

RDA Initiative

Data Foundation and Terminology

Basic Concept Note

Major Version 7

Contributors: Stan Ahalt, Daan Broeder, Stefan Heinzl, Bob Kahn, Larry Lannom, Michael Lautenschlager, Reagan Moore, Arcot Rajasekar, Johannes Reetz, Ulrich Schwardmann, Rainer Stotzka, Tobias Weigel, Peter Wittenburg

Version Update Note

- This major version 7 is being written after the Case Statement has been formulated and after we had several interactions with interested people.
 - Its scope is purely limited to basic terminology issues that are related to what we could call the **"Data Fabric"** which may be thought of as a data layer on top of the computer network layer defined, for example, by the Internet protocols. There are a number of efforts underway to add this new level of abstraction, including Digital Object Architecture, Data Centric Networking, and Named Data Networking. In order to address these issues in RDA we need to converge on terminology.
 - Some colleagues have suggested that the terminology group should also address interpretation and re-use for example - topics that are related to structural and semantic interoperability of the contents of data objects. These are very important but will need to be addressed in a new or different Working Group or a branch of this WG.
-

This concept note should not be confused with the Case Statement of the RDA terminology group, which has been submitted for council evaluation. This concept note was written by some of those who were preparing the suggested terminology WG topic to determine the focus and scope of what should be discussed and achieved. The Case Statement has been agreed upon by all people interested in the topic and participating in the WG so far. While the Case Statement at this moment is neutral with respect to the terms that need to be defined, this note is not. It makes some statements about terms, their underlying concepts and the relations among them. However, this version of the note is not meant to be a comprehensive document which elaborates all issues. It introduces terms which need to be defined in the working group.

Everyone else interested and basically in agreement with its content can contribute to this document and will be listed as contributor. There is no particular ownership other than the group mentioned at the top.

Rather than citing papers at different places in this note we refer to a number of papers which were at the basis of this note.

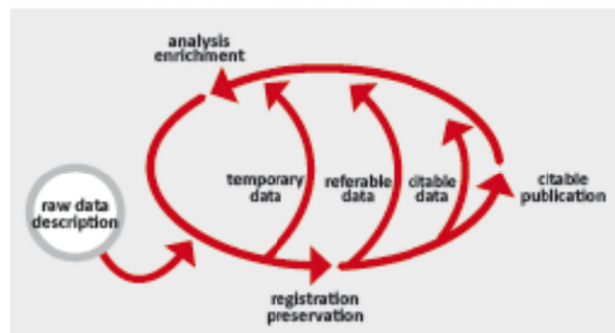
1. The Digital Era

A number of factors have changed dramatically in the transition from analog to digital information:

- Digital information is no longer bound to its medium, i.e. the content exists independent of any specific physical media.

- One of the key characteristics of physical objects, or the physical substrates, used for information storage is that long-term preservation requires them to be left alone to the greatest extent possible. That is, manipulating or using a device poses the risk of distorting or changing its physical properties and thus the information the device holds. For data objects¹ the reverse is true, i.e., most digital data needs to be checked and/or refreshed periodically in order to ensure their existence and their integrity. Further, data typically needs to be migrated and curated to keep up with technical innovation. Touching does in general not change integrity and quality.
- Thus, due to the fragility of the physical substrates that are used to house data objects, the data objects need to be replicated – hence the need for digital repositories that store all copies.
- Along the same line of reasoning, since we can readily separate digital content from its physical medium, we can transmit data via digital networks, i.e., data can be copied arbitrarily between any nodes on the Internet.
- Therefore, because of this separation of content from medium, there is no longer a direct physical relationship between a producer and a consumer of data. That is, all digital data is effectively anonymous, and this anonymity creates a trust gap and also an information gap regarding data interpretation.
- Accesses to digital objects and collections can be tracked, enabling validation and usage restrictions.
- Procedures used to manage and curate digital objects can and need to be tracked to be able to do appropriate interpretations.

