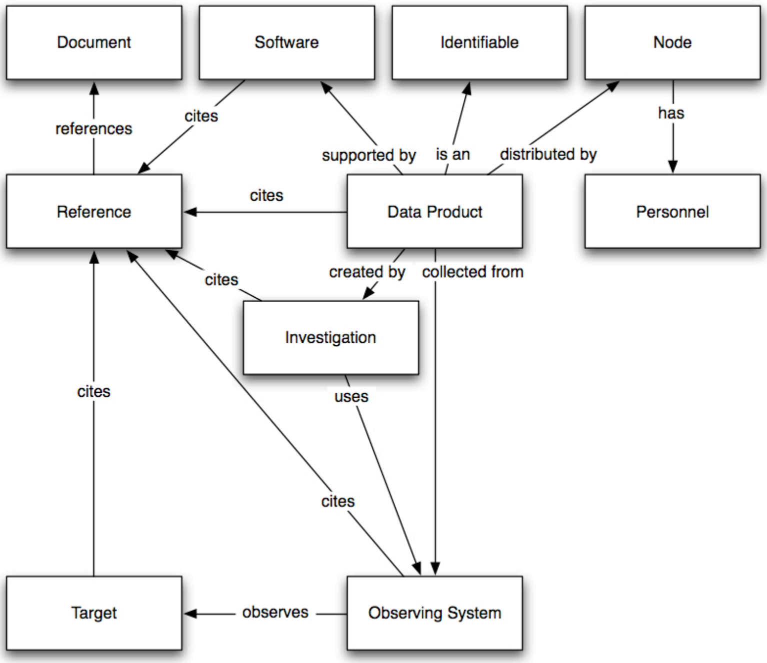


Figure 2. Concept Map – PDS Classes and Relationships

The data processing requirement suggests the need for attributes that formally define the object's data structure. For example, the fundamental structure used for a grayscale image, an array, must have attributes that provide the dimension of the array, the number of elements in each dimension, and the array element data type.

The discovery and distribution functions both suggest a richer model, for example coordinate system attributes support common geographical information system queries on terrestrial planet surfaces. However finding features in Saturn's rings or tracking a storm in Jupiter's atmosphere requires dynamic metadata from complex calculations in addition to that metadata that is statically generated. Finally correlative information discovery and distribution requires shared models with common taxonomies and associations across classes to meet requirements.

A vital concept within OODT and within the e-science domain as a whole is the *information object* [12]. Formally defined as the unique combination of a *data object* (the bits) and its descriptive metadata (or its *metadata object*), the concept is used to uniformly describe, to allow for comparison, and to identify all things in the e-science domain into a core component for the model. For example, a Mars image is a digital instance of a data object, a sequence of bits. Metadata is associated with the data object to define its structure and describe the object so that it can be processed and made useful to scientists. In a similar manner, conceptual things like investigations and physical things like instruments are modeled as information objects as well. This concept is illustrated in Figure 3.

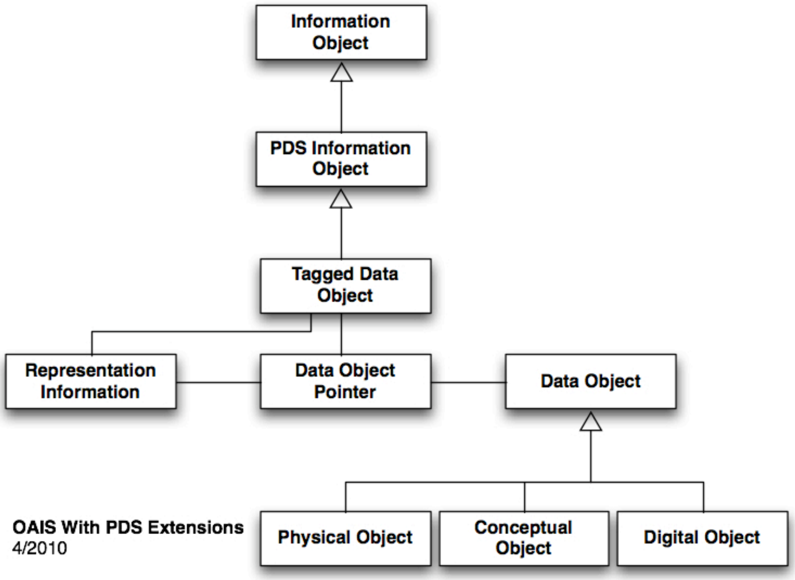


Figure 3. UML class diagram of information object, adapted from [12].

One of the canonical elements of an information model is its *data dictionary*. Whereas an ontology focuses on the definition of classes using attributes, a data dictionary focuses on the definition of the attributes. Intuitively a data dictionary defines an attribute as having a name, description, and a value. Our work within the context of e-science domains and OODT has led us to define attributes using a standard, more comprehensive data model. The model manages attributes separately from an attribute's permissible values and provides a range of specifications from effective dates, registration authority, submitter, steward, and classification schemes to the use of one or more natural languages for definitions. Data dictionaries also provide a means of defining the language of discourse for the e-science domain, namely the terms used by the scientists and the system to communicate. The importance of a standard model for the data dictionary is especially evident when considering system interoperability at the data level. System interoperability is best built by laying a common foundation for communicating the most basic components in the system, the attributes used to define e-science domain terminology.

Based on our experience in the context of OODT, a shared ontology is the single most important element for enabling system interoperability and science data correlation. Uschold [14] states that the process of assembling a single shared ontology automatically from separately developed ontologies is essentially cryptology. This is also true regarding the development of interoperable systems from disparate information models. The model driven aspect of the OODT infrastructure focuses on the use of an ontology to generate almost all of the design, implementation, and operational artifacts, all the way from the information model specification and data dictionary to registry configuration files and XML schema (recall, this is a core architectural principle, P3, allowing for the separation of data and software models, as described in Table 1). We have summarized the OODT model driven process in Figure 4.

The system requirements and domain knowledge are captured in an ontology-modeling tool and exports from the ontology database are translated to various notations depending on the need (as shown in the upper left portion of Figure 4. For example, an XML Metadata Interchange (XMI) file is generated for import into UML modeling tools for the creation of UML class diagrams and potentially software code. XML Schemas are generated for generating and validating XML documents used to capture metadata. RDFS/XML [24, 26] and OWL/XML [26] are supporting technologies used to implement search/browse functionality, and used traditionally in OODT based project implement capture, discovery, and distribution of information in the OODT system.

