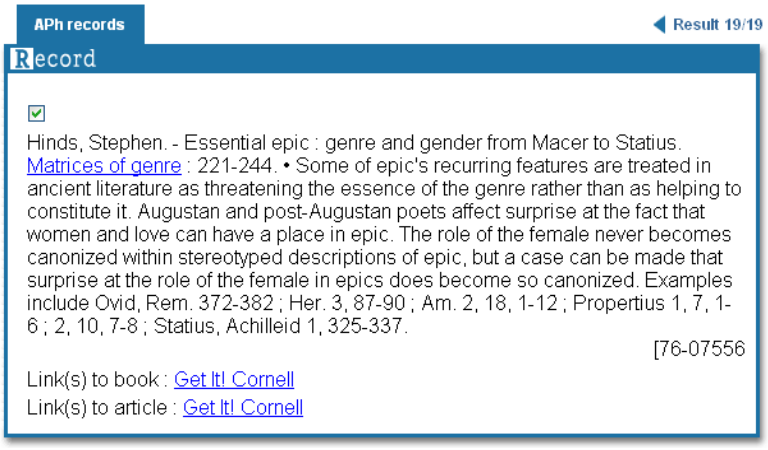
Canonical Citation Linking and OpenURL

May 26, 2009

Problem Statement

A recent grant from The Andrew W. Mellon Foundation ("Planning Grant for the Future of *L'Année philologique* on the Internet"), funded several efforts at measuring, improving, and extending OpenURL resolution services within the context of Classical literature. A portion of this grant supported an exploration into the possibilities and challenges of using OpenURL to provide system independent linking between citations of Classical literature and an increasing array of available online resources in Classics. The current problem, or lack of such linking, can be illustrated with a record found in *L'Année philologique* (APh), an abstracting and indexing service specializing in scholarship about Classical literature. Figure 1 is the web display of a record from the *APh* database.

Figure 1: Record within *L'Année philologique*



This record contains citations to specific passages within five works of Classical literature. Scholars within this discipline immediately understand these citations. They know, for example, that "Am. 2.18.1-12" is a citation to Ovid's *Amores*, and they know how to look up this passage in a print copy of the *Amores*, should they have one available. They may also know that an electronic version of the Latin work is accessible online within the Perseus Digital Library, as well as in an English translation, and they may know how to navigate directly to these.

With an increasing number of Classical texts available online, however, not having immediate linking capability between citations and texts is a limitation. One remedy to this problem is for services such as *APh* to build bilateral links between individual citations and one version of the electronic resources to which they refer. For example, the citation "Am. 2.18.1-12" in Figure 1 could link directly to Book 2, Elegia 18, line 1 of the Latin text of the *Amores* in the Perseus Digital Library.

While a step forward, this approach also has limitations:

- It is costly to implement, requiring that each service that references Classical works learn and implement the linking heuristics of potentially many online text resources.

- It is inherently unstable and costly to maintain. Any change in the linking mechanisms of a target resource will immediately break all embedded links to that resource and require that they be updated.

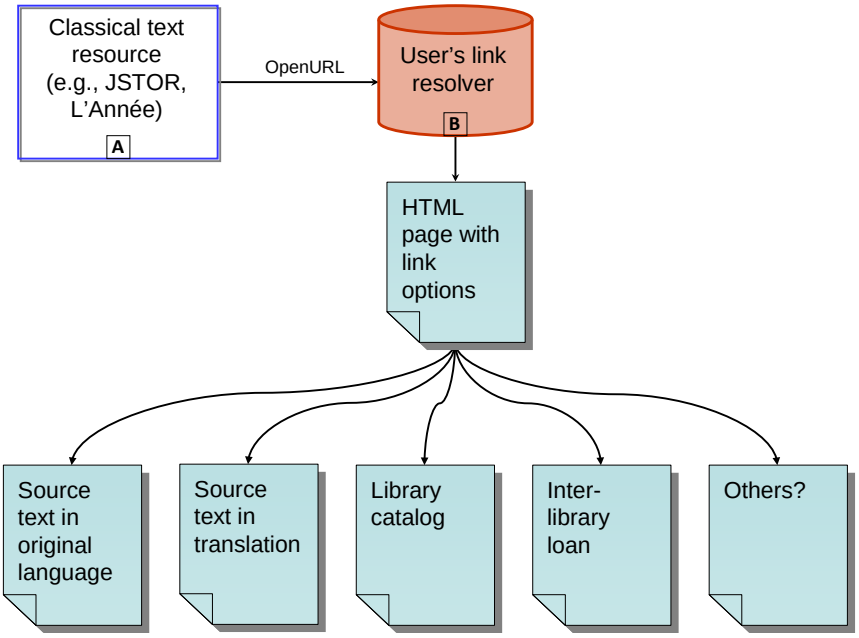
- It does not easily allow for a one-to-many relationship, between the citation and an array of possible options, such as different editions of the original language, different translations of the original, and other potentially relevant resources.

- It does not allow for "appropriate copy" linking, so that a user is only referred to resources to which he or she has access.

OpenURL Solution: In Theory

Since OpenURL was designed to confront just these sorts of limitations in linking between and among scholarly references, we have investigated its application to the problem of canonical citation linking. At an abstract level, the proposed solution is rather simple, illustrated in Figure 2.