Digital technology has also allowed us to create new types of data producers (sensors, simulation, analysis, crowd sourcing, etc.) that create ever-growing volumes of data and that increase the complexity (internal and external) of data sets.



In particular data complexity will require advanced automation in order to manage the data and provide access to the data. This is depicted in the diagram contained in Figure 1, taken from a DataCite/EPIC/Handle flyer. Citable publications are being created now in digital as well as non-digital versions and their number is increasing. Some data (collections) will be associated with publications and need also be citable. However digital technology allows us to create a huge amount of data as part of the automated workflows. In doing so millions of data objects are being created every day and they are being re-used for various purposes. Unfortunately, at their creation it is not clear whether any one or many of the data objects will be used in citable collections, subsequently the data objects need to be made referable, and thus managed.

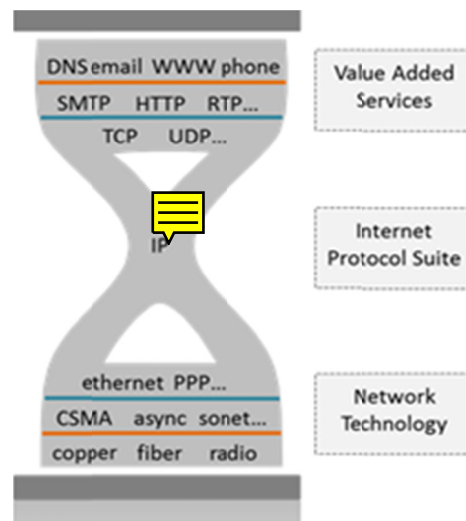
¹ In this document we use the term “data objects” taking that data objects are “digital objects”.

It is obvious that we are at the edge of dramatic changes again and that we need to understand the dimension of these changes. This issue is the raison d'être of the Research Data Alliance.

2. Domain of Explicitly Registered Data

In the area of networking it is widely accepted that we only acknowledge or use the collection of nodes that are registered and have a clear node identifier - e.g. an IP address. We know that there are many other nodes that send and receive messages, but they are out of the domain of the Internet and all its protocols and processes. We argue that, in the same fashion, in the future we will only be able to acknowledge and use those data objects and collections that are explicitly registered.

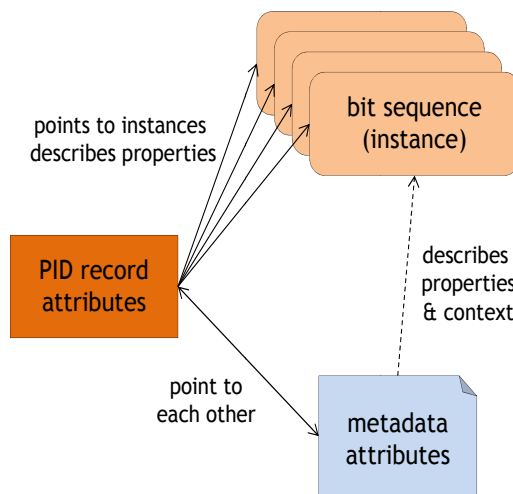
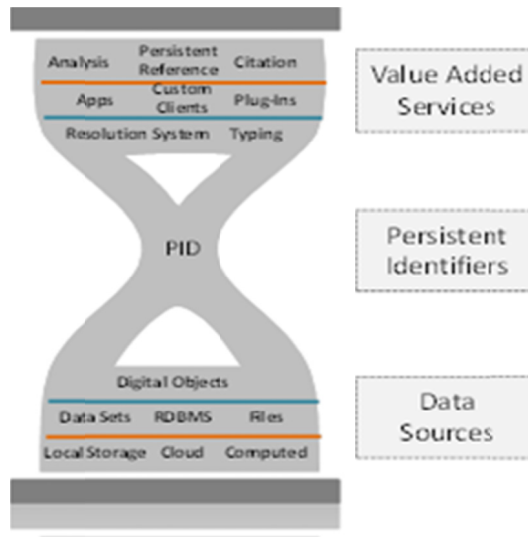
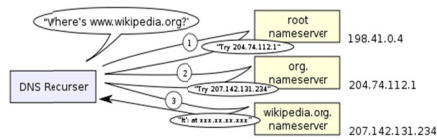
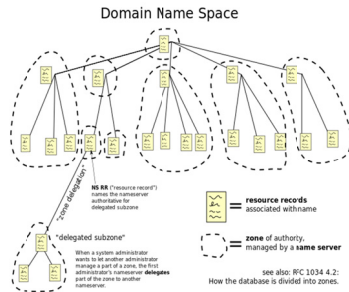
Of course, there will always be considerable data for which the state is completely unknown – it is not referenceable. Thus we will, increasingly, need to ignore those objects and collections which have not been registered explicitly in a widely agreed-to registry infrastructure. Each data object and collection in any domain needs to have an identifier issued by a worldwide accepted infrastructure for persistent identifiers. We believe that we will need to ignore, or consider non-existent, all other data when speaking about a (globally) accessible data domain. That is, since we cannot make any assertions regarding the existence, accessibility, sharability, integrity and quality of a non-referenceable data objects and collections, it ceases to be part of our shared discourse and our shared knowledge base.²