Armed with OODT's reference architecture, its core functions and principles, and its information architecture focus, the following sections illustrate real OODT deployments in the domains of planetary science, earth science and cancer research. Along the way we will tie back the domain requirements, functionality and ultimate architectural and implementation principles discussed, illustrating OODT's ability to effective model and implement software in the e-science domain.

The Planetary Science Model

The planetary science discipline has engendered scientific achievements that are poised to stand the test of time. The robotic missions that have been flown to study the solar system represent some of mankind's greatest engineering achievement. Yet, the design, launch and observations made by the spacecraft developed represent only part of the story. Capturing, processing, sharing and analyzing the scientific results are critical stages in the overall mission necessary to increase the understanding of the universe in which we live. The planetary science data sys-

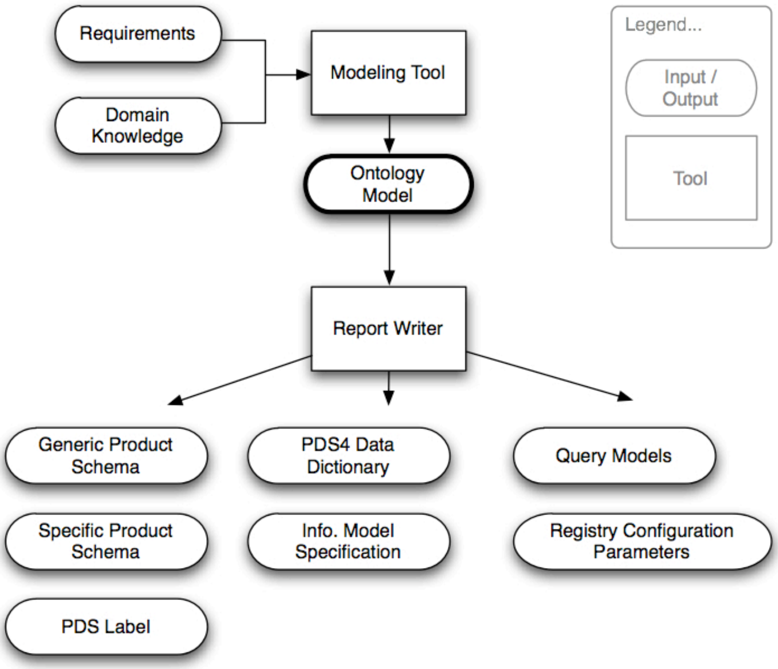


Figure 4. The model driven process of the OODT architecture: separation of software and information model allows each to evolve independently. At the core of the process is the Ontology, used to codify the requirements, flow through functionality to the actual components, and ultimately validate the implementation of OODT architectures.

tems are invariably distributed and must be designed to support new science investigations with a variety of different types of data from images to complex data structures. Yet, there is a critical need to ensure that these systems can be interoperable to allow for interdisciplinary research as well as research multiple missions and studies.

In the early 1980s, the National Research Council formed the Committee on Data Management and Computation (CODMAC) [27]. CODOMAC focused on making a number of recommendations on the long-term management of planetary science data. The NRC report identified seven core principles (1) Scientific involvement; (2) Scientific oversight; (3) Data availability including usable formats, ancillary data, timely distributed, validated data, and documentation; (4) Proper facilities; (5) Structured, transportable, adequately documented software; (6) Data storage in permanent and retrievable form; and (7) Adequate data system funding.

In the late 1980s, the United States National Aeronautics and Space Administration (NASA) formed a facility known as the Planetary Data System (PDS) [7, 8] that is responsible for curation and management of all scientific data results from robotic exploration of the solar system. The structure of the PDS is based on the CODMAC report organized to provide scientific expertise on the use of the discipline-specific scientific data sets by the worldwide scientific community. Over the years, the PDS has become a national resource, housing well over 100 terabytes of data across eight nodes covering NASA missions starting in the 1960s. These nodes cover scientific discipline areas including planetary atmospheres, geosciences, imaging, magnetospheres, radio science, planetary rings, and small bodies. A node covering overall engineering of the system is based at the Jet Propulsion Laboratory. The overall structure of the PDS is depicted graphically in Figure 5.

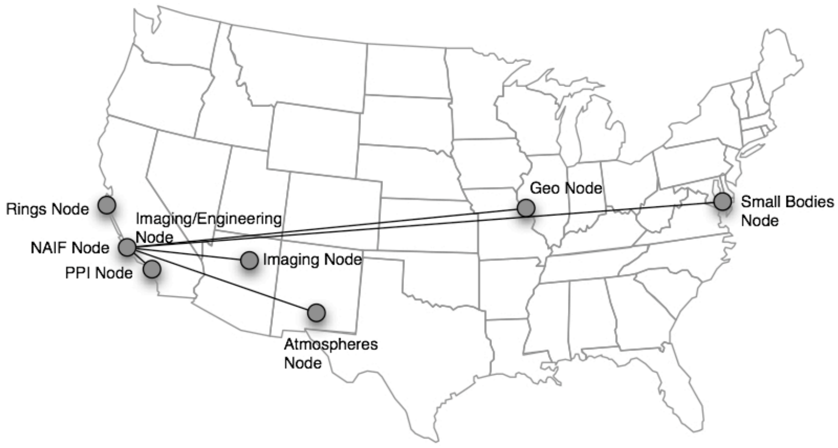


Figure 5. The geographic distribution of NASA's Planetary Data System. There are nine nodes geographically distributed across the U.S., broken down by scientific expertise.

PDS has been a leader in defining data standards, working with missions and instrument teams, and developing data system technologies. It has also been instrumental in changing the scientific culture by working with the planetary science community to publicly release and peer review the data it captures. It is often cited as a model by other domestic and international science data systems doing leading-edge scientific research [7, 8].

PDS has made several critical “architectural” [7, 8] decisions that have been paramount to its success. In the spirit of the OODT architectural principles (recall principle P3 from Table 1) PDS has defined a *Planetary Science Data Model* that all missions conform to when submitting data to the PDS. Having a common data model allows for searching across nodes, missions, instruments and products in a uniform manner which is important for turning the PDS federation into an integrated enterprise, as well as addressing a core e-science function of data discovery (recall Section 3). While many disciplines are addressing semantic interoperability after data has been archived, PDS is working, as early as possible, with the missions so they adopt the PDS data standards and use common terms for documenting science data. PDS’s common data model, having transcended several technology upgrades and changes to the system, has remained critical to the entire project since its inception (in line with principle P3 from Table 1).

