Open Archives Initiative – Object Reuse and Exchange

Compound Information Objects: The OAI-ORE Perspective

Carl Lagoze & Herbert Van de Sompel

March 27th 2007

1. Introduction and Motivation

Many information systems provide access to *compound information objects*[1] that are aggregations of distinct information units that when combined form a logical whole. Some examples of these are a digitized book that is an aggregation of chapters, where each chapter is an aggregation of scanned pages; a CD that is the aggregation of several audio tracks; an image object that is the aggregation of a high quality master, a medium quality derivative and a low quality thumbnail; a scholarly publication that is aggregation of text and supporting materials such as datasets, software tools, and video recordings of an experiment; and a multi-page web document with an HTML table of contents that points to multiple interlinked HTML individual pages.

Individual information systems vary in the manner that they configure, identify, and provide access to these compound objects. Because the web is the de facto platform for interoperability, and because web-based applications such as search engines have emerged as primary information portals, it is essential that these heterogeneous information systems project their compound objects in a standard manner onto the web. This *private* to *public* mapping can simply be done by associating an URI with each of the constituent units of information object, thereby making them web *resources*[2]. Using these URIs, web-information services and applications can obtain representations of the information units via protocol negotiation. For example, in such a private-to-public mapping, each scanned page of the book can get a HTTP URI assigned to it. The same can be done with each chapter and actually with the book itself. But the mapping can be more complex, for example, when the book needs to be made available for desktop computers, hand-held computers and cell phones. In this case each scanned page may be mapped to multiple URIs from which different views of the same scanned page are available.

---

[1] We will use the term *compound object* as shorthand for *compound information object* throughout this document.

[2] For the remainder of this document we use the terms *resources* and *representations* in the manner defined in the Web Architecture Document.

Such private-to-public mappings are common and help provide the source materials that make the web the rich information space it is. However, as is the case with many mappings between models or systems, this mapping is often lossy and ad hoc. Moreover, the mapping to the web is frequently targeted towards human users rather than machine agents, resulting in embeddings of the actual content of the information object in ”splash” pages, user interface “widgets” and the like. This mapping approach frequently leaves the essential structure of compound objects invisible to machine-based applications such as crawlers, search engines, and networked desktop applications.

Consider an example where all pages of a scanned book are assigned HTTP URIs. A crawler may subsequently land on any page, or individual URI. Depending on the individual private-to-public mapping the crawler may obtain from this URI a representation that contains links to other scanned pages of the same book, or to the containing chapter or book. The representation may also contain links to related resources that are not part of the book, for example to resources that provide information about the author, the publisher, etc. Unfortunately, the crawler or search engine cannot distinguish between these links, cannot interpret their semantics, and hence cannot understand which resources belong to the book and which do not, or what the relationship is among resources that are part of the book. Arguably, the result sets of crawler-based search engines would make more sense if components of a compound object were treated as a unit rather than as individual resources.

This problem arises from the fact that there is currently no standardized method for expressing semantics of links on the web. Web applications are therefore unable to distinguish which links point to resources that correspond to a specific compound information object and which to resources that are external to the object. In the public web the URIs of the components of the compound object exist only as a set of distinct resources from which a variety of representations are available. The notion of the compound information object as a logical whole with a distinct *boundary* is lost in the mapping.

This problem is acknowledged by the web community, and is described to some extent in the W3C Note On Linking Alternative Representations To Enable Discovery And Publishing. However, the W3C Note focuses on problems with expressing the relationships among resources that make available different versions of a specific information unit (e.g. various HTTP URIs available for the same scanned book page). This is a subclass of the problem described here, which is the relationship of resources that form a logical whole (e.g. various HTTP URIs available for scanned pages, chapters, book). Also, the Recommended Best Practices provided in the W3C Note are clearly of interest to the broader problem described here, yet are not specific enough serve as an implementable interoperability specification. A more thorough and expressive solution is required for communities such as the scholarly and educational communities, for whom compound objects are at the core of information communication.