The above picture created by Dave Clark demonstrates the crucial role of IP numbers for exchanging messages of all sorts in a functional Internet. Of course the Internet makes use of various services for example to resolve “names” into IP addresses and vice versa. For data we can draw an analogous diagram (drawn by Larry Lannom). Here persistent identifiers are the pivotal means for referring to data objects and accessing them in a dynamic world of data. Analogous to the Internet world we need to maintain a second domain which is the domain of names and contextual attributes of various sorts. Persistent identifiers need to support references to the names, properties of objects and collections, and to the metadata³ - both PIDs and metadata contain valuable attributes describing properties of data objects and collections.


² We will need to develop much better tools as part of a data fabric where registration and metadata generation is being done automatically by the software we are using even at creation time.

³ The term metadata is used here in its restricted sense as machine-readable key-value type of descriptions of objects.

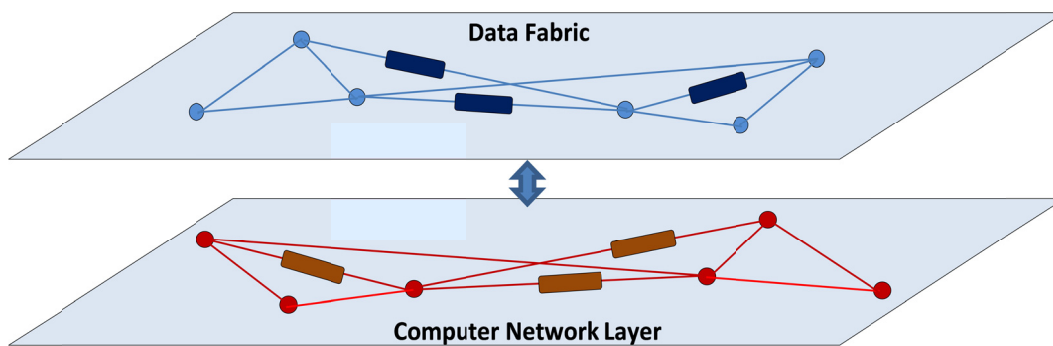


3. Data Fabric

On this basis we can make some assertions about the Data Fabric (DF):

- DF is a layer on top of any network of nodes in which packets are routed between arbitrary network nodes following protocols such as, but not restricted to, those defined by TCP/IP.
- DF is a virtual domain  er data objects and collections the properties of which are described by metadata and PID records.
- In DF registered Data Objects have attributes specified in open metadata descriptions and PID records are exchanged between data sources and consumers which are themselves registered with their own identities and certified as trusted entities. In many cases only metadata objects will be manipulated in LDN to reduce network load. It is LDN that will optimize data exchange in terms of caching, deploying algorithms close to the data etc.
- DF data object traffic is transformed to “packages” that are exchanged between computer network nodes. Currently this network layer is dominated by TCP/IP traffic, but other protocols can be thought of for more efficiently exchanging bulks of data. The data network depends upon, but is fundamentally independent of, the underlying network structures just as network protocols such as TCP/IP are independent of the lower level protocols upon which they depend as well as the physical media layer.