PDS continues to evolve towards a broader vision of an online, distributed system based on international standards. The focus of PDS, over the next five years, is to enable the PDS to move towards a fully online, distributed system that supports the evolving needs of both PDS data providers and users while improving the overall efficiency and reliability of the system. A further objective of PDS is to continue to architect tools that can be deployed in a variety of heterogeneous computing environments to allow for specific adaptation and use within different mission contexts as early as possible in the life of a mission, helping to address principles P1 and P2 from Table 1, and ultimately to realize the necessary e-science services such as data access, discovery and distribution as shown in Figure 1.

The PDS itself is a classic virtual organization where the organization represents a number of distributed elements, principally people, data and systems. The purpose is to build a homogeneous federation of archives to promote greater interoperability and construction of the virtual science environment for planetary research. This leads to common governance challenges whereby policies for local versus federal control and standards must be well defined. The PDS allows a substantial amount of autonomy at each of the nodes, but requires that all data that is produced and captured within the PDS be compliant to a common set of data standards (addressing principle P3 from Table 1). This common model has helped to improve the ability to access and integrate data that is physically distributed across the PDS network. As PDS has evolved is technical implementation over the years, there has been continued migration towards create a single, virtual system, where discovery and access to the data is transparent to the user. In other words, the physical topology of the system becomes less important as the maturity of the system and movement towards virtualization continues (in line with principle P2 from Table 1). This is illustrated in many ways by the recent work (2006 and beyond) helping to form the International Planetary Data Alliance (IPDA) [37]. The IPDA is an international standard organization, focused on the development of international standards for the purposes of enabling interoperability and data sharing of planetary science data archives across space agencies. In large part, many of the early work on IDPA has focused on implementing a common set of functions, defining the necessary information architecture, and realizing an implementation driven by the e-science aspects of OODT employed by PDS discusses thus far in this section. Beyond the location independence (principle P2 from Table 1), necessary access (principle P1 from Table 1) required for federating PDS within the U.S., moving to an international virtual organization has only strengthened our belief in the small set of core principles upon which e-science systems can be based.

Besides the service-focused principles, even more so in PDS the information architecture (principle P3 from Table 1) emerges as a critical component of the system. PDS's information architecture largely employs the use of data dictionaries, core data elements, domain models, and other information-centric principles (recall Section 4) necessary in the e-science domain. Specifically, for our work on PDS and in planetary science as a whole, we have constructed an ontology that describes the planetary objects and their relationships within the domain. The ontology model allows capture of rich semantics within the model and mechanisms to export the model into both schemas and standard documentation for use by data producers within a mission (recall Figure 4). The model contains the core ele-

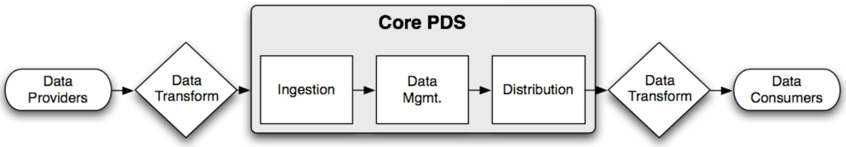


Figure 6. Information flow within the Planetary Data System (PDS).

ments of planetary science (missions, targets, spacecraft, data, etc) and is extended to engender domain-specific data services (subsetting, coordination transformation, mining, etc) beyond those core e-science services (recall Section 3) common to many e-science systems. Data that is captured and sent to the PDS is validated against the model to ensure semantic and syntactic compliance. The purpose is to build a homogeneous federation of archives.

Figure 6 shows the common information flow for data, whereby data providers (missions, instrument teams and individual principal investigators) submit data to the PDS that is stored within the distributed system, and then distributed to the data consumers.

In summary, the planetary science community is benefiting from the e-science paradigm change through the ability to access, search, download and use scientific data results from missions. Without a well-defined data and software architecture, this would not be possible. The core data standards developed for the NASA Planetary Data System, for example, have been essential for representing metadata and data in a common way and ensuring that it can be located across highly distributed repositories and then loaded into common tools. The existence of such standards has helped to pave the way towards greater interoperability at an international scale.

Earth Science Research

Earth Science is another domain that has complex data sets that are captured across a variety of distributed data systems. These systems capture and process observational data acquired from satellites as well as other measurement instruments in a variety of data formats using different information models. In addition to capturing observational data, a significant amount of work occurs in the development of complex scientific models to analyze such challenges as climate change and weather prediction. As the computing capabilities have increased, there has been significant interest in sharing data across various communities and data systems. One such example is in the area of climate change to compare climate models to satellite observations.

Over the course of next few years, the Intergovernmental Panel on Climate Change (IPCC), the leading international organization studying global climate change, will undergo a battery of experiments whose results will be recorded in the 5th Assessment Report, or AR5 [28]. The experiments are geared towards simulating dozens of climate related variables, from air pressure, to sea surface salinity, all the way to the world's temperature, which has been a huge subject of debate and interest (inter-)nationally, and of which major U.S. and global funding initiatives have arisen from.

The UN Climate Change conference in Copenhagen meetings held during December 2009, which included participation from some of the most influential members of our global society, including U.S. President Barack Obama, highlighted the importance of the upcoming IPCC AR5 activity. Decades-long climate model simulations over multiple variables and parameters require massive amounts of data and computation in order to provide meaningful results in a timely fashion. Further, these simulations require complex climate models, which themselves require tuning and observation by hundreds of scientists looking to identify the next important prediction that can be used to inform national policy and decision making based on the Earth's climate.

A recent IEEE workshop[1] brought together IT professionals and climate researchers with the goal of understanding how information technology, grid computing, data science and computer science could be brought to bear to help climate scientists participating in AR5. One of the principal conclusions of this workshop (as well as that of a meeting[2] that preceded it) was identifying the role of technology in the AR5 was helping to shepherd in observational data as a means of climate model improvement and diagnostics. As it turns out, though the prior IPCC model runs (AR4) was deemed widely seminal, and produced over *2000* peer-reviewed science publications, the organizers of AR5 believe that the reliability of projections (and also the number of publications resultant from this activity) could be improved if the models were validated and measured against remotely sensed observations.