Figure 2: Canonical citation linking



Although we are using an example from Classics, the figure illustrates the standard OpenURL information flow. A user, interacting with an online service that includes citations to other resource, clicks on a link to another resource. That action sends an OpenURL to the user's link resolver [B], which in turn is able to present an array of options to the user. In this particular example, the resource [A] includes citations to Classical texts. Examples of such resources are indexing and abstracting services, such as *APh*, and online sources of scholarly research, such as JSTOR, where users are likely to encounter a reference to a Classical work or, more likely, to some component of that work.

The OpenURL is a package of data in a standard format. The central focus of the OpenURL is on the referenced resource, called the "Referent" in the OpenURL Framework, and the OpenURL contains metadata and/or identifiers describing that Referent. The OpenURL may also contain information about the Requester (the person or resource making the service request), the ServiceType (what type of service is requested), the Resolver (the user's link resolver), the Referrer (the resource that creates the OpenURL), and the ReferringEntity (the resource, such as an article, that references the Referent).

Link resolvers, such as [B], are typically operated by libraries for the communities they serve. They contain a customized knowledge base, which allows them to match incoming metadata and identifiers about the Referent to resources available to the community. For example, if the OpenURL's Referent is an article available in Journal X, the knowledge base is able to decide among multiple possible sources of Journal X, choosing the one that is most appropriate for its specific community. A particular community may also

wish to offer other options to users, such as a library catalog search or Inter-Library Loan request service.

An OpenURL approach to linking among Classical text citations and appropriate resources has a number of appealing features. For one, it would allow services such as [A] to build stable and system independent links from citations. All of these links, regardless of the particular text being cited, would be constructed using the same logic and syntax. The service [A] would not need to know how to build a link to a specific text resource, but only how to construct an OpenURL in a standard format containing the citation information. This separation between link creation and link resolution frees services such as [A] from the costs of maintaining these links in a changing online environment. It also is a more cost-effective method for dealing with the inevitable change and expansion of linking possibilities, as these changes would be managed by link resolvers such as [B]. Finally, the OpenURL approach gives more control to a user's specific community to determine appropriate linking resolution possibilities.

OpenURL Solution: In Practice

There are practical challenges to using OpenURL to allow linking from citations of Classical literature. To understand these challenges, the entity-relationship model developed in FRBR (Functional Requirements for Bibliographic Records) is useful.[1] FRBR makes a hierarchical distinction among the entities of interest to users of bibliographic records: the *work*, *expression*, *manifestation*, and *item*. Current OpenURL implementations, focused as they are on secondary scholarly literature, have been used to generalize the syntax and semantics of references to physical forms of a work—a particular *manifestation* or *item*.

Classical citations, however, such as "Ovid, Rem.," are references at the FRBR *work* level. The FRBR *work* is an abstract entity, "a distinct intellectual or artistic creation" distinguishable from any material representation of it; distinct even from any particular realization, or expression, of the work (16). The reference to the *work* must therefore be independent of any particular version, edition, or translation.

Furthermore, for such citation linking to be useful, it must be able to reference not only the work but its components. The citation "Ovid, Am. 2.18.1-12" is to a particular passage within a work, and a useful citation would need to convey unambiguously the component designation.

OpenURL encodes data about a reference using defined metadata formats. As noted, existing OpenURL metadata formats were developed for physical forms of a work, such as journal literature, books, and dissertations. The OpenURL Framework, however, is designed for extension and allows communities to define and register new metadata formats. One challenge for this project, therefore, is to develop an OpenURL metadata

[1] Final Report / IFLA Study Group on the Functional Requirements for Bibliographic Records. München: K.G. Saur, 1998. (UBCIM Publications, New Series; v. 19). Also available as http://www.ifla.org/VII/s13/frbr/frbr.htm or http://www.ifla.org/VII/s13/frbr/frbr.pdf.

format that allows for the description of a *work* and its components, and which can be reliably machine processed. This effort is described in more detail below.

A second practical challenge concerns implementation. Assuming there is an effective way to describe works and their components, how would such references be resolved to the correct and appropriate online resources? In existing OpenURL implementations, this responsibility lies with link resolvers. The link resolver contains a knowledge base that allows it to resolve the OpenURL to an appropriate resource. For canonical work citations, that knowledge base would include, at a minimum, the ability to identify a described work, understand any reference to its components, and build an array of possible links, primarily those directly to the referenced passage in potentially multiple online resources.

This knowledge, in the case of Classical literature, is not only highly specialized and complex, its potential user base is relatively small compared to the much larger universe of OpenURL users. Since link resolvers are commercial products, it is arguable whether the creation and ongoing maintenance of such detailed knowledge bases could be expected from multiple link resolver vendors.

In considering different ways that canonical citation resolution might be implemented, we have explored whether a centralized and community supported knowledge base could better serve the community. Centralizing the collection and maintenance of such knowledge would likely be a more cost-effective management approach. A specialized knowledge base could also offer other potential advantages. The operation of such a domain specific knowledge base is examined in more detail in a later section below.

Workshop on Canonical Citation Linking

On October 10, 2008, this project sponsored a one-day workshop held at Cornell University Library to discuss the project's goals and feasibility. The twelve invited guests included domain experts in Classics and OpenURL. See Appendix A for a list of participants and meeting agenda.

Prior to the meeting, a proposed metadata format for describing work level citations was distributed, and part of the meeting focused on this document. Workshop participants also discussed several models for how a centralized knowledge base might function.

Insights from the workshop inform the following sections about the canonical citation metadata format, the need for work identifiers, and the practical benefits of a centralized knowledge base in Classics.