We can formulate that in the case of the Internet all applications are making use of the same basic protocol where the “packet” is the basic object being exchanged and where endpoints have addresses and names. In the case of data we describe a layer on top of the Internet where all applications are making use of the same basic protocol where “data” is the basic object and where PID and metadata attributes describe essential object properties.



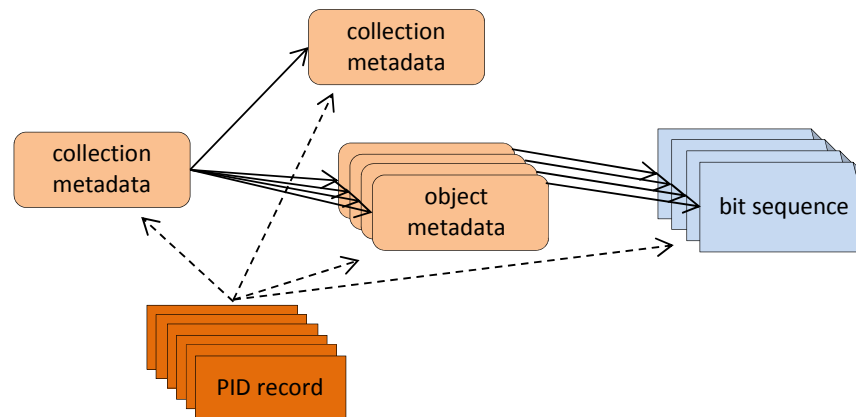
4. Data Objects and Collections

Basically Data Objects (DO) can be described as a triple:

- a sequence of bits⁴ stored in one or more repositories
- a PID that identifies the bit sequence, refers to its instances, and may store specific attributes about object properties and also may refer to a metadata object
- a metadata description⁵ that refers to the PID record and also stores specific attributes containing information about object properties

⁴ Sometimes experts speak about “bit sequences” indicating that a “bit sequence” may have an internal structure pointing to different fragments.

The metadata description itself is also a data object. Thus there can be an overlap between the attributes stored in the PID and in the metadata object such as typical citation information which might be extracted from the metadata object and which is included in the PID record to allow quick inspection. But in general the division is that PID records contain information that permits identifying the object, allows for a check on its integrity, allows essential data management processes, etc, while metadata records must contain contextual information (incl. provenance) that allows interpreting and re-using the data.



Collections are virtual aggregations of data objects that are identified and described by a PID and a metadata description. However, instead of referring to instances of bit sequences that contain the content, they refer to metadata descriptions of data objects or collections. So collections can include hierarchies of objects and collections that are related⁶. The type of relation is defined by the researchers, and is dependent on the objects' properties stored in the metadata descriptions or on the content. Since there is a wide agreement that **metadata must be open**, every researcher can in fact re-combine metadata objects and thus create new virtual collections without copying or moving the bit sequences. As indicated in the schematic drawing it is the metadata and the PID records that store all relevant information about collections. Thus, identifying collection member objects must be possible from the information stored in the PID record and the metadata description.

Often presentation formats and versions of a document are treated in special ways indicating the special relationships between these objects. At first instance they are just special collections where the specific types of relation is either expressed in attributes of the PID and/or of metadata records.

According to Abrams, Morrissey and Cramer (2009) objects can be categorized in 4 types:

- | | |
|---------------------------------|---|
| A. most simple object type | object consists of one file and has 1 format |
| B. format inclusion object type | object consists of one file and has m formats |
| C. split up object type | object consists of n files and has 1 format |
| D. most general object type | object consists of n files and has m formats |

⁵ Here we speak about a primary metadata description that is generated by the DO creation process. Obviously a variety of secondary MD descriptions will be generated for various purposes, partly extracted from the primary one, partly enriched with additional information, etc.