A core goal of OAI-ORE – Object Reuse and Exchange – is to develop standardized, interoperable, and machine-readable mechanisms by which individual information systems can map and thereby expose private compound objects to the public web. These mechanisms will allow web applications to

reconstruct the logical boundaries of these compound objects, the relationships among their internal components, and their relationships to the other resources in the public web information space.

2. An OAI-ORE Interoperability Layer for Information Systems

Different information systems implement compound information objects in system-specific manners that vary according to:

- Internal storage implementations such as relational databases, directory and file structures, triple stores, XML formats, etc.
- Application or community-specific identification schemes for information objects that may or may not be URI-based and protocol-based, such as strings, DOIs, Handles, URNs, PURLs, etc.
- Architecture or implementation specific APIs and/or user interfaces.

This is true for the platforms supporting well-known web portals such as Amazon, Flickr and YouTube, as well as systems that are commonly used by scholarly communities such as Fedora, DSpace, ePrints, arXiv, PubMed Central, Elsevier's Science Direct, JSTOR, ArtSTOR, the HighWire Press Journal platform, and others.

OAI-ORE defines an *interoperability layer* that defines a standardized means for exposing these *private* repository-specific implementations of compound information objects to the *public* web, leveraging the web architecture, which consists of:

- *URIs* that identify
- *resources*, which are "items of interest", that,
- when accessed through *standard protocols* such as HTTP, return
- *representations* of current resource state
- and which are linked via *URI references*
- thus forming the *web graph*, which is the basis for services (e.g. robot-based search engines) and data mining (e.g., citation analysis) over the entire web.

The OAI-ORE interoperability layer allows standard web clients and services such as robot-based search engines to access and introspect upon compound objects. It provides the foundation for the development of other value-adding cross-repository services for analysis, reuse, and re-composition of compound objects.

This reconciliation via ORE of the private and public facets of compound objects has the following additional features:

- The interoperability (public) facet does not replace the system-specific (private) facet, but rather co-exists with it. In fact, a scenario where similar (e.g. Fedora, DSpace , etc.) information

systems discover each other via the interoperability layer and then communicate via possibly higher-functionality system-specific mechanisms is feasible.

- Although the interoperability layer is not bound to or dependent on aspects of the system-specific architectures, it may have provision for the exchange of system-specific or application-specific information among cooperating systems. For example, information systems that share an identifier scheme such as a DOI might use the OAI-ORE interoperability layer to exchange those identifiers and thereby leverage the additional functionality or semantics provided by them, such as establishing equivalence classes based on shared DOIs or other identifiers.
- While OAI-ORE is intended to augment interoperability among "information systems", it should be noted that there is no clear definition of such an entity. OAI-ORE resolves this with a solipsistic approach similar to OAI-PMH. In both cases an information system is referred to as a "repository" which is a network accessible service that supports the interoperability fabric (OAI-PMH or OAI-ORE). This avoids concerns with whether an information system actually "contains" compound objects, and focuses on whether it projects them into web space in the manner defined by OAI-ORE. This useful for aggregators and, notably, standard web servers that may have no real internal concept of compound objects but may use the OAI-ORE interoperability layer to expose aggregations of content – for example, sets of web pages – as compound objects.

This notion of heterogeneous information systems and services exposing compound objects to the web infrastructure is illustrated in Figure 1.