Within the last year, the Climate Data eXchange (CDX), an effort to improve use of NASA's earth observational data in the improvement and analysis of climate model outputs, was initiated under the supervision of NASA's JPL [29]. The major focus of CDX is directly *enabling* the aforementioned IPCC activity, and to provide NASA observational data products (both raw level 2 in the long-term and level 3 in the short-term) to the IPCC AR5 community. The data products vary broadly in their formats (e.g., HDF vs. netCDF), geographic coverage, access

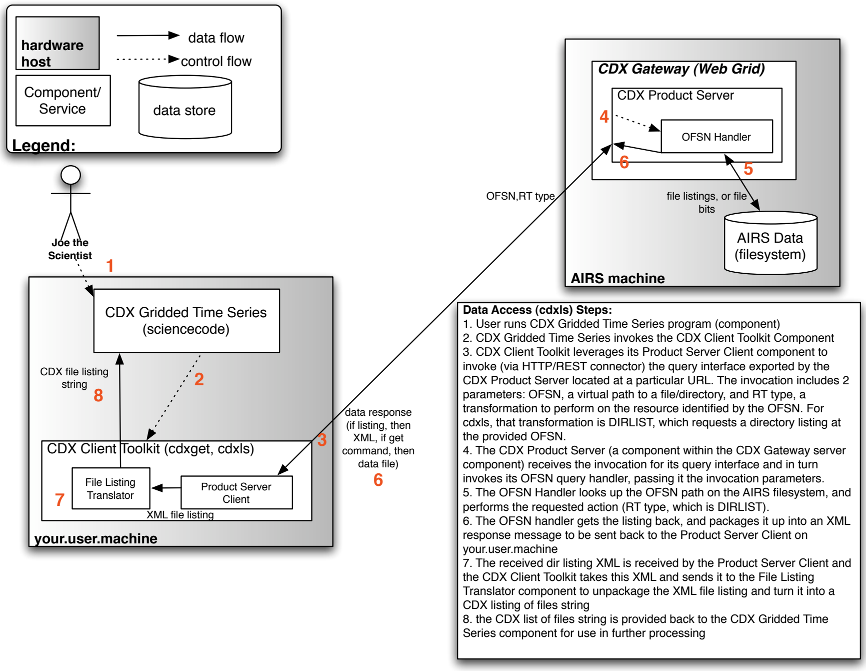
[1] http://smc-it.org/workshops/crichton.html

[2] http://www.ipcc.ch/workshops-experts-meetings-ar5-scoping.htm

methods, and volume. Additionally, the science and observations within the data files are highly instrument specific, including temporal and spatial properties that must be harmonized in order for model comparison.

The crux of the problem is that global climate models provide measurements of parameters in all places at all times for which the model is run; observational data, on the other hand, does not. In turn, the CDX project's focus is that of obviating these heterogeneities and providing an open source software toolkit for use in the IPCC AR5, and to help its science users rapidly and programmatically improve and validate climate models.

A large effort has been made to deploy web-services and a client toolkit based on OODT [10]. Much of the focus on leveraging OODT for CDX to date has been to expose data access, discovery and processing (subsetting) services (recall Figure 1 and see Figure 7) provided by NASA's mission science computing facilities, specifically the Atmospheric Infrared Sounder (AIRS), the Microwave Limb Sounder (MLS), CloudSAT, and the Multi-angle Imaging SpectroRadiometer (MISR). OODT is focused on providing the substrate for unlocking data, metadata and computations; the orchestration of those operations is provided by the CDX client toolkit as shown in Figure 7.



Data Access (cdxls) Steps:
1. User runs CDX Gridded Time Series program (component)
2. CDX Gridded Time Series invokes the CDX Client Toolkit Component
3. CDX Client Toolkit leverages its Product Server Client component to invoke (via HTTP/REST connector) the query interface exported by the CDX Product Server located at a particular URL. The invocation includes 2 parameters: OFSN, a virtual path to a file/directory, and RT type, a transformation to perform on the resource identified by the OFSN. For cdxls, that transformation is DIRLIST, which requests a directory listing at the provided OFSN.
4. The CDX Product Server (a component within the CDX Gateway server component) receives the invocation for its query interface and in turn invokes its OFSN query handler, passing it the invocation parameters.
5. The OFSN Handler looks up the OFSN path on the AIRS filesystem, and performs the requested action (RT type, which is DIRLIST).
6. The OFSN handler gets the listing back, and packages it up into an XML response message to be sent back to the Product Server Client on your.user.machine
7. The received dir listing XML is received by the Product Server Client and the CDX Client Toolkit takes this XML and sends it to the File Listing Translator component to unpackage the XML file listing and turn it into a CDX listing of files string
8. the CDX list of files string is provided back to the CDX Gridded Time Series component for use in further processing

Figure 7. The Anatomy of a CDX operation. The client toolkit provides underlying data access services to the example application (CDX Gridded Time Series) by remotely contacting the CDX gateway service on the airscdx machine.

The interactions illustrated in Figure 7 demonstrate the manner in which CDX and the principles of OODT are changing the regular day-to-day activities of a climate researcher. We will use a use case scenario to demonstrate. In our scenario, the climate researcher desires to build a time series comparison of a particular measurement (we will select water vapor for the purposes of discussion; many other measurements could be used) available from the AIRS data system. To begin, the researcher would select a series of observations for a time range, and then, for each day in that time range, download (recall data access from Section 3) around 240 Hierarchical Data Format (HDF) version 4 files [30] to her local drive, the sum of which would be used for a ground truth comparison against simulated NCAR Community Climate System Model (CCSM) version 3 data containing measurement predictions, e.g., for water vapor. The model data, however, are stored in a separate archive, and in a different data format, NetCDF [31]. Once the data is downloaded (again, recall data access from Section 3, and Figure 1), both sets (the HDF and netCDF) of data are loaded via an OPeNDAP interface [32] (recall principle P3 from Table 1) into a few Python scripts, responsible for: (1) averaging the observational data and ensuring it is on the same space/time continuum; (2) computing a statistic, e.g., an average or a covariance needed to assess the observational data against its predicted values from the model (recall, data processing, and data access from Section 3 from Section 3, and Figure 1).

As shown in Figure 7, the CDX approach for addressing this use-case scenario involves pushing as much of the computation as close to the data as possible, insulating location of data and transference of service to the OODT middleware layer (steps 3, 4, 5 and 6 from Figure 7) as possible, and ensuring the time series component is unaware of the actual remote data access and computation that is occurring (principles P1 and P2 from Table 1). The transformation of the water vapor observational measurements is removed from the actual Python program, and pushed to the remote OODT product service, co-located with the AIRS data as shown in the upper right portion of Figure 7), addressing data processing, and solidifying its interface with data access, as demonstrated in Section 3 and Figure 1.