⁶ Thus collections differ from data objects in function, but in form are simply digital objects created for a specific purpose.

Here the **notion of “files”** needs a clarification. Digital data is always being delivered by making use of software layers. In general the operating system of a computer provides a file system that is able to obtain bit sequences from storage devices in a fairly robust way and hand over the bit sequence to some other software that transmits, renders, analyses etc. the bit sequence. Thus we are used to associating “bit sequences” stored on discs with files. But bit sequences can also be hidden in containers such as relational data base systems. In this case there will be a stack of database routines responsible to return the correct bit sequence, i.e. any external reference will not be translated into a file node, but into a single database query, for example⁷. Thus, the use of the term “file” above is intended to encompass both **structured** and unstructured data objects.

The 4 cases above can be translated into simple concepts in the data object domain:

Cases C and D speak about “n files” which simply means that we have a specific type of collection where the “type” information should be included in the metadata attributes. Indeed we need to consider the existence of data objects that have an internal structure whether it is indicated by different formats or by other means⁸. What we need is a mechanism to address “**parts of an object**”. This means that the general addressing scheme to access data objects will have the generic form:

<object identifier><delimiter><part identifier>⁹

The object identifier resolution mechanisms needs to be agnostic with respect to the part identifiers, since they can only be resolved by the local software stack dependent on some agreements about structure.

5. Persistent Identifiers

Persistent Identifiers (PIDs) are unique and persistent **things** that can be resolved into meaningful information about data objects. There is a wide **agreement** about PID systems¹⁰:

- In order to use PIDs in the manner summarized above we need to ensure that PIDs are registered externally with recognized registries¹¹.
- In order to be available for all repositories that store data in a persistent way we need to have one worldwide system of PID registration and resolution that performs well and is highly scalable, highly available and persistent.
- To be useful PIDs must provide a level of indirection to dynamic attributes that both change and are essential to access, for example, location.
- To be useful PIDs must be associated with attributes that describe external properties of the data object such as links to stores of instances of the bit sequence, a checksum, citation data, mutability flag, etc.¹²

6. Metadata

This section is not meant as a broad discussion about metadata, which is too large of a topic for a short consideration. Instead, this discussion serves only to fit metadata within the model we are proposing. Metadata is needed for various purposes, and the challenges of automatic data gathering

⁷ see also below for active data objects

⁸ A long time series recording (seismographic, video, etc.) can include a variety of different events that are relevant for the researchers as separate entities.

⁹ Part identifiers can be offsets, path descriptions, queries, etc. to refer to a fragment of the identified object

¹⁰ Some see the suitability of hierarchical PID systems where local and global registries serve different tasks.

¹¹ Many initiatives and institutes have their own internal PID system that serves particular needs, but this is not sufficient when speaking about a domain of registered data.

¹² There is a RDA initiative which will work out a recommendation for such information types

and processing, as well as finding or developing the appropriate tools to operate on data objects will extend the requirements on the metadata attributes that will need to be stored.

Metadata is used to store all information that is required to find objects and collections, and to interpret and to re-use objects and collections. In general we speak about storing properties and the spectrum of possible properties is fairly large, and dependent on what specific data production and consumption communities need and the functions they need to carry out¹³. It is important that metadata also includes contextual and provenance information that are important to interpret and re-use data.

What we can require is that metadata providers adhere to a few general principles in order to make it easier to manage and re-use the metadata descriptions for a variety of purposes:

- Schemas should be registered in open schema registries.
- Categories used to describe object properties, and appear in schemas, should be defined and registered in open data category registries.
- Each metadata description should be associated with a PID so that we can refer to the metadata description in a persistent way and include the metadata in the data management processes.
- Each metadata description should contain the PID that refers to the data object it describes.

