Arcadia Falcone
May 2, 2013
Archival Enterprise II
Prof. Ciaran Trace

Linked Data and Archival Description: A Guide for the Perplexed

Abstract

Linked data has evoked increasing attention in the archives profession, but for many archivists its technical components and practical applications remain obscure. This paper offers a clear explication of the *what*, *how*, and *why* of linked data geared specifically to archival concerns. Part I lays the foundation with a low-jargon introduction to linked data, followed by a comparison in Part II of linked data's web and EAD's tree as alternative methods of structuring archival description and representing the archival bond. Part III explores the Social Networks in Archival Context project as an example of how linked data operates in archival practice. Examining linked data from an archival perspective illuminates its strong relevance to the profession's current trends toward greater polyvalence, interoperability, and flexibility.

Part I. From Web Documents to Linked Data: The Basics

The World Wide Web displays semantic content configured for human perception, not machine processing. The Hypertext Markup Language (HTML) that underlies most webpages tells the web browser how to distribute objects across the canvas of the page, without encoding much about what those objects signify.[1] Does an image portray a symbolic icon, a person's portrait, or a photograph of St. Paul's Cathedral? Does a paragraph of text describe a published book, a historical event, or a commercial product? Does the text form part of an article, a poem, or a letter? Does the same text also appear in a printed newspaper or in a manuscript held in an

[1] HTML5 begins to address this, but it is not yet broadly implemented; throughout this paper, "HTML" refers to the current standard, HTML 4.

archive? Do the words "Francis Bacon" refer to the Elizabethan philosopher or the twentieth-century painter? A human reader often notices the contextual clues to answer many of these questions almost unconsciously. A web browser, however, knows nothing about the semantic content of its constituent elements, but only how to display them: position this image file at the top of the page, insert that image file next to a particular block of text, play this video file when the mouse clicks on the screen area where that button appears. Faced with a web page displayed in a browser (given certain basic caveats, such as understanding the vocabulary of the page's text), a person may discern the concepts, physical objects, and digital entities to which the page's semantic content refers, while the browser itself registers only structural, HTML-encoded links indicating the locations of other digital objects.

Linked data provides a structure for enriching the contextual information computer applications may derive from HTML links. Rather than the Uniform Resource *Locators* (URLs) that HTML uses to indicate the network location of a resource, linked data relies on Uniform Resource *Identifiers* (URIs) to identify a resource as a conceptual entity, instead of as a particular file stored on a particular server. The use of "resource" in both terms is somewhat misleading, as the word functions in two different, but overlapping, senses. The "resource" in URL must be a digital object of some kind, as those are the only resources that have a location on the web, while the "resource" in URI may be digital, physical, or abstract, and may refer to any object, entity, or concept for which the URI acts as an identifier. In "A Short History of 'Resource' in Web Architecture," Tim Berners-Lee distinguishes these two usages of "resource" under the umbrella terms "Document" and "Thing," respectively,[2] and I will borrow the shorthand of "document" and "thing" for the general classes of resources that URLs and URIs may each identify.

[2] http://www.w3.org/DesignIssues/TermResource.html.

In simplest terms, a URL identifies a place on the web, while a URI identifies a thing in the world. Linked data combines the two by using URIs that are also URLs: the document whose *location* is the URL identifies the *thing* associated with the URI. For example, http://id.loc.gov/authorities/names/n79045512 is both a URI for George Eliot and the URL for a document[3] that describes George Eliot. Note that it is *a* URI, not *the* URI, for Eliot: while a URI represents only one thing,[4] that thing may have any number of URIs according to different shared and local ontologies. (An ontology in this context is a set of specifications for terms and the relationships between them in a particular information system, analogous to a controlled vocabulary thesaurus.) The Library of Congress Name Authority File URI for George Eliot links to Eliot's URI in the Virtual International Authority File (VIAF)[5], which in turn links to her URI in the OCLC Identities Network.[6] A URI thus functions somewhat like an authority record, in that it provides an unambiguous identification of a resource. Unlike an authority record, however, the URI does not depend on a rigorously controlled set of natural-language terms. Instead, it relies on a "pay as you go" approach, where, for example, an online dataset of characters in Eliot's novels may use URIs from common vocabularies, like those listed above, to identify Savonarola, a real historical figure, but create its own set of URIs for fictional characters not identified in an existing accessible URI ontology (or, indeed, add its own URI for "Savonarola the fictional character in *Romola*" that in turn references the URI for "Savonarola the historical figure"). Linking the URIs for these characters to one or more of the existing URIs