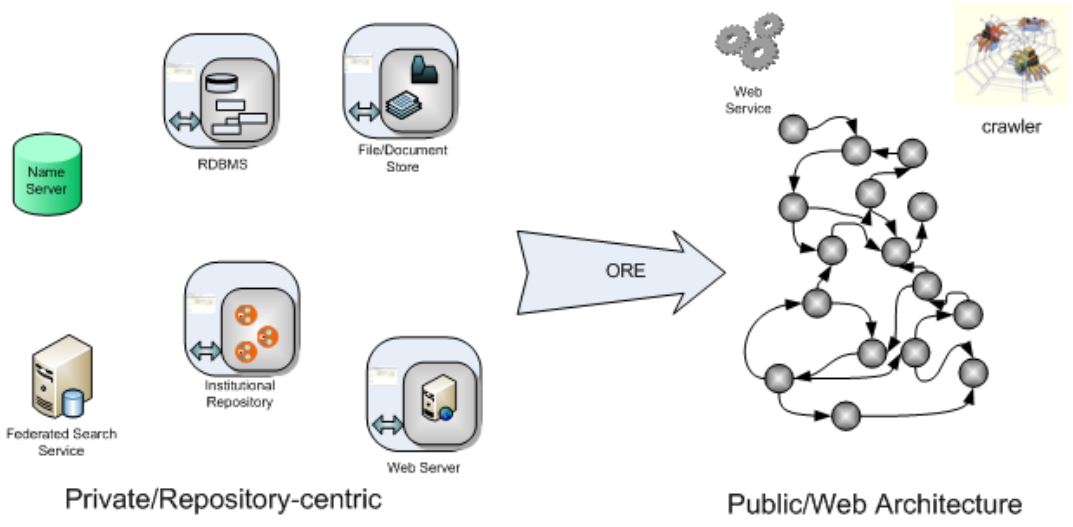

Figure 1 – Heterogeneous information systems exposing compound objects in a common manner

The remainder of this document is a point-by-point description of the aspects of this interoperability layer, and its mappings from private information systems to the public web. Each section expands on a

fundamental definition or aspect, and thereby the document as a whole serves as a "checklist" for developing consensus in the ORE community.

3. Compound Information Object: A Bounded Aggregation

A *compound information object* is an identified bounded aggregation of related information units that together form a logical whole. Each constituent information unit of a compound information object is referred to as a *component*. Components can themselves be compound information objects.

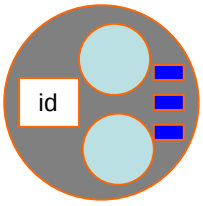

Figure 2 - Abstract view of compound information object

4. Compound Information Object: Aggregation of Multiple Component Types

Components of a compound information object may vary according to semantic type (article, book, video, dataset, etc.), and media type (pdf, XML, mp3, etc.).

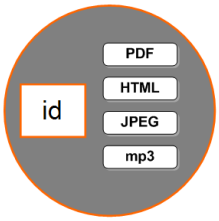

Figure 3 - Multiple media types in a compound information object

5. Compound Information Object and its Access Repository

The information system that provides access to a compound object is known as its *access repository*. This repository exposes compound objects via interfaces and identification schemes, both machine-oriented (APIs) and user-targeted, that may be specific to the respective access repository architecture. Such *private* interfaces and identification schemes may be shared among similar repository architectures or specific application profiles, but do not comprise an interoperability architecture across heterogeneous repositories. Details about the compound objects within individual repositories, for example their structure, location, or composition, are essentially opaque to clients and services that do not share the specific interfaces and identifier schemes.

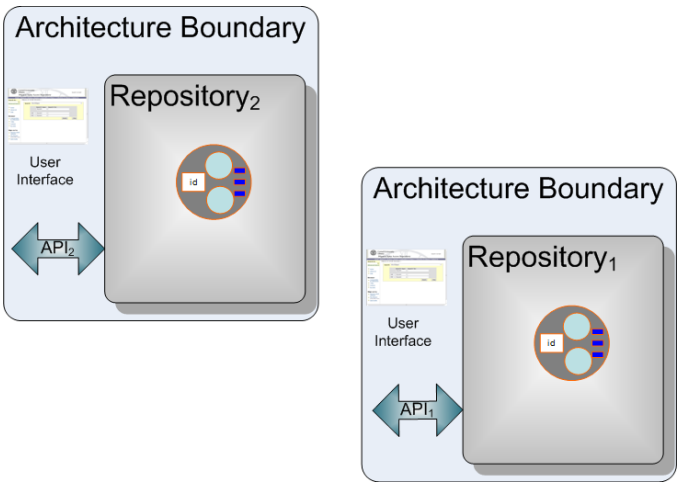

Figure 4 – Multiple access repositories with distinct interfaces and identifier schemes