Canonical Citation Metadata Format

As noted above, one of the first challenges for canonical text linking is to design a metadata format for representing a canonical work citation. There are two parts to a

citation reference that need to be represented. One is a reference to the work itself. The other is to a passage, or component, within the work.

Works will always have a title and often an author. In current OpenURL metadata formats, author names are captured either as a full name in a single data element (au), or broken into two elements, for family (aulast) and given (aufirst) names. Because the impetus of our effort was on Classical literature, which predates the common modern understanding and use of first and last names, these elements were recognized immediately as problematic. On the other hand, they were perfectly adequate for names after about 1500 and are widely understood and easy to apply. It was decided to provide two schemes for encoding author names. In general, we believe that the decision to use one or the other scheme will roughly follow whether the work is pre- or post-1500, but other domain or cultural considerations may apply. For names that fit the modern first/last convention, family (aulast) and given (aufirst) name elements are provided.

The challenge to capturing Greek and Latin names was to keep it simple, so as not to require expert knowledge in order to apply the metadata format, and at the same time making it general enough so that it would be applicable beyond Classics. Rather than distinguishing parts of a name, it was decided to distinguish different forms of a name. Two forms have been proposed. The first (auform1) is the name as it is found in the OpenURL producing resource. For a resource that publishes scholarly literature, either current or historical, this may be the form of the author as it is found in the source literature. For an indexing and abstracting service, this form may be a standardized one used throughout the service. The second form of the name (auform2) is the full name of the author in a format that is authoritative within the domain in which it is used. How this authority form is established is not specified in the metadata format and would be determined within specific domain applications.

To convey work titles, we decided to apply the same logic used for author names. For modern works, a single title element (title) is available. For earlier works, two forms of the title are provided (titleform1, titleform2).

Together, the author name and work title identify the work, a method of identification that has several weaknesses. The canonical citation metadata format therefore includes another element to capture a work identifier. The need for this identifier and how it would be created is discussed in the following section.

The second part of the canonical citation is a reference to a particular component or passage of the work. While this part of the reference may be considered secondary to the identification of the work, in an online environment it will be especially important to connect to specific passages.

In considering how to reference the components of works, two approaches were explored. The first was to attempt to name all the possible component parts of a work that might be referenced, such as a book, section, paragraph, canto, stanza, epigram, line, etc. This approach presented several challenges, the first being whether all possible component

names could be exhaustively named. Beyond that, there is the question of whether all those applying the result would agree on the terms used for any particular work. A second approach was more abstract. Component citations all have a hierarchical structure to them, typically indicating a nested arrangement of components from top down (Act 5, Scene 3, line 12). If one ignores what the components are called in any particular work, the structure can be easily captured using generic terms for each level of the hierarchy (level1=5, level2=3, level3=12). This second approach seemed the least difficult to implement across a range of heterogeneous material and less likely to require specialized knowledge to apply. Workshop participants supported this approach.

To allow citations to a continuous passage, in addition to a specific line, the abstract approach adopted includes start and end indicators for each hierarchical level needed. For example, a citation to the passage at Act 5, Scene 3, line 12-24, would be encoded as:

```
slevel1=5 & slevel2=3 & slevel3=12 & elevel3=24
```

End indicators can be omitted if they are equivalent to the start indicator of the same level. In other words, a more verbose, though equivalent, expression of the above is:

```
slevel1=5 & elevel1=5 & slevel2=3 & elevel2=3 &
      slevel3=12 & elevel3=24
```

It should be noted that this proposal for indicating particular passages within a work assumes that the passage is continuous. A continuous passage may extend across an arbitrary number of hierarchical components. For example, the citation:

```
slevel1=2 & elevel1=3 & slevel2=4 & elevel2=2 &
      slevel3=1 & elevel3=24
```

indicates a passage beginning at Act 2, Scene 4, line 1 (assuming drama) and running through Act 3, Scene 2, line 24. In order to indicate a set of discontinuous passages using the proposed approach, multiple, separate citations would be required.

The currently proposed metadata format for representing canonical citations using a key encoded value format is presented in Appendix B. This proposed format requires additional testing and feedback from those most likely to implement it.

Work Identifiers

In most current working examples of OpenURL, the heart of the reference, the Referent, is a FRBR *manifestation*, and the aim of link resolvers is to present users with a range of possibilities focused on that reference. The most common goal is to connect users to the FRBR *item* that embodies the *manifestation*—that is, the OpenURL references a scholarly article, and the link resolver's goal is to find an appropriate copy of that article for its user. Other linking possibilities are clearly possible, but the primary goal is to get users to the item.

In pursuit of that goal, the link resolver works to identify the Referent using the data encoded within the OpenURL. Commonly, this includes descriptive metadata and identifiers. At the FRBR *manifestation* level, there is a substantial amount of possible descriptive metadata: author and title, obviously, but also journal or book title, date of publication, volume, issue, or series numbers, ISSN and/or ISBN numbers, page numbers, and perhaps more.

Yet even with this relative wealth of data, it has not proven easy to match references to physical (or virtual) copies. Author and title strings can be unreliable in numerous ways. Modern references, designed to be read and decoded by humans, present machines with numerous parsing challenges. The most reliable, and thus useful, data in a reference, for OpenURL link resolvers, have proven to be numbers, such as ISSNs, years, and pages. Probably the most reliable are Referent-level identifiers, such as DOIs.[2]

While the FRBR *manifestation* has a relative wealth of potential descriptive metadata, the FRBR *work* has very little—at most, an author and title, and perhaps only a title. This puts enormous pressure on these elements in their role as identifiers of a work. But author names and titles are essentially text strings and can be rendered in a wide variety of forms. While such information to a human reader may be unambiguous, in a machine mediated environment, and out of the context of the citing work, identification will be difficult. Further, for the Classical and Medieval periods, there will be many identical titles (e.g., "Sermons") and many identical authors (e.g., "Augustine", "[Anonymous]").