[3] Even when documents are automatically generated on the fly from a database (to provide personalized recommendations or respond to user actions, for example), rather than persisting over time as static documents, HTML is still what is delivered to the browser for display.
[4] "One thing" may be distinguished at any level of granularity: "George Eliot," "the text of *Middlemarch*," "Virginia Woolf's personal copy of *Middlemarch*," "the fifth word of the second chapter of *Middlemarch*," and "the Victorian novel" may each operate as "one thing" for the purposes of a URI.
[5] http://viaf.org/viaf/89000553.
[6] http://www.worldcat.org/wcidentities/lccn-n79-45512.

for George Eliot would identify them as Eliot-related resources for computer applications accessing the data.

Besides using URIs to identify things in a way computers can recognize, linked data also describes the relationships between those things through a data structure called Resource Description Framework (RDF). If URIs are the vocabulary, RDF is the syntax. The basic unit or "sentence" of RDF is the *triple*: two things (which again may be objects, ideas, etc.) linked by a predicate showing how the first thing (the subject) is characterized by the second (the value). "George Eliot → was born in → England." "Dorothea Brooke → is a character in → *Middlemarch*." "*The Mill on the Floss* → was published in → 1860." Or even, "The proofs of *Middlemarch* → are held by → the Harry Ransom Center." The center term is referred to as the "attribute:" subject → attribute → value, as in "this subject → has this attribute → with this value." In linked data practice, any or all of the parts of these triples could appear not as words but as URIs, so that not only the things, but also the relationships between them, are machine-readable. The use of URIs to "type" the relationships means that "you can combine two RDF graphs [datasets composed of triples] and just get a bigger list of relationships between things."[8] Unlike URLs, which simply make a navigational bridge between one document and another without encoding any information about the nature of the connection, URIs in RDF triples not only create a connection between two things, but explain what that relationship is in a way that computers can interpret.

Together, these connections through URIs build what Tim Berners-Lee calls "a Semantic Web – a web of data that can be processed directly or indirectly by machines."[9] In terms of semantic structures, HTML is rather like a language whose only verb is "links to," and which

[8] http://openorg.ecs.soton.ac.uk/wiki/Linked_Data_Basics_for_Techies.
[9] Tim Berners-Lee, *Weaving the Web: The Past, Present and Future of the World Wide Web By Its Inventor* (London: Texere, 2000), 191.

parses "George Eliot" and "Mary Ann Evans" (and "Marian Evans") completely differently, but the "George" in Georges Eliot, Sand, and Washington exactly the same. The Semantic Web, in contrast, speaks in "things not strings":[10] regardless of the sequence of letters, or string, that a particular document displays, the URI identifies the "thing" that the string represents. Or, if using an information system to illustrate an information system isn't too much of a rabbit hole of an analogy, HTML is a bit like a simple version of Facebook: you have one relationship, "friend," that links to the version of the person represented within the site by their Facebook profile. In an RDF social network, you could specify the nature of the relationship ("had midnight walks in high school with," "met at a party once," "still hold a grudge against"), and choose any number of profiles from any sites to represent the person – identities which may themselves link to each other, connecting data across sites. While such a system might be less than ideal for maintaining a congenial social life or a sense of privacy, it would produce a much richer dataset for an application trying to model the social network structure. While all linked documents are simply "friends" in HTML, and the same "friend" may have unconnected profiles on different sites, the Semantic Web provides a structure for capturing a more complex array of relationships, and for linking together different representations of the same object, idea, or person.