6. Exposing Compound Information Objects to the Web

An access repository may assign URIs to a compound information object and/or to its components, thereby making them resources in the web graph, and allowing web-based applications to obtain representations from them through protocol negotiation. However, because the web architecture provides no standard method for expressing the logical boundary of a compound information object and the relationships among its parts, web applications have no way of detecting that these distinct URIs originate from the same compound information object.

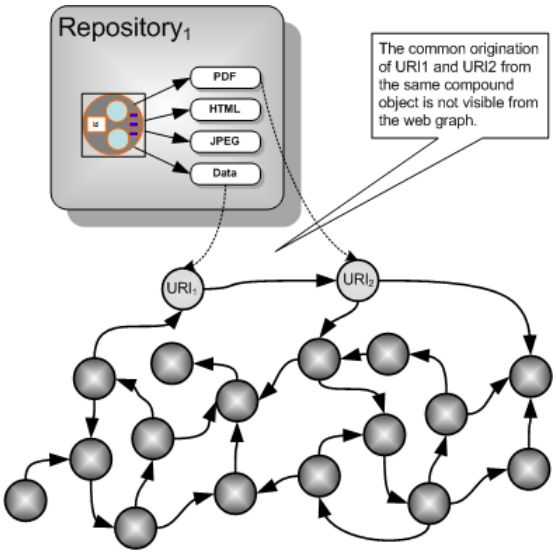

Figure 5 –URIs assigned to selected components of a compound object

This form of exposure has even less utility to machine-based (in contrast to user-oriented) applications when the components of the compound object are embedded in “splash” pages, navigational context,

or client-based scripts (e.g., javascript). For example, the page images of books in the web-accessible National Academies Press are embedded within user-interface context, making machine interpretation and reuse of the actual page text difficult.

7. ORE Resource and Canonical Representation

An access repository exposes its compound objects to the public web in an interoperable manner, and thereby becomes an *ORE repository*, by:

- Denoting a single distinguished resource, the *ORE resource*, for each compound information object. This ORE resource must be identified by means of an HTTP[3] URI.
- Making available via the HTTP URI from this ORE resource a distinguished representation, the *Canonical Representation (CaR)*. The CaR is an encoded description of the compound information object according to the *ORE model*.

There are two important points to note about an ORE resource:

- It may be a pre-existing resource exposed by the repository for reasons other than OAI-ORE interoperability or it may be a new resource introduced for the express purpose of providing access to the CaR.

- Correspondingly, the ORE resource may provide access only to the CaR, or it may provide access to other application or architecture specific representations of an ORE resource, including representations for human consumption.

As a result, an ORE resource can be strictly defined as a resource from which at least a CaR is available.

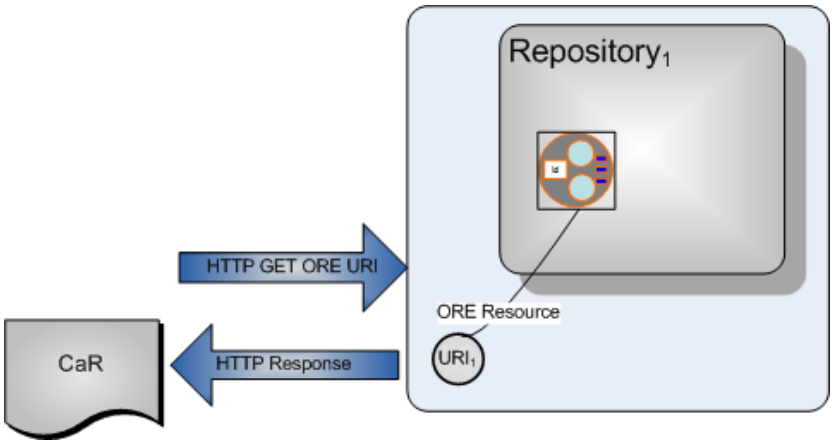

Figure 6 - Access to the CaR of an ORE resource

[3] For the remainder of this document we use HTTP to denote standard HTTP and secure HTTP (HTTPS).