There was a consensus among workshop participants that relying on text strings alone (author and title) to identify works would prove unreliable and problematic. We have therefore added a work-level identifier within the canonical citation metadata format. The type or syntax of this identifier is not defined, as its structure and use would need to be determined within specific domains. The newly defined International Standard Text Code (ISTC) could be a strong candidate for providing the type of identification required by this work identifier, although it may take some time to mature.[3] The following section describes a proposed implementation and use of the work identifier within the domain of Classical literature.

Classical Works Knowledge Base

As noted above, there are a number of challenges related to the implementation of canonical citation linking using OpenURL. We believe that the most cost-effective way to meet these challenges is by means of domain specific knowledge bases, rather than depending on commercial link resolver vendors to assemble and maintain the necessary information. This is especially true in the near term. In time, link resolver vendors may absorb some of this functionality into their systems, but whether it will ever be in their financial interest to maintain such specialized data is uncertain. This section describes

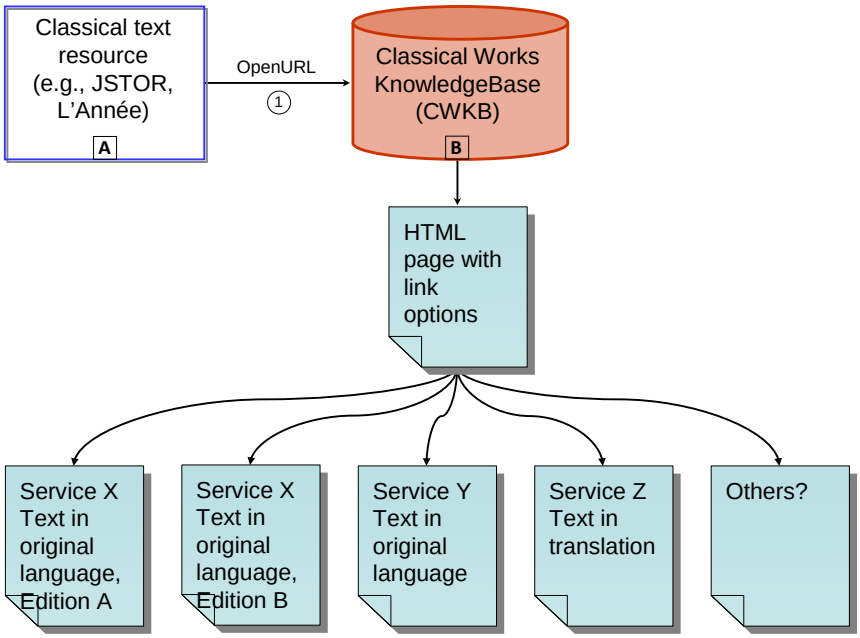
[2] A good description of how metadata inconsistencies impede linking is still very relevant: Miriam E. Blake, Frances L. Knudson. "Metadata and reference linking." *Library Collections, Acquisitions, & Technical Services* 26 (2002): 219–230. DOI:10.1016/S1464-9055(02)00253-1

[3] ISO 21047:2009(E).

how a domain specific knowledge base would function, at times in tandem with local link resolvers.

Figure 3 below illustrates a simple model for OpenURL canonical citation resolution with a common, community supported knowledge base.

Figure 3: Simple CWKB Resolution



User behavior begins with resources that include citations to Classical texts [A]. As described earlier, one goal of this technology is to provide an open-ended array of context sensitive services based on these citations. We can immediately suppose that many users would wish to link to the full-text of this passage in the original language, or to a translation of it in some other language, but other possible actions exist and more will likely emerge.

To use this technology, the only requirement on a resource like [A] is that it must be able to construct an outbound OpenURL that uses the canonical citation metadata format. To do this, it will encode the author and title and, ideally, the work identifier to the citation (or, perhaps, only the work identifier), and it should be able to encode component information according to the canonical citation metadata format. Whether this OpenURL is immediately behind the actionable link presented to the user, or is constructed in another process, does not matter. The user clicks a citation reference and, one way or another, an OpenURL (1) is sent to a resolution service [B] that contains a knowledge base specific to this community.

