Comments on Paris paper

**Layers of Enabling Technology**

A better approach is to differentiate between the entities that a data fabric manages and the services applied to the entities. A data fabric should manage the properties of:

* Collections
* Workflows
* Data Flows

independently of the choice of hardware and software technologies. The Data Fabric manages the operations on these entities in a distributed environment, as the underlying technologies change. These requires a Data Fabric that can evolve over time, by “plugging-in” interfaces to each new technology. The corresponding layers are:

* Management of interactions with clients. Each community will require the ability to use their domain client for accessing entities within the Data Fabric.
* Management of the properties of Collections, Workflows, and Data Flows. A typical application is a research collaboration that needs to control access and analysis while generating new publications.
* Management of interactions with technology. The operations performed by the research community may require transformation to new protocols for execution.

This is a three-layer architecture that separates management of technology evolution from management of digital entities from management of client interactions.

**Data Management Commons**

Software middleware currently works as applications installed at each site. There are current efforts to migrate middleware into:

* Storage controllers. Storage vendors want to migrate from storage provisioning to data management provisioning. This means each new disk will automatically come with an installed Data Fabric that supports collection and workflows.
* Internet2. The Internet community is pursuing the extension of network capabilities to include network-based services. An example is the provision of a content-delivery system within the network. This will enable the virtualization of data flows, the dynamic construction of virtual private networks, caching within the network, and support for operations within the network.

These two transitions will enable the formation of Data Fabrics that are inherently distributed across international infrastructure. The transition rely upon the ability to encapsulate domain-dependent capabilities into plug-ins that can be dynamically added to the Data Fabric infrastructure.

**Central role for PIDs**

The choice of PID depends upon the operation that will be applied. Multiple versions of PIDs are in production use:

* GUIDS. Provide globally unique identifier.
* Handle. Provide unique identifier (relative to resolver) and location.
* Tickets. Provide unique identifier, location, and access control
* Logical name. Provide unique identifier, location, access control, arrangement, and descriptive metadata.

**Trusted Repositories**

The ISO 16363 standard provides several hundred requirements for trusted repositories. The enforcement of these requirements can be a component of the Data Fabric. The number of policies that are needed is about 170.

**Logical Store**

The formation of logical collections is well understood. In data grids it is possible to map from the physical location to a logical arrangement that resembles a physical file system. Schema indirection is used to manage arbitrary descriptive and provenance attributes for each digital object. Queries can be made on the metadata attributes to form “virtual” collections, that can be listed and manipulated as if they were in a file system.

This means we need to differentiate between:

* Physical file systems
* Logical file systems (map organization structure on top of distributed data)
* Virtual file systems (dynamically created by queries on metadata as a browsable representation of distributed data)

**Automatic Workflows**

There are multiple examples of production systems that automate the application of workflows. These examples include:

* The iPlant Collaborative. A logical collection is manipulated by registered workflows that can be executed either where the data reside or on High Performance Computers. The user can select between workflows that are applied, create new workflows, and register the workflows for use by other persons.
* Hydroshare. A workflow is provided that automates the analysis of watershed hydrology analyses. This includes the identification of the watershed extent, the acquisition of input variables from federal repositories (DEM, soil, wind, temperature, precipitation, roads and dams), executes the analysis, and archives the results.

Data grids provide a way to automate provenance tracking. This makes it possible to share workflows, re-execute workflows, and compare results to test reproducible research.

**Federations**

Federation requires mechanisms for interoperability. There are a wide variety of protocols used by infrastructure components (authentication systems, storage systems, databases, transport, rule languages, operations). Three interoperability mechanisms are in production use:

* Tightly coupled mechanism. This involves the establishment of trust between two systems, enabling members of the first system to apply operations in the second system.
* Loosely coupled mechanism. This involves accessing the remote system using the protocol of the remote system, discovering desired data, and retrieving a copy. A variant of this is the encapsulating of the service provided by the remote system for execution at the local system on local data.
* Asynchronous mechanism. This involves posting a message to a queue for retrieval and processing by an independent system. This is the purest form of interoperability, since the two collaborating system do not directly interact. A variant of this is to use a broker to manage the interactions between two systems.

**Principles**

The approach that is in production use is the use of virtualization to ensure that properties desired for a collection can be managed and maintained independently of the choice of technology. Virtualization means that all required properties, and operations upon the digital objects, can be controlled within the Data Fabric. In turn, the Data Fabric maps the desired properties and operations to the capabilities of the underlying technology.

An implication is that the Data Fabric will evolve over time as new technologies are plugged into the infrastructure. Both the properties that are maintained, and the operations that are performed will evolve.

**Technical Components**

Collaboration environment. This is the infrastructure that enables sharing of data, workflows, and metadata while enforcing access controls.

Workflow management. This is the sharing of workflows, tracking of workflow input, tracking of workflow output, naming of workflows and outputs, re-execution of workflows. The goal is reproducible research.

Data Flow management. This is the ability to describe the sources, destination, mapping to disjoint parallel I/O paths, caching, access controls, and naming related to required data movement. A data flow characterizes the data movement independently of the choice of network protocol.

**Organizational Approaches**

Data Fabrics are domain dependent. Even though generic infrastructure can be implemented that supports data management operations, each discipline will require unique:

* Data formats
* Metadata
* Operations
* Clients
* Policies

The application of a Data Fabric to a domain requires the ability to plug-in the specific technologies needed for each discipline. This capability is in production use within data grids, with the same underlying infrastructure supporting projects in:

* Medicine
* Social Science
* Genomics
* Storm surge modeling
* Hydrology watershed
* Plant genomics
* Astronomy
* Neutrino physics
* Astrophysics
* High energy physics
* Seismology
* Environmental science
* Climate
* Digital libraries
* GENI
* Archives

Each discipline plugged-in the specific capabilities that they needed into the generic infrastructure.

Appendix:

When differentiated services are implemented for each desired property for digital objects, maintaining coherence between the services becomes very difficult. The integration of the differentiated services requires the ability to enforce policies that span the separate technologies. Thus if a service moves a data set, a policy enforces the update of the location in a metadata catalog. If a version is made, metadata is generated that automatically links the version to an original. The coordination between the services has to be flexible. An example is the UK e-Science data grid. They required the ability to prevent archived data from being deleted, even by an administrator. However, they also needed the ability (for some collections) for an administrator to replace corrupted data. They also needed the ability for users to delete files at will. The ability to change the policy that is applied (coordinate between services) must be supported by the Data Fabric.