8. Canonical Representation and ORE Aggregation

The CaR, a serialization of the ORE model for a specific compound information object, represents the object as a bounded directed graph with the following characteristics:

- A single *root node*, denoting the ORE resource, identified by means of a HTTP URI. The ORE resource is the web projection of compound information object itself, and thereby provides an anchor or target point for any relationships (links) that apply to the "whole object"; e.g., citation, lineage, etc.
- A finite number of additional nodes known as *component nodes*, denoting other resources, each identified by means of a protocol-based URI[4]. These resources are web mappings of components of the compound information object.
- A finite number of edges, which express typed relationships between the nodes (root node, component nodes). These relationships are known as *internal relationships* because they connect resources within the boundary of the respective compound object.

The entire subgraph described by the CaR, both root nodes, component nodes and edges, is known as an *ORE aggregation*.

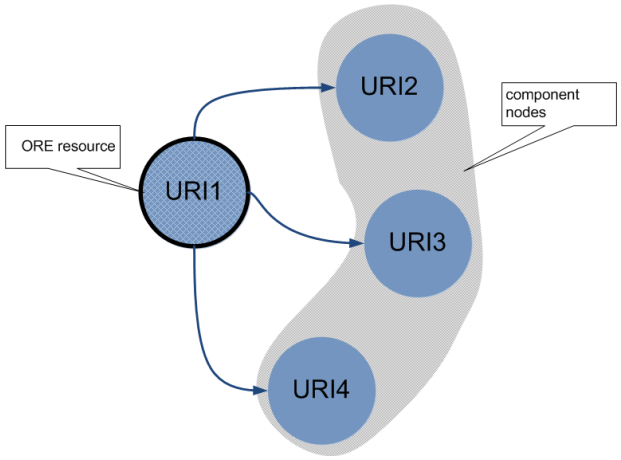

Figure 7 - Compound information object modeled as an ORE aggregation

9. Distributed Resources in an ORE Aggregation

As described, a compound information object is projected to the web when an access repository denotes a single resource, the ORE resource, as the web projection of that object, and correspondingly provides access to a distinguished representation, the CaR, from that resource that serializes an ORE aggregation, a subgraph describing the compound object as a union of nodes and edges.

[4] For the remainder of this document, a "protocol-based URI" is a URI of a URI scheme for which there is a commonly accepted protocol based resolution mechanism. Examples are HTTP, HTTPS, and FTP.

Correspondingly the HTTP URI of the ORE resource must be serviced by that access repository. However, the protocol-based URIs of the component nodes in the ORE aggregation may be served by the access repository or by other services (e.g., repositories) on the network.