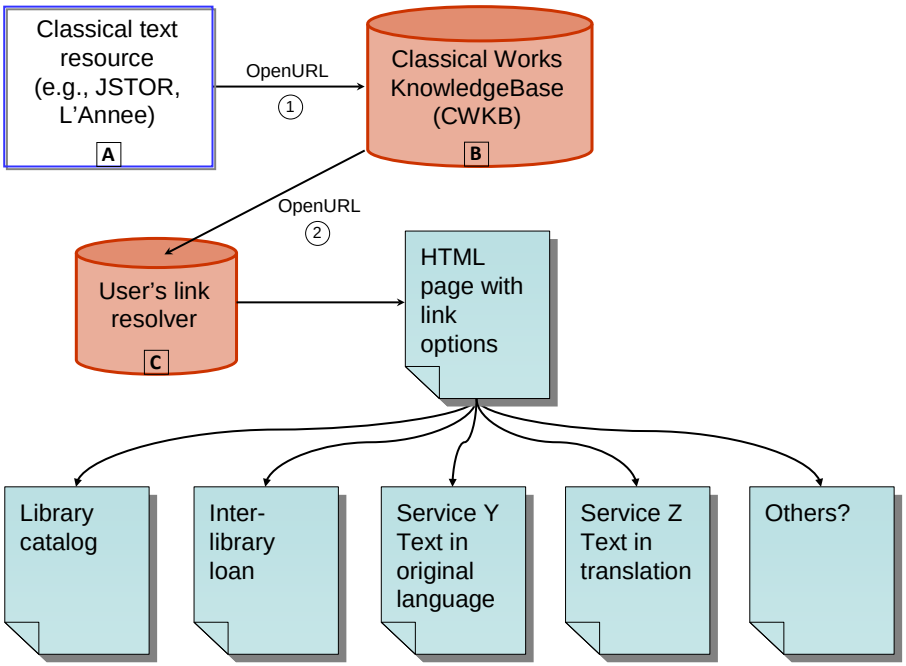
In our example, this domain specific resolution service is called the Classical Works Knowledge Base (CWKB). This service allows for the assembly and maintenance of specialized knowledge about works within its domain and about online resources that can

provide services related to those works. It understands the linking heuristics used by online text resources within its domain and is able to create for any given canonical citation one or many URLs that can take users to specific texts within these resources, and ideally to specific passages. Although our example, and the focus of our study, concerns Classical literature, such domain specific knowledge bases would operate in the same way for other disciplines.

In this simple model of the CWKB (Figure 3), the user is presented with an array of options. If any of these options are restricted to subscribers only, then some users may not have access to them. This model, then, does not provide a good solution to the 'appropriate copy' problem that OpenURL was designed, in part, to solve. This is because the common knowledge base is just that, common, and does not include specific data about particular users and the subscriptions their communities hold. Furthermore, this model does not allow for localized web presentation, page layout, or resource discovery. For example, it would not be possible to present options in a manner that a particular library's users were accustomed to, or to present users with local library catalog searches or inter-library loan options.

To remedy this, and based on feedback gathered at the workshop, we have proposed another model for canonical citation resolution which relies on both the CWKB and a user's local link resolver. This model "chains" link resolvers together, allowing knowledge bases to work in tandem to provide enriched services. The model is diagramed in Figure 4.

Figure 4: CWKB Resolution with User's Link Resolver



As before, a user working in a resource that has Classical text citations [A] will click on a single citation, and an OpenURL (1) is sent to the CWKB [B]. Because the user's library has registered its link resolver with [A], the OpenURL will now include the user's link resolver (in the Resolver entity). Since it has received a Resolver address, the CWKB knows that it is not to display options to the user directly, as in Figure 3, but to pass the OpenURL (2) on to the user's link resolver. Before doing so, however, it modifies the information contained in the OpenURL in these ways:

- Using whatever descriptors of the work are included with the incoming OpenURL (1), the CWKB will seek a match in its knowledge base. Assuming success, it will supply or normalize the data values describing the work. For example, if only a recognized work identifier has been provided, the CWKB will supply form2 author and title fields. Any provided form2 author and title fields will be normalized to the authoritative forms in the CWKB. Any incoming work identifiers and form1 author and title elements will not be included in the outgoing OpenURL (2).

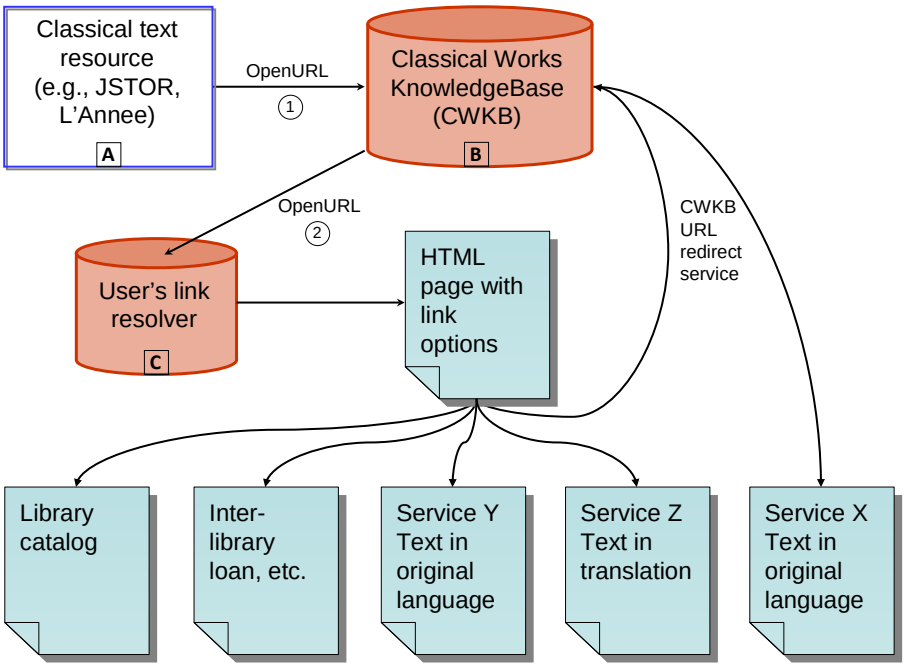
- In the outgoing OpenURL (2), the CWKB will include a list of resources that provide services pertaining to the citation, such as resources that provide the text in the original language, resources that provide translations, and perhaps others. Each resource will be identified by a service code, and a URL directly to the target citation will be included. This information will be presented in a list of ServiceType identifiers employing a CWKB SID (Source Identifier).