7. Properties

Data Objects have a number of properties that describe their external and internal characteristics. Typical internal characteristics are properties such as the technical encoding and structure (often also summarized as format) used to encode scientifically relevant phenomena (for texts for example Unicode and a schema) and the semantics of the elements (categories) used to specify the phenomena. It is a widely accepted trend that there should be references to registries where encoding principles, schemas and also categories used are being registered and thus publically available. Only such explicit registration will enable seamless interpretation and re-use.

External properties characterize the data object as a whole as opposed to looking into its internals. However external properties are essential, for example, for the life cycle management of data objects and thus should be explicit, and harmonized. In general external properties can be crafted in discipline independent form, and thus can be harmonized across disciplines. Internal properties are usually discipline specific, although the underlying principles should be discipline independent.

There is not a generally accepted norm for which properties should be stored in the PID record and which in the metadata object. However, because the end objective requires us to reference both, in principal, it should just be a matter of developing suitable tools to access all information in a seamless way. DataCite for example defines “citation information” to be associated with PIDs, this is typically information that should be stored in the metadata records as well and obviously the citation information can be easily extracted from the metadata. This is an example of overlap in the information to be stored.

This working group leaves it to others groups to determine what are essential and common properties and their representation inside or outside PID record attributes.

¹³ A metadata initiative has been formed under the RDA umbrella that will work on metadata aspects.

8. Policy-Rule based Operation

Complementary to the notion of data objects and collections are operations that need to be carried out to support life cycle management, long term survival, access, interpretability and trust with respect to integrity and authenticity. Reagan Moore analyzed the scenario where explicit policies are designed to govern all operations on officially registered data objects in the emerging domain of complex data. Policies are designed to ensure the maintenance of properties of data objects and procedures are functions that implement these policies. Execution of sequences of such procedures results in state information that can be used for validation purposes. Thus we can state that explicit policy rules, proper procedures implementing them and a stable workflow engine executing sequences of such procedures are crucial for the trustworthiness of curation, preservation and accessible environments for data objects. It is a widely agreed trend that repositories that store data objects and collections will need to undergo quality assessments according to a specified procedure. The assessment of policy rules will be one of the most important aspects of such assessments.

Another working group has been set up to do address this issue.

9. Active Objects/Collections

Both digital objects and collections don't have to be restricted to static entities. The concept of active objects and active collections needs attention. The procedure that can be executed to generate a specific set of digital information can be registered. Accessing the registered procedure causes the information to be instantiated as would be the case when storing data in complex databases for example. A collection can then consist of persistent identifiers pointing to the procedures (active objects) that will generate information. An active collection associates a procedure with a collection. Accessing the collection causes the procedure to be applied to the digital information within the collection. An example is a "time-series" collection which organizes sensor data streams. When the collection is accessed, the **desired time sequence is** extracted through aggregation of individual sensor data files and partial I/O on the files that hold the end-points of the sensor stream¹⁴. Such procedures can be implemented as policies that control interaction with the collection allowing quality assessments.

In the same way metadata can be generated or extended by applying a procedure to digital data. The metadata then consists of the information generated by reification of knowledge relationships. Typical applications consist of state information that tracks the application of management procedures. It will be the task of another working group to identify the different types of metadata and to harmonize terminology and define procedures.