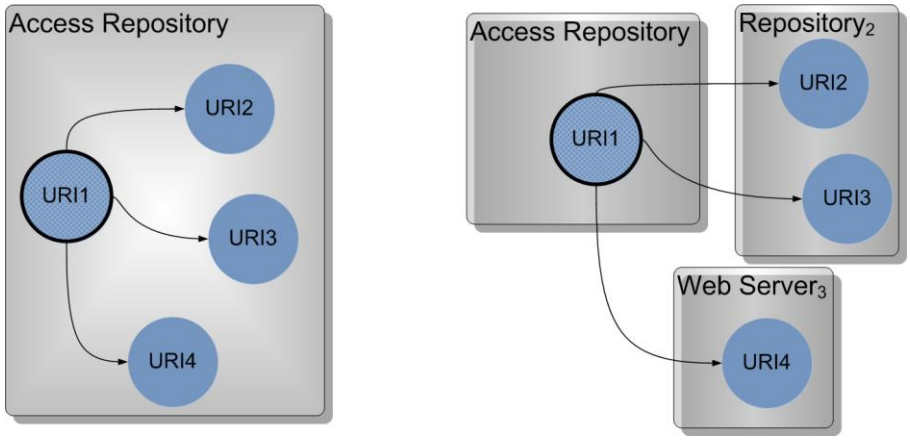

Figure 8 – Local and distributed ORE aggregations

10.Exclusivity of ORE Resource, Sharing Component Nodes in ORE Aggregation

One resource can be the root ORE resource in only one ORE aggregation. On the other hand a component node in an ORE aggregation can appear as an ORE resource in one other ORE aggregation and/or as a component node in multiple other ORE aggregations. In other words, the boundaries of ORE aggregations can overlap, but the root ORE resource is exclusive to a specific ORE aggregation.

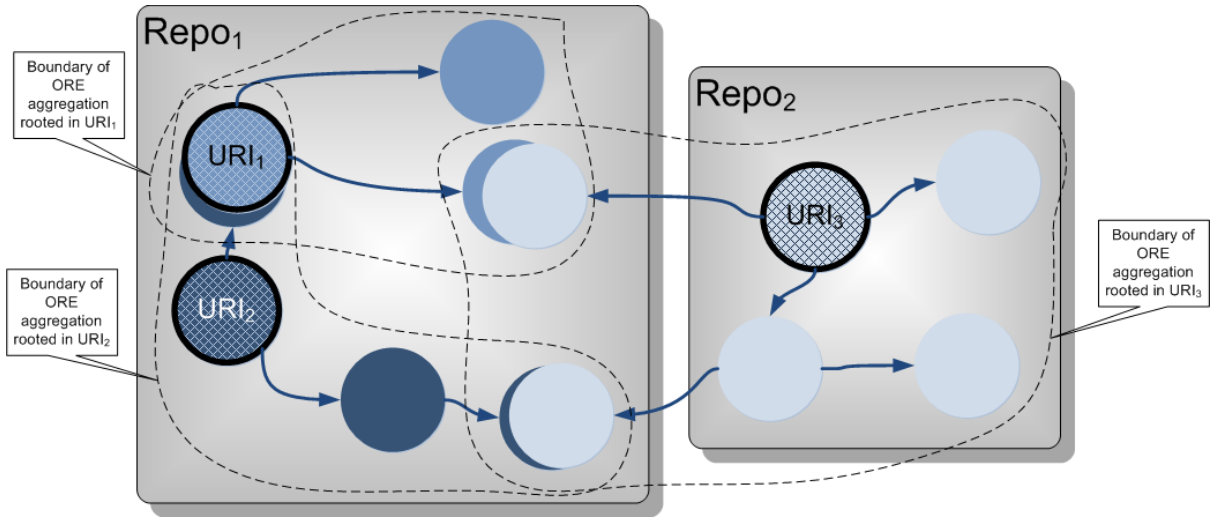

Figure 9 – ORE aggregations that share resources

11.Relationships among Compound Information Objects

A compound object and its components may have relationships that reach beyond the logical boundary of the object. These relationships include citation, provenance, and the like. As a result, resources within a single ORE aggregation may have relationships to resources outside of the ORE aggregation. An ORE repository may express these relationships in the CaR that is available from the compound information object's ORE resource. These relationships are referred to as *external relationships*. The ORE model, and hence CaRs must make an unambiguous distinction between internal and external relationships.