But just as the friend of your friend may be your enemy, getting linked data from different sources to mesh may present significant challenges. Since anyone may create their own URIs or apply existing ones to their data, some means of evaluating reliability is necessary to promote the good-quality linked data and bury the bad. But what constitutes quality? While some work has been done towards creating explicit criteria, standards, and vocabularies for

[10] http://dublincore.org/resources/training/NISO_Webinar_20130123/nisodcmi-webinar-bibframe-20130123.pdf.

evaluating linked data,[11] "quality" remains a slippery term that depends on what is being evaluated, for what purpose, and by whom. These concerns are magnified when the data in question touches on contested, controversial, or ambiguous areas, for which different sources may offer variant or conflicting information: a URI describing Ayn Rand might have very different content depending on whether it comes from the perspective of an Objectivist, a literary critic, or a feminist, for example. Other online services have addressed similar issues (Google's PageRank algorithm pushes what it determines to be the most relevant results to the top; Wikipedia has a system of reader notifications editors may invoke when an entry does not meet quality standards; many comments sections support "upvoting" or otherwise rating entries), and their strategies may provide a starting point for linked data quality evaluation. Linked data may sweep away some existing data filters as arbitrary or irrelevant, but will need to construct new ones in order to make its results manageable. Strategies for mitigating these problems will become more imperative as the Semantic Web continues to grow.

Despite these unresolved issues, linked data does offer significant affordances for enriching description of resources of all kinds. The following categories focus on some of the benefits most relevant to archivists.

-*Interoperability*. Numerous schemas and standards for archival description are in use globally. Projects such as the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) and the Dublin Core Metadata Initiative (DCMI) have created high-level structures for encoding metadata so that it may be aggregated across repositories using different standards, but these structures continue to be predominantly document-oriented rather than data-oriented. Linked data allows not just crosswalks between schemas, but also connections to data more broadly, beyond the archival repository. Interpolating archival description into the web of data

[11] http://blog.law.cornell.edu/voxpop/2013/01/24/metadata-quality-in-a-linked-data-context/.

increases the discoverability of archival resources through services used for general (rather than archives-specific) information retrieval, such as Google.

-*Flexibility*. Linked data is capable of mixing attributes from different schemas, and applying those attributes with as much or as little granularity as desired. The machine-readable structure of the data supports multiple interfaces: the same data may be presented differently for different uses, rather than requiring a one-size-fits-all approach.

-*Extendibility*. New URIs may be created as necessary, and added to the system without disrupting those already in place, accommodating and contextualizing resources of general obscurity but local significance.

-*Decentralization*. Linked data is not dependent on any one authority to function, and no gatekeeper controls participation or implementation. While this may cause some issues with consistency and quality control, it also increases flexibility and adaptability.

-*Discoverability*. Linked data best practices encourage URIs to link to other resources, so that accessing one node leads to others. Instead of creating a self-contained information system within the individual repository, state consortium, or even archival institutions in general, linked data connects things, regardless of their containers.

Essentially, linked data proposes a metadata schema for describing the world, and for connecting things in the world regardless of who describes them, what site hosts the description, and which language is used. Structuring URIs into RDF triples enables applications to parse information about and connections between things in the world, opening them to more productive analysis. Linked data does not supplant the document-based web, but rather augments it. Just as the semantic web of data offers greater precision than the HTML web of documents for identifying things and their connections to a larger conceptual network, linked

data presents an alternative to document-based EAD for describing the complex relationships between texts and contexts within an archival collection.