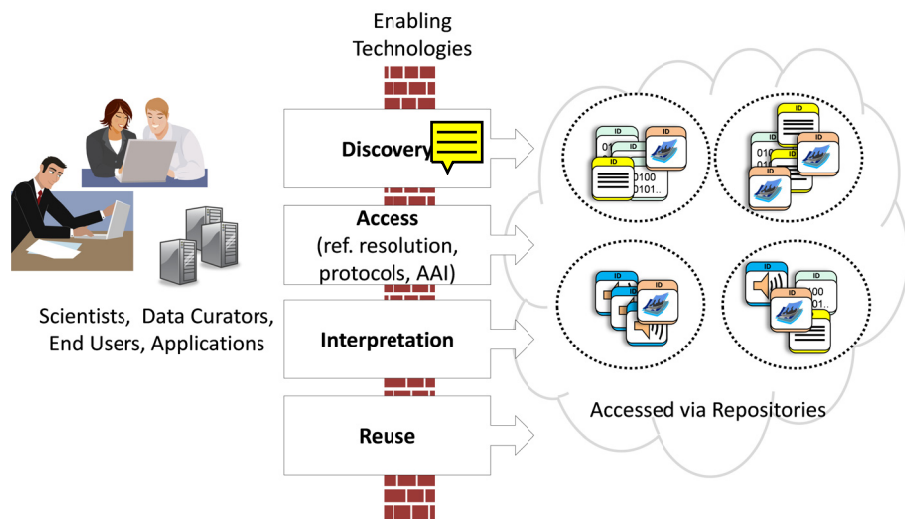
10. Data Access and Management


In this chapter we want to briefly indicate typical canonical workflows to indicate how different concepts which we are describing in this paper will be used. We are basing this on diagrams that were worked out by Larry Lannom.

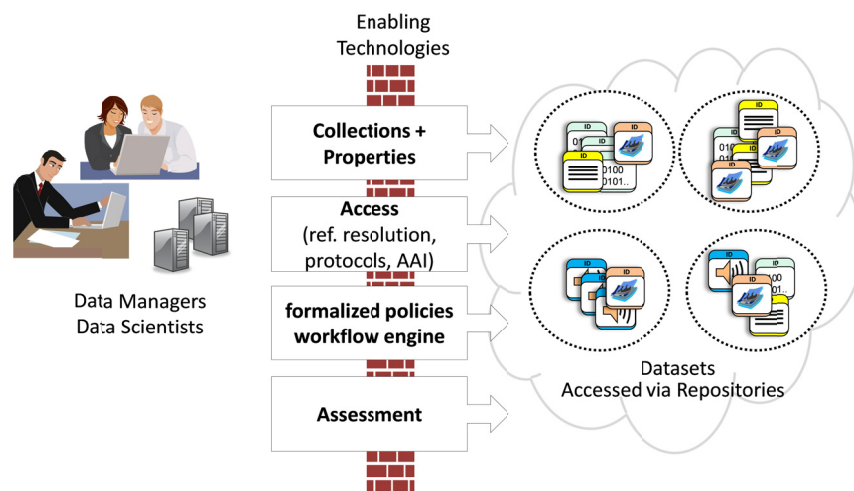
Consider one such canonical workflow for using data. Data users would first perform a metadata search to find useful objects or collections. The found metadata objects will yield the relevant persistent identifiers that amongst others allow a) quick inspections of at least a subset of the object's properties and b) initiating access to the data. The properties described in the metadata will allow users to determine whether the data object is possibly indeed what they are looking for and

¹⁴ This may also be achieved by specifying a collection and referring to fragments within the objects of the collection.

whether they will be able to interpret the content. The resolution machinery will look up the PID included in the metadata record, and this would also permit making use of the associated information to, for example, check the integrity of the object etc.



The PID will point to one of the instances of the bit sequence stored in some repository and start the access procedure. In general this will first lead to an authentication and authorization procedure, requiring a distributed mechanism to obtain the final access permissions. With the access permissions secured there will need to be protocols and software which will allow operations on the object's content. Re-using data, in general, requires more information about the data  just the information about the technical encoding, the syntax and the semantics. All relevant additional context information therefore should be included in the metadata record.



For typical management tasks we can also describe such a typical canonical workflow as indicated in the diagram above. In general management operations will be executed on collections of objects sharing some properties such as all video recordings encoded with H.264/MPEG4. Management operations can cover a whole spectrum of activities such as data migration, replication, transcoding, etc. Some of them just manipulate external properties; some of them will manipulate internal properties, for example when transcoding. The first steps in the diagram are basically similar although the first step will in general not be to execute a search on metadata, but instead prepared lists of collections that refer to their metadata. The access step is similar.

