

1st Interoperability Design Challenge Workshop

Report from Workshop I

Prepared by: [John Helly / UCSD / SDSC](#), [Peter Cornillon / URI](#), [Tom Jordan / USC](#), [Tim Ahern / IRIS](#), [Ben Domenico / Unidata](#), [Reagan Moore / SDSC](#), [Dale Chayes / Lamont-Doherty](#), [Jim Frew / UCSB](#)

TABLE OF CONTENTS

EXECUTIVE SUMMARY	2
1.1 Types of Data	2
1.2 Supporting environments	2
1.2.1 Data distribution systems	3
1.2.2 Federation	3
1.2.3 Standard Operations Spanning Data Collections	3
1.2.4 Data Manipulation Services	5
1.2.5 Data preservation systems	5
1.2.6 Intellectual Property Management	5
1.3 Interoperability Interfaces	6
1.3.1 Metadata	6
1.3.2 Data	6
1.3.3 Conceptual Models	7
2 OTHER INTEROPERABILITY ACTIVITIES	7
2.1 NASA	7
2.2 NOAA	7
2.3 ESMF	7
2.4 Earthscope	7

EXECUTIVE SUMMARY

The challenge to the design teams is to describe the characteristics of the blue, circular outer layer in Figure 2 and, in doing so, to answer the following questions.

1. What features of current systems should be contained in a unifying interface definition?
2. What features are common or very similar across existing systems?
3. What features are unique to a disciplinary community or application?

We are taking a top-down and bottom-up approach. The top-down aspect is defined by the Level 0-2 terms of the Interface Definitions in Table 2. The bottom-up aspect is defined by the set of system descriptions provided by cognizant individuals for each system.

A key result of this first workshop, perhaps the key result, is the summarization of the results of the two breakout groups. This resolution was performed in plenary session on the afternoon of the second day. Each group presented its results and these can be found as an appendix to this report.

The broad purpose to be satisfied by all these systems and derived requirements are to acquire data of interest in the form desired, when it is wanted and with confidence in the content. Reproducible results are important, we must be able to identify all data used to create derived data, and the processing steps, and have sufficient assurance of quality for data (provenance) received by a user. To support this we have identified an number of sub-topics that require consideration.

Reproducibility

1.1 TYPES OF DATA

The two broad types of data that have been identified include streaming, real-time data that may be buffered in primary storage media and 2) data files. This is an important distinction that has important architectural significance. This difference has also led to the development of data systems that are designed to operate principally on data streams or principally on data files. This will continue to be an important design difference although we anticipate future systems to be built to accommodate both types of data and therefore hybrid-type systems will be needed. A further consideration was identified in the temporal dimension related to access latencies. For example, archived data may be stored with relatively short retrieval latency (i.e., near term) or longer latency (i.e., deep archive).

1.2 SUPPORTING ENVIRONMENTS

This category includes the entire range of functions that historically are referred to as data management systems. However, the emergence of highly integrated systems with services ranging across data acquisition to long-term archival and preservation services are not well-described by this term.

1.2.1 Data distribution systems

Data distribution systems provide user access to data. There is no preferred method of doing this at this time. What is clear, is that there are many functions and services that have been explored and the best we can do at this time is to list them for further consideration.

1. **Dynamic Data Systems:** These are dynamic data delivery systems for data that are not persistent. They generally employ primary and secondary storage systems. If the data are not obtained and processed by a user or client, for example written to files, they are lost. Examples include real-time sensors, streaming audio and video but are not limited to these. These methods are commonly employed in predictive and diagnostic nowcasting and forecasting of climate and weather, for example.
2. **Stored Data Systems:** These are systems that ingest, store, export and serve the data that are already in a persistently-stored form. They generally employ secondary and tertiary storage systems. These types of systems can be divided into file-oriented systems, such as operating system-level file systems and transaction-oriented system, such as database management systems (DBMS). These methodologies are frequently employed in digital libraries and most, common computing applications.

We consider this topic area to encompass the following architectural design topics: distribution versus centralization of data and metadata, context catalogues as well as the characteristics and services of storage management systems.

1.2.2 Federation

- ♣ Mechanism that supports links into remote resources
- ♣ Consistency of content behind link
- ♣ Access control for content behind link
- ♣ Ability to maintain control of data when federated access is provided
- ♣ Trusted mediators for access to data
- ♣ Management of versions of data
- ♣ Assertion about authenticity of data
 - Publication links to track resources using your data
- ♣ Shadow links to point to source of data
- ♣ Peer-to-peer federation mechanisms that allow resource hierarchies
- ♣ Support metadata consistency across caches
- ♣ Support cross-registration of data between collections

1.2.3 Standard Operations Spanning Data Collections

Data system interoperability is inherently about operation across disparate data collections and therefore the systems that support them. There must be a definite standardization that is jointly supported among the participating collections. This is central to the notion of federation that is also discussed below. Not surprisingly, we have

identified the search, browse and retrieval functions as fundamental and common to the systems investigated. Key to this capability and standardization is the need for a UID (Unique Identifier) that is a globally unique name for any given data object. We have also come to use the term ADO (arbitrary digital object) to refer to the digital objects stored within a collection.