To date, we have leveraged the CDX infrastructure and client toolkit to directly enable two critical use cases for climate change. The first example involves delivering NASA observational data to the Earth System Grid gateway at Lawrence Livermore National Laboratory (LLNL) with the direct intention of sharing the observational data for AR5 – to date, AIRS level 3 data, as well as MLS level 3 data has been delivered to the Earth System Grid, with the information being provided by the underlying CDX infrastructure. The second major use case involves performing model to observational data time series comparisons between AIRS level 3 data, and NCAR CCSM model output, available from LLNL, as described above.

Our experience has shown that a well-defined architecture and a set of common standards and software components are useful for deploying and building e-science architectures. Given the maturity of our work with the OODT software framework and the development of common information and software architectures, the Climate Data Exchange came together very quickly. While the common problems of heterogonous data systems existed, the experience and technologies available allowed us to deploy an infrastructure that could access climate observations and models, and bring them together into an environment that allowed for greater scientific discovery opportunities.

Cancer Research

The capture and sharing of scientific data to support advances in biomedical research is another domain that is benefiting from the e-science paradigm. As we have seen above in the planetary and Earth science disciplines, cancer research has experienced an explosive growth over the past decade in the amount of raw data produced by observational instruments. Furthermore, the inherent complexity of the challenges facing cancer researchers today has made the cooperative collaboration among geographically distributed researchers an attractive approach. As a direct result, the development and utilization of informatics tools capable of supporting these new "virtual organizations" has taken on a new importance in this domain as well – and so has the notion of e-science systems as the majority of this chapter has focused on.

In 2000, the Early Detection Research Network (EDRN) was formed as a collaborative research organization funded and led by the Cancer Biomarkers Research Group of the U.S. National Cancer Institute [4]. The EDRN consists of scientists from more than 40 institutions around the United States who are focused on the discovery and validation of biomarkers for the early detection of cancer [4, 5]. The EDRN program has required an informatics infrastructure that is tightly integrated with its scientific program and supports the capture and sharing of biomarker data results.

As with other scientific domains, cancer biomarker research today involves the collection and processing of significant quantities of data (recall Figure 1 from Section 3), as well as the assimilation of diverse information from many disparate sources for investigation and analysis (dealing with architectural principles P1-P3 from Table 1). What often distinguishes research in cancer biomarkers, however, is the heterogeneity of the information that its researchers must interact with (related to architectural principle P3 from Table 1). Everything from clinical studies, peer-reviewed publications, statistical data sets, imagery, and human and animal tissue samples can contribute something of value to the overall research picture. With so much technological progress having been made in recent years, however, investigators are increasingly finding themselves awash in data and faced with increasingly acute pressure to efficiently extract the signal from the noise. As a result, tools for managing and understanding this data have become critical to providing researchers with the ability to efficiently and reliably obtain, process, preserve, and publish research results.

Specimen tracking and query systems, relational models for biomarker data, literature search engines, and data warehousing technology for long-term secure storage and statistical analysis are concepts whose implementations have been around in one form or another for several years. The pressing challenge today is in the integration these tools and the data they contain into a seamlessly connected, multi-institution research platform to support the increasingly collaborative efforts of modern research scientists (dealing with location transparency as highlighted in architectural principle P2 from Table 1).

The EDRN is an excellent example of an e-science virtual organization. Its research is a coordinated effort by many distributed participants to join forces in attacking the complex and multi-faceted problem of early detection of cancer. The viability of the EDRN model, where distributed participants collaborate and share data seamlessly, is predicated on the existence of a technology infrastructure capable of supporting domain-specific distributed research efforts. Such infrastructures are an example of science and technology working hand-in-hand to achieve results that would have been impossible to attain using a traditional, monolithic approach. E-science virtual organizations promote collaboration, and EDRN is no different. It was conceived with the understanding that none of its members had the requisite human, material, or financial resources to take on the challenge of finding new biological indicators for the early detection of cancer alone. Collaboration, however, provides an avenue for subdividing the problem, and targeting the resources and expertise of each individual institution for maximum effect (architectural principles P1 and P2 from Table 1).

Recognizing this, the EDRN has consistently placed a strong emphasis on the role of technology in helping to alleviate the technical, scientific, management, and policy challenges associated with conducting large-scale distributed scientific research. The development of an informatics infrastructure to promote the coordination of efforts and the sharing of research results has been a cornerstone of the organization's success.

Since the EDRN's inception, JPL has played a central role in the development of an enterprise-wide informatics infrastructure for the EDRN, focused on a number of concrete goals, and designed to provide a sturdy technological platform for the

EDRN's distributed research efforts. Leveraging lessons learned from several of the planetary and earth science data systems discussed earlier, and taking into account the unique needs of researchers in the cancer biomarker domain, JPL leveraged the OODT software product line to develop a distributed research grid of tools and services that collectively came to be known as the EDRN Knowledge Environment, or EKE.

Utilizing OODT provided us with a strong base anchored in the principles of distributed information representation and sharing. The layered services approach used in the planetary and Earth domains was again leveraged here to develop domain specific extensions to the core services as well as data-type specific tools for high-fidelity data analysis and interpretation (similar to the example in Section 5 from planetary where planetary specific data services were developed on top of those discussed in Section 3 and in Figure 1).

The domain information model for EKE consisted of two key components: a semantic ontology, which described classes of information objects in the domain and explicitly mapped their relationships to one another; and a "data dictionary" consisting of terms whose definition had been agreed upon and that could be counted upon to have a shared interpretation across institutional boundaries (architectural principle P3 from Table 1).

Each component of EKE was designed to be in conformance with the EDRN domain information model. The fact that OODT was architected with an "Expect the Unexpected" mantra (as described in Section 3) was particularly valuable to us in this implementation as the domain information model expanded and evolved many times in a variety of directions that would have been very difficult to predict a priori.

Similar to planetary science, having a well-defined domain information model to guide development of the EKE infrastructure and tools was absolutely critical (as noted in architectural principle P3 from Table 1). Due to the open-ended design of the underlying OODT architecture, the natural evolution of the domain model did not pose a threat to the integrity of the infrastructure. On the contrary, the presence of a guiding model, even one in occasional flux, proved crucial to rationalizing implementation decisions in the context of the domain, and maintaining sanity in the face of integrating technologically and geographically diverse systems into a unified virtual scientific environment.