The resulting OpenURL (2) is sent on to the user's link resolver [C], which then makes its own set of decisions regarding what options to present to the user. Certain options may be dictated by local practice, such as providing catalog searches or inter-library loan options. Since the OpenURL now has a standardized form of the author's name and the title of the work, such options should be successful.

The user's link resolver will also analyze the list of services provided by the CWKB and present to the user those that are appropriate—in other words, those that it knows the user can access. These can be presented to the user as discrete links, one per resource, which will resolve directly to the target resource.

One additional service may be required of the CWKB. Some vendors will only accept POST requests into their systems as a method of targeting specific passages. This can present problems for certain link resolver systems, where programming outgoing POST requests is not possible. To allow access to these services, the CWKB can easily accept a special GET request that is then immediately redirected to the appropriate service. In this way, the CWKB can serve as a broker between vendors and link resolvers, providing this URL redirect service for those vendors that do not wish to publicize the mechanics of direct linking into their systems. Figure 5 illustrates this redirect service of the CWKB.

Figure 5: CWKB Resolution with URL redirect service



See Appendix D for example ContextObject representations for the two OpenURLs used in Figures 4 and 5.

It should be noted that the models diagrammed in Figures 3 and 4 are not mutually exclusive in the operations of a domain specific knowledge base. Such a knowledge base would be expected to support the Figure 3 model at a minimum, and even if capable of supporting a Figure 4 model, should still be able to provide an array of options directly to users who have no local link resolver. In other words, these models represent levels of service as opposed to operational alternatives. In this way, enhanced service can be offered to all users, regardless of whether they have a link resolver or not.

In addition to the OpenURL resolution services described above, such a domain specific knowledge base would likely be capable of supplying other services to its user community. We have already mentioned a redirect service, as a way of linking directly to specific passages within system that do not make such linking publicly available. In addition, since the knowledge base would assemble and maintain a comprehensive list of works, tied to the various identifiers and descriptors associated with them, it could provide an authoritative lookup service. For example, through a defined API, resources with numerous canonical citations, perhaps in dated or non-standard forms, could send what metadata they have to the knowledge base and receive in return normalized and augmented metadata. Or perhaps a text resource wishes to synchronize its work identifiers with others from its domain, a service the knowledge base would be in a position to provide. We imagine that there are numerous services that would develop

once authoritative metadata within a specific domain was collected and made searchable and accessible via a database.

In summary, domain specific knowledge bases as proposed here could provide a number of advantages:

- The collection and maintenance of knowledge peculiar to a specific domain is centralized, lowering the overall costs of this work and making the domain less dependent on commercial link resolver vendors for the operational success of its linking.

- The knowledge base becomes the responsibility, and under the control, of the community most directly impacted by its operations, improving incentives for data collection and maintenance.

- OpenURL data elements can be normalized and/or augmented by specialized knowledge bases before being passed to local link resolvers, improving the likelihood that the data can be acted upon.

- A domain knowledge base may serve as a broker between certain vendors, such as full-text resources, and individual link resolvers, so that direct access to texts could be permitted through known and regulated interfaces.

- A domain specific knowledge base would be able to provide other services to its community, such as work lookup mechanisms, metadata augmentation services, and citation disambiguation.

- This approach provides a model of chaining knowledge bases together that may have wider applications in the OpenURL community.

Prototype

To test the feasibility of the CWKB, Cornell University Library has built a prototype of the system as sketched in Figures 3-5. A publicly accessible demonstration of the prototype system is available here:

http://cwkb.org

The components of the prototype include:

- A set of web pages that illustrate the starting point of a user's actions. These are simple HTML mockups of fictitious resource records, containing sample citations to the works within the prototype knowledge base, and employing a range of potential OpenURLs targeting the knowledge base.

- A knowledge base, populated with sample data on 4 Greek works. This knowledge base includes information on how to build precise links to these works within the Thesaurus Linguae Graecae and the Perseus Digital Library.

- A link resolver. We have used Cornell University Library's WebBridge link resolver (Innovative Interfaces, Inc.), making modifications as necessary.

By means of this prototype, we have tested, and continue to test, the canonical citation metadata format and the proposed functionality of a domain specific knowledge base, both on its own (offering users direct links to resources) or in conjunction with a user's link resolver. One outcome of the prototype has been a sample database table structure for such a knowledge base. The prototype also demonstrates how a local link resolver, with minimal configuration, can accept incoming canonical citation OpenURLs from a domain specific knowledge base and present users with alternatives based on locally available resources (for both on and off-campus users). Finally, the prototype demonstrates the feasibility of using the domain knowledge base for a URL redirect service, in the case where requested resources require specialized treatment (such as a POST request).