Part II. Encoded Archival Description vs. Linked Data: The Tree and the Web

Traditionally documents have comprised not only the material contents of archival collections, but also their primary descriptive genre. Finding aids, inventories, indices, and calendars are all documents that narrate the structure of the collection according to a pre-determined order. This order may reflect an intellectual arrangement of the items rather than their physical positions within the collection, but a central tenet of archival theory is the archival bond: the idea that a record signifies not only through its content as an individual document, but also, and importantly, through its context as part of a collection that the related actions of a creating entity have produced. The evidentiary value of a record is not in its paper but in its provenance. Since the relationships between documents, actions, and creators are thus key to the archival value of a collection, the finding aid evolved to represent a nested, hierarchical sequence of categories: series, subseries, folder, document. At each level, the categories fork into mutually exclusive segments like the branches of a tree, so at the most granular level each unit retains an unambiguous relationship to the whole structure. Description of each level appears at that level, and is assumed to apply downwards to the levels below without being repeated. This information architecture supports a search strategy that begins at the topmost level, evaluates at each level which category is most likely to contain the desired information, and continues to drill down to the smallest unit receiving description.

When archives began to transform finding aids from print to digital in the mid-1990s, a group of researchers developed a standard markup language for these documents, Encoded

Archival Description (EAD). What EAD encodes, for the most part, is not descriptive content but structural arrangement: while a few EAD tags identify specific data types (e.g., <title>, <date>), the primary focus is on delineating the structural levels and components organizing description along the different branches of the finding aid tree. Unlike the infinitely extensible and polyvalent web of linked data, EAD constructs a closed system that positions each of its composite parts in static, orthogonal relation to each other. EAD mirrors the limitations of physical arrangement, in that an archivist must assign each document to a single place within the organizational whole, forcibly resolving any ambiguity or multiplicity. Constructing this order, as Jennifer Meehan has argued, ideally "produces a single conceptual and physical entity—a processed collection—from different groups or accessions of records in various states of (in)completeness and (dis)array."[13] Michelle Light and Tom Hyry have called attention to the subjectivity of this act of construction, and argued for a more transparent representation within the finding aid of the decisions that shape its structure and content.[14] Linked data goes a step further, by deconstructing the very notion of a singular, unitary, hierarchical order that EAD embodies.

Proponents of traditional arrangement and description practices may at this point be imagining linked data as kudzu running riot over their carefully tended topiary. But while the EAD hierarchy structures archival description as a tree that branches taxonomically into more and more precise classification of documents, a linked data approach to archival metadata emphasizes the web of things that those documents reference. "Things" in this case might include creators, functions, and other documents – in short, all the ingredients for maintaining

[13] Jennifer Meehan, "Making the Leap from Parts to Whole: Evidence and Inference in Archival Arrangement and Description," *The American Archivist* 72 (Spring/Summer 2009):89.
[14] Michelle Light and Tom Hyry, "Colophons and Annotations: New Directions for the Finding Aid," *The American Archivist* 65 (Fall/Winter 2002).

the archival bond. Rather than obscuring context, linked data has the potential to represent a contextual network with greater precision, complexity, and richness than EAD allows. Imposing one definite order means that the archivist must choose which context to prioritize, at the expense of all others. Linked data, in contrast, allows multiple contexts to co-exist, and new connections to be added without overwriting existing ones. A letter from T. S. Eliot to Amy Lowell that EAD files in "Correspondence – Received – Eliot, T.S." may in linked data radiate connections not only to Eliot and Lowell, but also the Criterion magazine as featured in the letterhead, the upcoming book the letter discusses, the draft of a poem that was included as an enclosure, the place where it was written, the place it was received, other documents Eliot created on the same date, and Lowell's reply held by a different archive – weaving together the various threads that each conduct information about the letter's provenance. In representing context as a multivalent web of relationships, linked data offers a more complex conceptual structure for archival arrangement than EAD's carefully pruned tree of documents.

In delineating document-oriented and data-oriented models for description, archival theory has drawn boundaries in somewhat different places from those distinguishing between the HTML web of documents and the semantic web of data. In "Technology and the Transformation of Archival Description," Daniel V. Pitti enumerates the following characteristics of document-centric information:

- An irregular number of parts or pieces. Documents, even documents of a particular type, do not all have the same number of textual divisions (parts, chapters, sections), paragraphs, tables, lists, and so on.
- Serial order is significant. It matters whether this chapter follows that chapter, and whether this paragraph follows that paragraph. If the order is not maintained, intelligibility and sense break down.
- Semi-regular structure and unbounded hierarchy.
- Arbitrary intermixing of text and markup, or what is technically called mixed content.