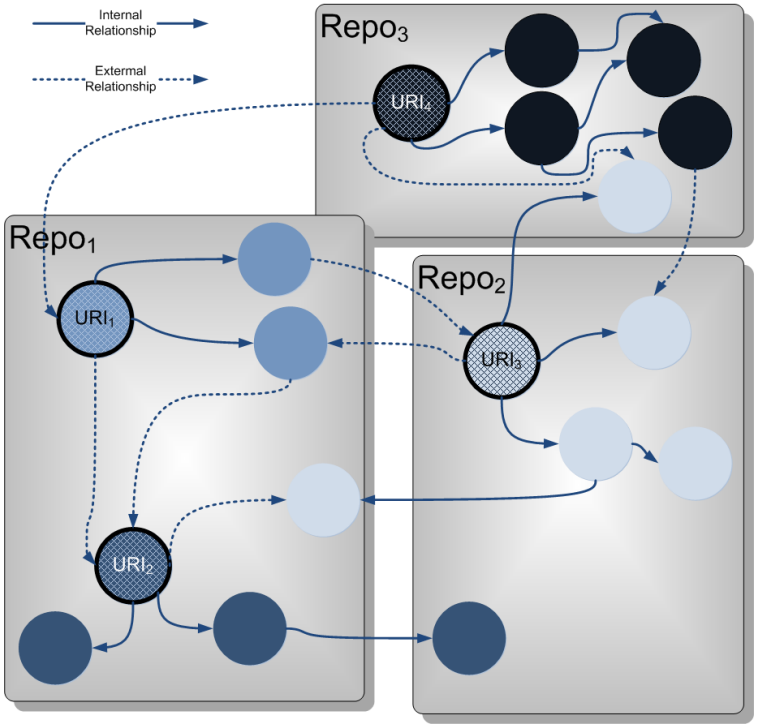

Figure 10 - Compound digital objects linked via relationships

12.Relationship of the CaR graph to the Web graph

Section 6, describes the shortcomings of merely assigning URIs to components of a compound information object. The ORE resource and CaR overcome these problems by providing a common mechanism for HTTP access to the ORE aggregation, allowing introspection into the structure of that aggregation. External services such as web search engines can request this CaR through protocol negotiation and thereby understand the composition and boundaries of a compound information object and adjust their processing accordingly.

This process is illustrated in the following three figures. Figure 11 shows the topology of a web graph before accessing the CaR from an ORE resource. As shown the ORE resource, labeled URI_1, is already present in the graph. Also shown is another resource, labeled URI_2, which is a component of the compound information object in $Repository_1$. The node may be visible in the web graph for a variety of reasons – in this case it is linked to from yet another resource in the web graph.

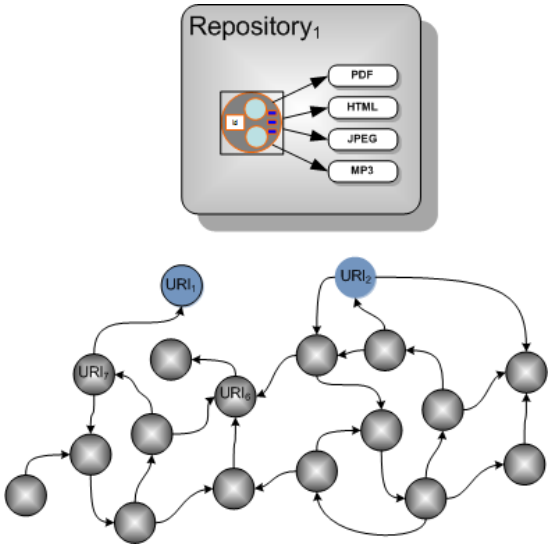

Figure 11 – Web graph before access to CaR from ORE resource

Figure 12 then illustrates the CaR graph that is available upon HTTP(S) access (e.g., by a robot-based search service) to URI_1.