The EDRN Knowledge Environment was built around the now familiar principle of loosely connected components capable of communicating among one another by virtue of a shared information model. Rather than a traditional, monolithic stack of applications and services tied to a particular technology set physically installed into a single centralized location, this architecture permitted the development of an ecosystem of applications and service endpoints that were physically located near the data they manipulated, and yet transparently accessible from anywhere via the grid, dealing with architectural principles P1 and P2 from Table 1.

Although several of the component applications of OODT have been introduced earlier in the chapter (recall section 3), a few merit more detailed discussion in the context of their ability to break down institutional barriers to data discovery and sharing and truly enable distributed scientific research.

The EDRN Resource Network Exchange, or ERNE, was one of the EDRN's early success stories. Designed as a distributed specimen query system, ERNE leveraged OODT's product and profile server architecture to provide unified query access to the numerous specimen repositories located at EDRN member sites (see Figure 8 for a detailed view of this architecture). Prior to ERNE, a centralized query mechanism for specimens did not exist and there was no way for a researcher to reliably know with any certainty that he or she had a comprehensive understanding of specimen availability, short of actually contacting each site individually to inquire.

With the help of the Common Data Elements from the domain information model, it was possible to determine a set of data attributes that would be able to adequately describe specimen resources. However, because the specimen repository information systems at each of the sites were technologically heterogeneous, querying all of them in a unified manner meant the need for site-specific translations or mappings between the ERNE query based on EDRN CDEs and the site-specific naming conventions in place at each repository.

By placing OODT product server software at each site and working with sites to develop the requisite mapping, it was possible to develop ERNE in a way that allowed for unified query access to the distributed specimen repositories without perturbing the host site's internal data model or operating procedures. As a result, ERNE queries run from the web-based query interface return a unified picture of the matching specimen resources available at each of the participating EDRN sites. As of this writing, ERNE had connected specimen resources at thirteen different sites around the US, totaling over a quarter million specimens.

The type of location-independent access (recall architectural principle P2 from Table 1) to data embodied by ERNE has been one of the overarching tenants of the EDRN's informatics infrastructure. Another way that EKE provides researchers with a truly virtual scientific environment is by seamlessly integrating with external (non-EDRN) data sources. The EDRN, while ambitious, is relatively small, and relatively young compared with similar organizations worldwide. EDRN recognized early on that collaboration, not only among its member sites, but also between itself and the myriad other international efforts at combating cancer through research, would be highly valuable to its research community. With that in mind, the EDRN has developed its Biomarker Database application [33] to flexibly integrate links to resources and content physically housed and cataloged in repositories external to the EDRN itself.

The Biomarker Database is an attempt to provide researchers with a unified picture of the state-of-the-art for research on particular biomarkers. This curated resource provides access to annotated information from a wide variety of sources some within and some external to the EDRN itself. Because of the flexibility of the EDRN domain information model, and the architecturally supported abstraction of the physical location of data from an to end user perspective, the EDRN Biomarker Database has attracted attention for its ability to quickly provide researchers with context about ongoing and past research efforts related to a particular biomarker.

In the course of carrying out its research, the EDRN generates a considerable amount of data. Some of this data is "raw", and some has undergone various processing steps to transform it into an informational resource. While sharing information is central to EDRN's mission, it also aims to preserve its research assets, thereby organizing them into a long-term, national resource that can be leveraged to aid future research efforts.

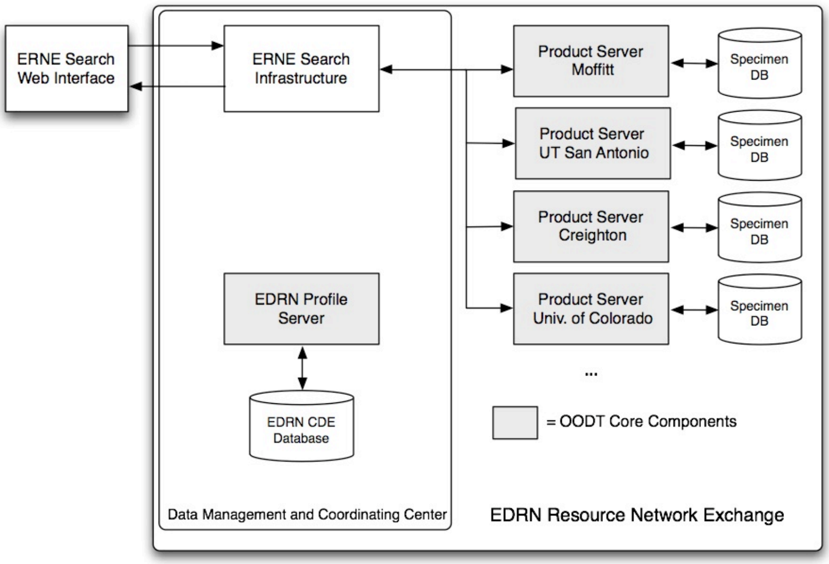


Figure 8. The EDRN Resource Network Exchange (ERNE) deployment for EDRN. In the diagram, OODT services (product servers and profile servers) implement the functions of data discovery, access, and distribution.

The EDRN Catalog and Archive Service, or eCAS, provides a data warehousing capability that is central to providing long-term, secure storage of research datasets. The system enables data generated from across the EDRN enterprise to be added to the archive, while associated meta-data is extracted, reviewed for quality, and indexed to provide a semantically rich catalog of the information assets stored in the repository.

The EDRN's data holdings are numerous, varied, and highly distributed as shown in Figure 9. The EDRN recognized that providing centralized access to the accumulated knowledge would be key to promoting its efforts and increasing the value of the research results by facilitating the degree to which they could be discovered, understood, and utilized. JPL developed a dynamic portal interface to provide access to resources from across the EDRN enterprise from a single, centralized web interface. Because the EKE components and services each adhere to the EDRN domain information model (architectural principle P3 from Table 1), the relationships between EDRN data are consistent and predictable. Furthermore, EKE has centered on the use of Resource Description Format (RDF) [24] to provide text-based semantic representations of the data that can be passed between applications as necessary. By analyzing and aggregating RDF streams from each of the EKE components, the EDRN Public Portal is able to consistently provide up-to-date, richly annotated information that communicates the full extent of the resources available through the EDRN.