- Arbitrary number of interrelations (or references) within and among documents and other information types, and generally the types of relations are unconstrained or only loosely constrained.

In contrast, data-centric information shares the following common traits:

- Regular number of components or fields in each discrete information unit.
- Order of the components or fields is generally not significant.
- Each information component is restricted to data. That is, it has no embedded delimiters, other than the formal constraints of data typing (for example, a date may be constrained to a sequence of eight Arabic numbers, in the order year-month-day).
- Highly regularized structure, possibly with a fixed but shallow hierarchy.
- Relations between discrete information units have a fixed number of types (though the number of occurrences of each type may or may not be constrained).
- Processing of data-centric information (such as accurate recall and relevance retrieval, sorting, value comparison, and mathematical computation) is highly dependent on controlled values and thus highly dependent on machine-enforced data typing, authority files, and a high degree of formality, accuracy, and consistency in data creation and maintenance.[15]

Pitti identifies EAD, and markup languages more generally, as an example of a document-oriented information structure, and databases as an example of a data-oriented information structure.

Linked data, however, operates under a different kind of data-centrism, and thus has different points of contrast and comparison with EAD from what Pitti describes. Linked data is not a self-contained information system like a database or a finding aid, but rather a schema for structuring references that identify things in the world. While databases have little hierarchy and EAD has rigorous hierarchy, linked data accommodates but does not require hierarchy. It treats hierarchy as just another relationship whose type may be defined by a URI. EAD encodes one type of structure: its only predicates are "has child category" and "has parent category." Linked data, as discussed in the previous section, may identify any type of relationship as the link

[15] Daniel V. Pitti, "Technology and the Transformation of Archival Description," *Journal of Archival Organization* 3 (January 2006):12-13.

between two things. At each level, EAD divisions into categories are without remainder and mutually exclusive, with each object occupying a single position within the structure. In contrast to this either/or approach, linked data offers both/and, enabling the identification of any number of agents or functions that have participated in the creation of a record or group of records. EAD relies on the principle of inheritance, where, for example, a scope and content note for a series also applies by extension to all folders within that series, even though the note appears only at the series level. The folder element does not itself contain any reference to the series-level note; the only signifier of the attribution is the folder's position under that series heading in the hierarchy. Linked data, in contrast, asserts relevant data wherever it applies, regardless of repetition, creating entities that, like database records, are modular and may be reordered without losing meaning. Unlike database records, however, which may be rearranged only according to the structures defined within a closed system, the use of URIs means that linked data may maintain its contextual relationships without being dependent upon a single unified architecture.

In sum, EAD encodes a tree structure that branches into ever-narrower categories of documents, while linked data identifies things (including documents) and the polyvalent web of relationships that connects them. While EAD conveys information about its components through their arrangement, linked data constructs a conceptual structure that supports multiple arrangements without losing context. For the user, the primary points of access that the EAD finding aid structure allows are the choose-your-own-adventure approach of selecting a branch at each level of the hierarchy that appears to lead in the right direction, and the fishing-net approach of a keyword search that returns all instances of a particular string. Linked data offers the potential for an approach more akin to browsing Wikipedia, where conceptual links are many and diverse, as well as opportunities for filtering or faceting the array of links in order to

highlight particular networks within the larger web. While user experience in both cases depends on the particular interface that governs how the encoding displays within a browser or other application, linked data has the potential to provide a more complex, if less uniformly ordered, set of raw materials for an interface to draw upon.