Appendix A: Workshop on Canonical Citation Linking and OpenURL

Cornell University Library
October 10, 2008

Participants

Alison Babeu, Perseus Digital Library, Tufts University
Christopher Blackwell, Associate Professor and Chair, Classics Department, Furman University [Canonical Text Services project].
Adam Chandler, Coordinator, Service Design Group, Digital Library and Information Technologies, Cornell University Library.
Dee L. Clayman, Professor of Classics, Executive Officer PhD Program in Classics, Graduate Center, CUNY [Database of Classical Bibliography].
Hans Deraeve, Commercial Manager, Brepols Publishers [Library of Latin Text].
Theodore Fons, Director, OCLC WorldCat Global Metadata Network.
Eric Hellman, Director, OCLC New Jersey.
Michael Krot, Technology Research Associate, JSTOR.
Maria C. Pantelia, Professor and Chair of Classics, University of California, Irvine, Director Thesaurus Linguae Graecae [TLG].
Eric Rebillard, Professor of Classics and History, Cornell University, General Editor of *L'Année philologique* online.
David Ruddy, Director, E-Publishing Technologies, Digital Library and Information Technologies, Cornell University Library.
Neel Smith, Associate professor of Classics, Holy Cross College and Center for Hellenic Studies [Canonical Text Services project, First Thousand Years of Greek project].

Workshop Agenda

9:00 AM
- Welcome
- Introductions
- Project Introduction
  - description of problem and motivations
  - objectives of planning grant
- Discussion of potential OpenURL solutions
  - OpenURL metadata formats
  - proposed canonical linking metadata formats (KEV matrices)
  - pros/cons
  - examples

Lunch
- Discussion of potential implementation problems
  - in Classics
  - in the supply chain (knowledge base)
- Next steps

4:00 PM – Adjourn

Matrix defining the KEV Format to represent a canonical citation

dc:title	KEV Canonical Citation Format (draft)
dc:creator	Cornell University Library
dc:description	This Matrix represents the Canonical Citation Format as a string of ampersand-delimited Key/Encoded Value pairs
dc:identifier	http://cwkb.org/docs/matrix
dc:identifier	info:ofi/fmt:kev:mtx:canonical_cit (proposed)
dcterms:created	2009-03-30
dcterms:modified	

A representation of a Key/Encoded-Value pair is generated by concatenating the contents of the first four columns of a row that begins with an ampersand in the Matrix below. The ordering of KEV pairs is not important. Rows which have '#' in the first column are comments and **should not** be included in the representation.

The following data types are provided for the values of the Keys, which must be URL-encoded:

<data>	Character string
<id>	Character string for an Identifier (Z39.88-2004, Part 1, Section 7)
<fmt-id>	Character string for a Format Identifier (Z39.88-2004, Part 1, Sections 12 and 13)
<m-key>	Character string for a Metadata Key (Z39.88-2004, Part 2, Section 7.1)
<url>	Character string for a URL
<date>	Character string representing a date to the complete date level of the W3CDTF profile of ISO 8601, of the form: [YYYY-MM-DD | YYYY-MM | YYYY]
<time>	Character string representing a date and time to the seconds level of the W3CDTF profile of ISO 8601, of the form: [YYYY-MM-DDThh:mm:ssTZD | YYYY-MM-DD]

Abbreviations in column headings:

- Delim - Delimiter
- Min - minimum occurrence

- Max - maximum occurrence ('*' = unbounded)

The Matrix

Delim	Key	Equals	Value	Min	Max	Description
&	**work-id**	**=**	**<data>**	0	*	The identifier for this work.
&	**auform1**	**=**	**<data>**	0	1	Use this author element for pre-1500 works. First author's name as it is available to the OpenURL generating resource.
&	**auform2**	**=**	**<data>**	0	1	Use this author element for pre-1500 works. International authority form of first author's name. For an ancient Greek author, it is the Latin form of his name, e.g., Homer is recorded as "auform2=Homerus".
&	**aulast**	**=**	**<data>**	0	1	Use this author element for post-1500 works. First author's family name. William Shakespeare is recorded as "aulast=Shakespeare".
&	**aufirst**	**=**	**<data>**	0	1	Use this author element for post-1500 works. First author's given name or names or initials. William Shakespeare is recorded as "aufirst=William".
&	**titleform1**	**=**	**<data>**	0	1	Use this title element for pre-1500 works. The title as it is available to the OpenURL generating resource.
&	**titleform2**	**=**	**<data>**	0	1	Use this title element for pre-1500 works. International authority form of the title. For an ancient Greek work, it is the Latin form of the title, e.g., the Odyssey is recorded as "titleform2=Odyssea".
&	**title**	**=**	**<data>**	0	1	Use this title element for post-1500 works. The title of the work.
&	**slevel1**	**=**	**<data>**	0	1	The start of the first or highest hierarchical level in the canonical citation. Homer Iliad 1.125-2.35 is recorded as "slevel1=1".
&	**slevel2**	**=**	**<data>**	0	1	The start of the second highest hierarchical level in the canonical

						citation. Homer Iliad 1.125-2.35 is recorded as "slevel2=125".
&	**slevel3**	**=**	**<date>**	0	1	The start of the third highest hierarchical level in the canonical citation.
&	**slevel4**	**=**	**<data>**	0	1	The start of the fourth highest hierarchical level in the canonical citation.
&	**slevel5**	**=**	**<data>**	0	1	The start of the fifth highest hierarchical level in the canonical citation.
&	**elevel1**	**=**	**<data>**	0	1	The end of the first or highest hierarchical level in the canonical citation. If omitted, equivalent to slevel1. Homer Iliad 1.125-2.35 is recorded as "elevel1=2".
&	**elevel2**	**=**	**<data>**	0	1	The end of the second highest hierarchical level in the canonical citation. If omitted, equivalent to slevel2. Homer Iliad 1.125-2.35 is recorded as "elevel2=35".
&	**elevel3**	**=**	**<data>**	0	1	The end of the third highest hierarchical level in the canonical citation. If omitted, equivalent to slevel3.
&	**elevel4**	**=**	**<data>**	0	1	The end of the fourth highest hierarchical level in the canonical citation. If omitted, equivalent to slevel4.
&	**elevel5**	**=**	**<data>**	0	1	The end of the fifth highest hierarchical level in the canonical citation. If omitted, equivalent to slevel5.