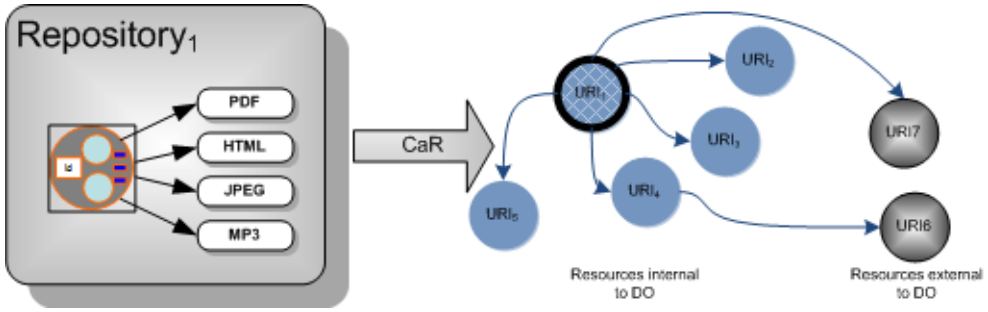

Figure 12 – CaR graph returned as representation of ORE resource (URI_1). Note the internal and external relationships

Finally, Figure 13 shows the net effect of the service's access to and interpretation of the CaR. As illustrated, the nodes and edges of the CaR are fully introduced into the web graph and the boundaries of the compound information object have become visible.

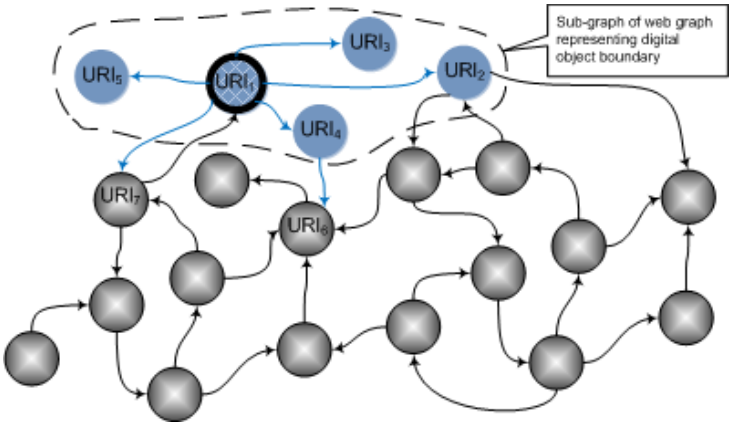

Figure 13 - CaR graph imposed on pre-existing web graph

13.Viewing it from the Opposite Perspective

This document thus far has approached the compound information object problem from a repository-centric perspective. It has done this by first describing the "private" notion of a compound object within a repository. It then followed by showing how a repository maps from this private to a "public" web-based notion of a compound object through the use of the ORE resource, CaR, and the ORE aggregation.

We note in closing that this document could have been presented from the "back-to-the-front". Viewed from the opposite perspective, the mechanisms defined by ORE provide a means of extracting from the web graph about compound objects that was formerly hidden behind the barriers of specific information system interfaces and models. Figure 1 shows this web-centric perspective: a web graph with several compound object embedded in it, each rooted in an ORE resource that provides access to the structure of each object and its boundaries. The CaR may possibly contain additional information about the "private" facet of the object such as additional identifiers, metadata, and access schemes.

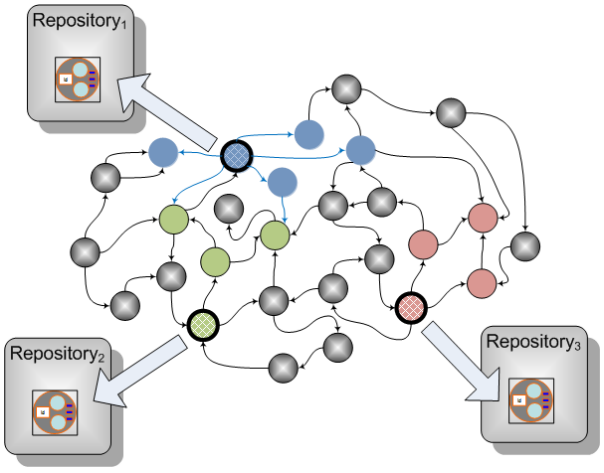

Figure 14 - Revealing compound objects in the web graph