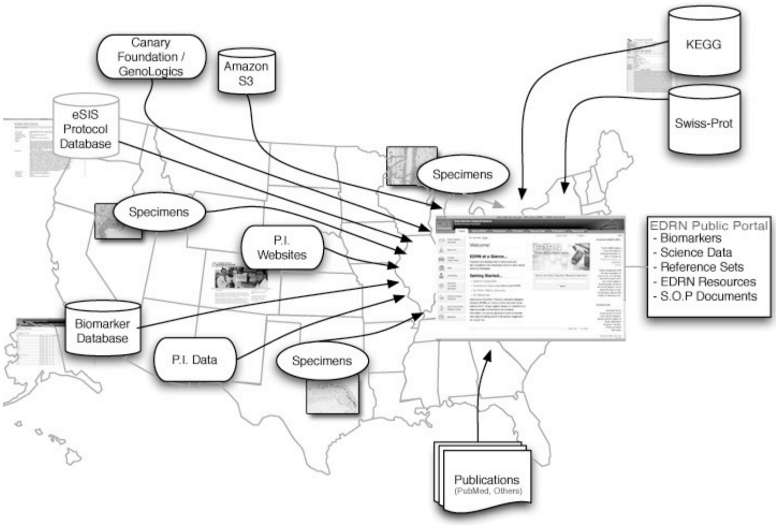


Figure 9. The geographic distribution of the Early Detection Research Network and the variety of data managed in the e-science enterprise. The distributed data in the system is linked together via a semantic portal (in the center of the diagram, unifying the view of research and progress within the EDRN.

The EDRN Knowledge Environment provides a virtual scientific environment, a technology platform which supports the EDRN's core efforts to collect, organize, process, and share the vast amounts of critical research it conducts on a daily basis. The informatics infrastructure forms a comprehensive, architecturally principled, and pragmatic approach to supporting cancer biomarker research through tools and interfaces, which, though each may be distributed, are linked to one another through a common information model, and capable of bi-directional communication via the grid. JPL has leveraged the e-science reference architecture promoted via OODT, deconstructing the process of biomarker research into a set of functions, and providing a layered system with applications on top of a core set of services to enable the logical integration of EDRN data. Furthermore, by integrating these domain-specific applications into an enterprise system, the informatics infrastructure enables EDRN as a national organization to provide the capability for managing the biomarker information assets at a national level.

The Early Detection Research Network (EDRN) is an excellent example of an e-science infrastructure for cancer research. The EDRN has been an important pathfinder to pioneer the use of informatics to deploy a distributed, model-driven architecture across geographically distributed cancer research laboratories. Our experience within EDRN confirmed our belief that a well-defined information model is critical to linking distributed, heterogeneous data systems together. The early work of developing a common information model that could be embedded within a distributed software service framework such as OODT, quickly transformed the EDRN from a set of independent research laboratories into an integrated knowledge system where various data such as scientific datasets, biospecimens, study information, etc could all be accessed and shared. Efforts to build and identify system architecture helped to provide a scalable and extensible architecture that has allowed for new services to be added As a result, the EDRN has become a recognized e-science model for the cancer biomarker research community [4, 5].

Related Work

A considerable amount of work has been done in advancing the principles of e-science and applying them to the construction of systems across the full spectrum of scientific research. The fact that these principles are so broadly applicable speaks to the power of the approach. The contributions to the field are too numerous to cover in detail here, but we present a selection of e-science efforts that specifically relate to the development of virtual scientific environments for carrying out distributed research.

De Roure et al [34] have addressed the issues related to applying a semantic layer to the traditional e-science grid concepts in an effort to add increased richness to the communication options available to e-scientists. De Roure shares in the vision of an infrastructure that achieves its goals through pragmatic decomposition of the problem into modular components that share a common communication methodology. In particular, he promulgates a scenario involving a *service oriented* approach to building the e-science infrastructure, laying out in great detail both the advantages and the research challenges inherent to this approach. Emphasizing the importance of the "knowledge layer" in the construction of e-science infrastructures, De Roure further provides a roadmap of sorts, in the form of categorized research challenges, for moving from the present state of the art to a more comprehensive, semantically rich e-science environment.

Hey and Trefethen [35] describe large-scale efforts in the UK at building an e-Science infrastructure, including an e-science grid testbed, to support research in multiple scientific domains. Motivated in part by the increasingly data-intensive work being carried out in European research facilities like the Large Hadron Collider (LHC), which is expected to generate on the order of petabytes of data annually, the program aims to leverage the power of the grid to support the access and analysis needs of scientists the world over. Hey references NASA's Information Power Grid (IPG) project as an "existence proof", and outlines the technical details of the UK's plan along with short- and long-term challenges, strongly emphasizing the need for international collaboration to ensure that the value of the effort is not constrained.

Yang et al [36] provide a brief examination of e-science infrastructure interoperability, taking a key concept that initially fueled the rise of grid-based e-science systems and applying it to those systems themselves. Yang concludes that while the systems surveyed have each made significant strides in connecting their respective research cohorts, the middleware upon which these systems are built are for the most part not yet interoperable with one another. Yang argues that integrating these e-science initiatives will become increasingly important and should be the next step in the evolution of increasingly interconnected global scientific research.

Conclusion

The e-science paradigm is only increasing. The infrastructures that are being built around the world are changing the way in which science is performed. No longer is science constrained by the boundaries of a local laboratory. It is being conducted across geographical, organizational and political boundaries allowing for world-wide collaboration among scientific researchers. As a result a focused software architecture approach is critical to supporting the ambitious goals of building virtual science environments, by integrating distributed organizations.

In this chapter, we have described an architectural approach, a set of principles, an information model and associated implementation framework that bridges the gap, allowing reuse of software and information architecture across scientific domains. Specifically, we have described the OODT architecture and implementation, used to build widely successful e-science applications in the areas of planetary science, earth science, and cancer research. We have addressed issues of data capture/curation, processing, dissemination and preservation in each of these heterogeneous application domains using OODT as the linchpin upon which domain-specific information models and software are constructed.

While our work to date has been highly successful, a number of pertinent research questions remain. Our current work is focused on the areas of analysis of distributed data sets, large-scale, wide-area data movement, and in the areas of cloud computing, each which we believe fit within the architectural paradigms of e-science systems. We expect the focus on information and software architecture, OODT's principle foundations, to aid our efforts and help make a strong contribution to each of these emerging areas.

Acknowledgement

The research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

1. M. Cook. *Building Enterprise Information Architectures: Reengineering Information Systems.* Prentice-Hall, 1996.
2. D. Crichton, J.S. Hughes, S. Kelly and J. Hyon. "Science Search and Retrieval using XML". *In Proceedings of the 2nd National Conference on Scientific and Technical Data*, Washington D.C., National Academy of Sciences. March 2000. http://oodt.jpl.nasa.gov/doc/papers/codata/paper.pdf
3. D. Crichton, J.S. Hughes and S. Kelly. "A Science Data System Architecture for Information Retrieval". In *Clustering and Information Retrieval.* Kluwer Academic Publishers. December 2003.
4. D. Crichton, et al., "Creating a National Virtual Knowledge Environment for Proteomics and Information Management," in Informatics and Proteomics: Marcel Dekker Publishers, 2005.