Appendix C: Proposed Work Identifier Specification for Classics

As discussed in the body of this report, the work identifier will likely be of critical importance to the unambiguous and reliable identification of works. The canonical citation metadata format therefore includes a key for providing the work identifier.

We anticipate that the use of such work identifiers will differ across disciplines and that communities will develop their own conventions and requirements regarding the use of these identifiers. The following identifier syntax and discussion is an example of this for the domain of Classical literature.

We propose to register a "cwkb" namespace within the `info` URI scheme. The `info:cwkb/` identifier would be used to identify a work within the CWKM knowledge base.

Identifiers within the `info:cwkb/` namespace will consist of two parts, both required. The first part designates an identification authority, within which any following parts are assured to be unique. The final segment of the identifier is an item number. The work identifier is case-insensitive.

The following specification uses the Augmented Backus-Naur Form (ABNF) notation as defined in RFC4234, including the ABNF Core Rules for ALPHA (letters), DIGIT (0-9), and VCHAR (visible ascii characters: x21-7E).

Proposed CWKB Work Identifier Syntax

```
work-identifier = namespace "/" identifier-source ":" item-
identifier

namespace = "info:cwkb"

identifier-source = *(ALPHA / DIGIT / "_")

item-identifier = 1*VCHAR
```

A work may have more than one identifier, as it may be included within more than one identification authority. For example, the CWKB should be able to recognize all of the following identifiers as referencing the *Supplices* by Aeschylus:

```
info:cwkb/tlg:0085.014
info:cwkb/tlg_demo:0085.001
info:cwkb/cts:greekLit:tlg0085.tlg001
```

All three of these work identifiers would be considered valid. Ideally, the Classical Works Knowledge Base would recognize all legitimate work identifiers within the discipline of Classics.

Appendix D: OpenURL ContextObject Representations (Figure 4)

The ContextObject representations below illustrate the two different OpenURLs in Figure 4. For simplicity, included here are only the data elements important to understanding the purpose and operation of the proposed Classical Works Knowledge Base, or a similar knowledge base in a different domain. Actual ContextObjects would very likely contain additional entities (such as the ReferringEntity). The ContextObject representations have been formatted for readability.

OpenURL (1)

This OpenURL ContextObject is generated by a resource with citations to Classical texts and is sent to the Classical Works Knowledge Base (CWKB). In this example, the Referrer has included a work identifier and available author and title forms. The user's link resolver address is included ("res_id"), and the Referrer, the service that generates the OpenURL, is identified by the "rfr_id" identifier.

```
   ctx_ver = Z39.88-2004
&  rft_val_fmt = info:ofi/fmt:kev:mtx:canonical_cit
   &  rft.work-id = info:cwkb/phi:0959.001
   &  rft.auform1 = Ovid
   &  rft.titleform1 = Am.
   &  rft.slevel1 = 2
   &  rft.slevel2 = 18
   &  rft.slevel3 = 1
   &  rft.elevel1 = 2
   &  rft.elevel2 = 18
   &  rft.elevel3 = 12
&  res_id = http://resolver.library.cornell.edu/net/openurl/
&  rfr_id = info.sid/aph
```

OpenURL (2)

The CWKB accepts the OpenURL above, augments and normalizes it, and sends the following ContextObject to the user's link resolver. Author and title form1 elements have been replaced by form2 elements in the Referent. If the vendor had provided form2 elements, these also would have been normalized to the CWKB's forms. The vendor provided work-id has been removed. The ContextObject also now includes a complete list of services pertaining to the Referent, as ServiceType identifiers utilizing a CWKB Source Identifier syntax. With this information, the user's link resolver can present a number of options to the user, such as a catalog search or ILL request. It will interpret the list of services, and depending on what the user's community has access rights to, it can present the user with several distinct options related to the full-text (the original Latin and an English translation).

```
   ctx_ver = Z39.88-2004
&  rft_val_fmt = info:ofi/fmt:kev:mtx:canonical_cit
   &  rft.auform2 = Ouidius, Publius Naso
   &  rft.titleform2 = Amores
```

```
   &  rft.slevel1 = 2
   &  rft.slevel2 = 18
   &  rft.slevel3 = 1
   &  rft.elevel1 = 2
   &  rft.elevel2 = 18
   &  rft.elevel3 = 12
&  svc_id =
   info:sid/cwkb.org:perseus_lat:url:http://www.perseus.tufts.edu/
   hopper/text.jsp?doc=Perseus:text:1999.02.0068:text=Am.:book=2:
   poem=18
&  svc_id =
   info:sid/cwkb.org:perseus_eng:url:http://www.perseus.tufts.edu/
   hopper/text.jsp?doc=Perseus:text:1999.02.0069:text=Am.:book=2:
   poem=18
&  svc_id =
   info:sid/cwkb.org:brepols_llt-a:url:http://cwkb.org/in/
   vendorpost.php?package_id=llt-a&local_package_id=0959001&scheme=w&
   hidden_w=2&hidden_x=18&hidden_y=1&hidden_z=1
&  rfr_id = info:sid/cwkb.org
```