1.2.3.1 Reproducibility of Results

- ♣ Notification to user of updates
- ♣ By collection or by element

1.2.3.2 Search

Certain fundamental queries should be supported resulting in lists of query results on entity metadata for the digital objects stored by the data system of interest. For example, it would be desirable to have a standard set of queries that can be universally satisfied:

- ♣ list of object UIDs by time and date, spatial location, data set type, themes, standard variable name, free-text
- ♣ Meta-information about metadata (what does an item name mean)
- ♣ statistical abstracts of collection structure.
- ♣ Do you have data of standard variable of type X?
- ♣ ADO provenance
- ♣ ADO item names (list of items)
- ♣ ADO size
- ♣ ADO relationships
- ♣ ADO Item formats (int., float)
- ♣ ADO physical quantities (units: feet, meters, degC)
- ♣ ADO standard variables (“temperature, depth”)

Such a commonality requires a common, minimum set of metadata attributes that are interpreted in a common way with a common, controlled vocabulary.

1.2.3.3 Browse functions

This function encompasses the ability to examine, in an interactive way (i.e., short latencies) a particular collection object or an object selected from a non-directed search.

1.2.3.4 Retrieve

We need to be able to retrieve both contextual information (i.e., metadata) as well as the data objects (i.e., ADOs) themselves. Multiple existing protocols support various approaches to this such as IP-to-IP connections but there are no general solutions to this protocol issue. Those that are used by the systems we reviewed and others known to us are listed below.

System	Data	Metadata
--------	------	----------

1.2.4 Data Manipulation Services

- a. Publication (i.e., upload or harvesting) of data and metadata into collection
 - i. Ingestion
 - 1. Manage updates
 - ii. Bulk loading
 - iii. User-defined collections
 - iv. Curation
 - v. Quality assurance
 - vi. Context creation
- b. Checkout (i.e., download) of data and metadata from collections.
- c. Data manipulation services
 - i. Example: subsetting, aggregation, transformations
 - ii. ADO access methods (subsetting, transformations) – ask for list of services
 - iii. ADO response formats
 - iv. Request notification of availability
 - v. Amount of data in collection
 - vi. Available services from systems
 - vii. Other ADO services – means ask for list of services
 - viii. Meta-information about metadata (what do item names mean) – metadata registry
 - ix. Extended examples: data fusion, data mining, transformations

1.2.5 Data preservation systems

Data preservation systems are those that focus primarily on the long-term preservation of data into the indefinite future. Systems such as these must account for the transient nature of digital encoding, formatting, hardware and software platforms. Very importantly is the growing need for migration strategies from present and past technologies to future technologies for the growing inventory of data to be preserved.

1.2.6 Intellectual Property Management

Intellectual property management (IP) management is an issue of broad concern across public and private data systems. It involves services related to user authentication and access control as well as issues of copyright, fraud, and precedence. There is no general implementation of the services required in this area.

1.3 INTEROPERABILITY INTERFACES

We are limiting discussion of interoperability interfaces to address data and metadata. We are assuming that the basic network interfaces exist and are reliable and that web-level protocols (e.g. http, SOAP, etc.) are in a high state of flux that will remain, for the indefinite future, in that state.

1.3.1 Metadata

1. Map context to a logical name space
2. Collection identifier – logical name space
 - i. Version identifier
3. **UIDs**
4. Globally unique identifier
 - i. Handle, Object ID, collection cross-correlation ID
5. Descriptive metadata
6. Provenance information
7. Physical name
 - i. File / sensor source for data stream / database record
8. Types of metadata
9. Administrative – physical location, access controls
10. Descriptive – discovery attributes
11. Authenticity – checksums, audit trails
12. Structural – format type, components
13. Behavioral – supported operations
14. Naming indirection mechanisms
15. Support mapping from URI to URL
16. Support organization of name into collection hierarchy
17. Import/Export of metadata formats
2. Import/Export of data formats
1. Standard APIs to execute operations
2. Ability to take data from multiple heterogeneous sources
3. Access Interfaces
 - a. Interactive interface
 - i. By collection or by element
 - i. Retrieve content (digital entities)
 - b. Programming interface
 - i. Required for interoperability
 - ii. Amount of data requested/found
 - iii. ADO retrieval
 - iv. ADO set (found via identifier mask)
 - c. Management of data flow across services

1.3.2 Data

- b. Consistent description of data models
- c. Discipline approved data models

1.3.3 Conceptual Models

- d. Syntactic models
- e. Semantic model

2 OTHER INTEROPERABILITY ACTIVITIES

In the final hour of the Interop Workshop we talked briefly about a "standards task force". I wanted to make sure that you were aware of the excellent foundation that has been provided by the NASA SEEDS group in this regard. See <http://lennier.gsfc.nasa.gov/seeds/stdprocRpt1.htm> (particularly sections 2 and 3).

2.1 NASA

2.2 NOAA

2.3 ESMF

2.4 EARTHSCOPE