Due to the increasing amount of data only highly automated operations will be able to solve these types of management tasks. In specific cases policy rules might also invoke the actions of human agents. So policy rules need to be definable per collection and operation type. These will be executed, resulting in a manipulated set of data objects and their properties, i.e. the bit sequences may have been changed as well, the properties stored in the PID and metadata attributes may have been changed. In the case of changed bit sequences it may be necessary to also create new PIDs and metadata records depending on the data model being used.

More such typical workflows can be easily described and it will be the task of another working group to determine generalizations of what “data access” means. It will allow us to specify appropriate APIs and protocols.

11. Interoperability

IETF defines interoperability as the ability of two or more systems or components to exchange information and to use the information that has been exchanged. With respect to the exchange of data between two systems many different components at different layers are being involved as indicated in the figures above. The biggest interoperability challenge is to be able to interpret syntax and semantics of the content of the messages. At lower levels syntax and semantics will be defined by system architects, at the highest level syntax and semantics are defined by the community needs for the exchange scientific messages.

Much work has been done across and within communities (XML schema, RDF, OWL, domain ontologies, etc.) to define and register protocols, schemas and to formalize semantics - both helping to build bridges. This document will not discuss this domain in detail and thus does not make suggestions about terminology that should be described by the current terminology working group. However we see the need to also make progress in harmonizing this domain.

12. Relevant Terms to be Harmonized

In this chapter we present those terms that are essential for a data foundation and that should be harmonized. This list will change over time.

- Access Operation
- Active Object
- Active Collection
- Assessment
- Bit Sequence
- Carrier
- Category
- Citation
- Collection
- Complexity
- Content
- Curation
- Data Object
- Data Life Cycle
- Data Set
- Descriptive Information



- Metadata Knowledge Relationships
- Management
- Metadata
- Metadata Attributes
- Metadata Object
- Object Parts
-  Property
- PID
- PID Attributes
- PID record
- Policy
- Policy Rule
- Presentations
- Preservation
- Procedures
- Property - External
- Property - Internal
- Provenance Metadata
- Quality
- Registered Data Domain
- Replicas
- Repository
- Schema
- State Information
- Versions
- Virtual Collection
- Workflow

References:

Moore, R. W., "Automating Data Curation Processes", NSF workshop on "Curating for Quality", September 2012, Arlington, VA.

Robert Kahn, Robert Wilensky, "A framework for distributed digital object services", International Journal on Digital Libraries (2006) 6(2): 115–123

DOI 10.1007/s00799-005-0128-x

http://www.doi.org/topics/2006_05_02_Kahn_Framework.pdf

Robert Kahn, "An Open Architecture for Managing Information in the Internet", EUDAT Conference, Barcelona, Oct. 23, 2012;

<http://eudat.eu/system/files/B.%20Kahn.pdf>

Larry Lannom, "DAITF: Enabling Technologies", Pre-ICRI DAITF Workshop, Copenhagen, 21 March 2012;

<http://www.daitf.org/?p=48>

EPIC/DataCite/Handle Flyer: "High-Availability, Complementary Infrastructures for Persistent & Unique Identifiers for Data Objects & Published Collections based on Handle System", March 2012

various RDA initiatives: <http://forum.rd-alliance.org/viewforum.php?f=2>

DAITF Preparation-Note:

<http://www.daitf.org/wp-content/uploads/2012/06/DAITF-Preparation-Note-v3.pdf>

Stephen Abrams, Sheila Morrissey, Tom Cramer, "What? So What: The Next-Generation JHOVE2 Architecture for Format-Aware Characterization", 2009, Vol. 4, No. 3, pp. 123-136

[doi:10.2218/ijdc.v4i3.122](https://doi.org/10.2218/ijdc.v4i3.122)

Named Data Networking (NDN): <http://www.named-data.net/>