5. D. Crichton, S. Kelly, C. Mattmann, Q. Xiao, J. S. Hughes, J. Oh, M. Thornquist, D. Johnsey, S. Srivastava, L. Esserman, and B. Bigbee. “A Distributed Information Services Architecture to Support Biomarker Discovery in Early Detection of Cancer”. Accepted for publication at the *2nd IEEE International Conference on e-Science and Grid Computing*, Amsterdam, the Netherlands, December 4th-6th, 2006.
6. D. Crichton, et al., "Facilitating Climate Modeling Research and Analysis via the Climate Data eXchange," In Proc. Workshop on Global Organization for Earth System Science Portals (GO-ESSP), Seattle, WA, 2008.
7. J. 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. J. S. Hughes, et al., "Intelligent Resource Discovery using Ontology-based Resource Profiles," Data Science Journal, 2005.
9. Y. Tina Lee (1999). "Information modeling from design to implementation" National Institute of Standards and Technology.
10. C. Mattmann, D. Crichton, N. Medvidovic and S. Hughes. “A Software Architecture-Based Framework for Highly Distributed and Data Intensive Scientific Applications”. In Proceedings of the *28th International Conference on Software Engineering (ICSE06)*, pp. 721-730, Shanghai, China, May 20th-28th, 2006.
11. C. Mattmann, et al., "A Reusable Process Control System Framework for the Orbiting Carbon Observatory and NPP Sounder PEATE missions," in Submitted to 3rd IEEE Intl’ Conference on Space Mission Challenges for Information Technology (SMC-IT 2009), 2009.
12. "Reference Model for an Open Archival Information System (OAIS)," CCSDS 650.0-B-1, 2002.
13. R. N. Taylor, N. Medvidovic and E. Dashofy. *Software Architecture: Foundations, Theory and Practice*. Wiley Press, 2009.
14. M. Uschold and G. M., "Ontologies and Semantics for Seamless Connectivity," SIGMOD Record, vol. 33, 2004.
15. Apache Tika. http://lucene.apache.org/tika/, 2010.
16. ISO/IEC CD 11179–3 Information Technology – Data Management and Interchange – Metadata Registries (MDR) – Part 3: Registry Metamodel (MDR3) (2002). http://www.jtc1sc32.org/sc32/jtc1sc32.nsf/Attachments/00DEC39D41D17B1288256A5300603FED
17. S. Weibel, J. Kunze, C. Lagoze, M. Wolf. Dublin Core Metadata for Resource Discovery. Internet Engineering Task Force RFC, 1998.
18. P. Cornillon, J. Gallagher, and T. Sgouros. Opendap: Accessing data in a distributed, heterogeneous environment. *Data Science Journal*, 2:164–174, 2003.
19. Apache Lucene, http://lucene.apache.org/, 2010.
20. X. Yang, L. Wang, G. von Laszewski. Recent Research Advances in e-Science. *Cluster Computing*, vol. 12, pp. 353-356, 2009.
21. I. Gorton, P. Greenfield, A. Szalay and R. Williams. Data-Intensive Computing in the 21st Century. *IEEE Computer*, vol. 41, no. 4., p. 30, 2008.
22. R. T. Kouzes, G. A. Anderson, S. T. Elbert, I. Gorton, and D. K. Gracio. The changing paradigm of data-intensive computing. *IEEE Computer*, vol. 42, no. 1, pp. 26-34, 2009.
23. S. S. Laurent, J. Johnston and E. Dumbill. *Programming web services with XML-RPC*. O’Reilly Media, 2001.
24. O. Lassila and R. R. Swick. Resource description framework (RDF) model and syntax, *World Wide Web Consortium*, http://www. w3. org/TR/WD-rdf-syntax, 2010.
25. R. Fielding and R. N. Taylor. Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology (TOIT)*, vol. 2., no. 2., pp. 115-150, 2002.
26. T. Berners-Lee and J. Hendler. Scientific publishing on the semantic web. *Nature*, vol. 410, pp. 1023-1024, 2001.

27. CODMAC, Data Management and Computation, Vol. 1: Issues and Recommendations. Committee on Data Management and Computation, Space Sciences Board. Assembly of Mathematical and Physical Sciences, National Research Council, 1982.
28. IPCC Intergovernmental Panel on Climate Change, http://www.ipcc.ch/, 2010.
29. C. Mattmann, A. Braverman, D. Crichton. Understanding Architectural Tradeoffs Necessary to Increase Climate Model Intercomparison Efficiency. *ACM SIGSOFT Software Engineering Notes*, vol. 35, no. 3, July 2010.
30. B Fortner. Hdf: The hierarchical data format. *Dr Dobb's J. Software Tools and Professional Programming*, 1998.
31. R. K. Rew and G. P. Davis. Netcdf: An interface for scientific data access. *IEEE Computer Graphics and Applications*, 10(4):76–82, 1990.
32. P. Cornillon, J. Gallagher, and T. Sgouros. Opendap: Accessing data in a distributed, heterogeneous environment. *Data Science Journal*, 2:164–174, 2003.
33. A. Hart, C. Mattmann, J. Tran, D. Crichton, H. Kincaid, J. S. Hughes, S. Kelly, K. Anton, D. Johnsey, C. Patriotis. Enabling Effective Curation of Cancer Biomarker Research Data. In *Proceedings of the 22nd IEEE International Symposium on Computer-Based Medical Systems (CBMS)*, Albuquerque, NM, August 3rd-4th, 2009.
34. D. De Roure, et al. The Semantic Grid: A Future e-Science Infrastructure. *Grid Computing: Making the Global Infrastructure a Reality*. John Wiley & Sons, Ltd. West Sussex. pp. 437-472, 2003.
35. T. Hey and A. Trefethen. The UK e-Science Core Programme and the Grid. *Computational Science*, vol. 2329/2002, pp. 3-21, 2002.
36. X. Yang, et al. Recent Advances in e-Science. *Cluster Computing*, vol. 12, pp. 353-356, 2009.
37. D. Crichton. Core Standards and Implementation of the International Planetary Data Alliance. *37th COSPAR Scientific Assembly*. vol. 37, pp. 600, 2008.