Part III. Linked Data in Archival Practice: SNAC and EAC-CPF

Two related projects within the archival community have begun to explore practical applications to fulfill linked data's potential, as well as to build an infrastructure supporting future implementations. Social Networks and Archival Context (SNAC) aims both to develop a discovery tool for archival collections across repositories, and to provide a databank for other linked data projects to reference. SNAC builds upon the newly-developed Encoded Archival Context – Corporate Bodies, Persons, and Families (EAC-CPF) standard, to aggregate entity names from existing archival metadata and link them together with URIs. SNAC illustrates several key practices for applying linked data in archival context: utilizing existing datasets, consolidating information from multiple sources, algorithmic processing of data, presenting each entity as a node within a larger network, and creating workflows and resources that are generalizable to other projects.

SNAC mobilizes linked data to develop a "historical research and access system"[16] for exploring the social networks embedded in archival metadata. Collaborators on the project include the University of Virginia's Institute for Advanced Technology in the Humanities, UC Berkeley's School of Information, and the California Digital Library. Started in 2010, the SNAC project completed its initial pilot stage in 2012, resulting in a prototype implementation and a set

[16] http://socialarchive.iath.virginia.edu/.

of software tools, both available through the project website.[18] Thus far, the project has analyzed EAD finding aids from the Library of Congress, the Online Archive of California, the Northwest Digital Archive, and Virginia Heritage in order to extract the names of individuals, families, and corporations. These identities were then encoded according to the EAC-CPF standard. Drawing on data from multiple sources increases the scope and thereby the utility of the dataset, but also produces duplicates: names appearing in different forms from different sources that represent the same entity. At this scale, identifying and removing these duplicates manually is unfeasible, and so the project team developed an algorithmic solution to match names to authority records in the Virtual International Authority File (VIAF).[19] The prototype interface supports browsing by name, occupation, or subject, as well as keyword search. Accessing an EAC-CPF record for a particular entity may provide the alternate names under which she/he/it appears, and biographical or organizational history notes drawn from finding aids. The neighboring nodes in the linked data web of things also appear, as derived from the finding aids: occupations and subject headings associated with the entity, things the entity created, things that reference the entity, and other people, families, and corporate bodies connected to the entity. Each of these relationships is typed (creatorOf, referencedIn) to create a RDF triple. All of the data in the EAC-CPF record exists elsewhere in finding aids, but SNAC extracts, aggregates, and structures this data, making visible connections that were previously obscured and providing new entry points for access and discovery.

[18] Prototype: http://socialarchive.iath.virginia.edu/xtf/search. Software: http://socialarchive.iath.virginia.edu/software.html.

[19] Initially developed by the Library of Congress, the Deutsche Nationalbibliothek, the Bibliothèque nationale de France and OCLC, VIAF links "all authority data for a given entity … into a 'super' authority record" that functions as a URI. Each VIAF record gives the name of an entity in multiple languages and alphabets, enabling, for example, the identification of "George Eliot" and "Джорж Элиот" as the same person. VIAF site: viaf.org. OCLC VIAF documentation: http://www.oclc.org/viaf.en.html.

SNAC demonstrates the dual benefits that linked data projects can have for both users and archivists. Consolidating and connecting data, not only about archival holdings but also about the entities, functions, and relationships they represent, aids users in discovering information relevant to their interests. Linking entities through URIs is particularly helpful when searching on common names, as it distinguishes sources related to one specific Jane Smith from all the others. Linked data also renders archival description more accessible to general search engines such as Google, enabling archival resources to emerge in conjunction with non-archive-specific information retrieval activities, thus making such resources available to users who are not already archives-savvy. For archivists, co-locating archival records based on their creators or contributors rather than on the archival institution that holds the physical items offers an opportunity to restore the archival bond when it has been fractured over the history of a *fonds*, without negotiating a transfer of possession from one repository to another. Linked data has the advantage over other methods of federated search in that URIs and typed relationships more easily accommodate multiple modes of structuring data. If shared finding aid databases resemble a model airplane that must be built to specification in order to work, linked data is more akin to Legos: modular, adaptable, and extensible to new projects without disrupting the structure already built. The additive nature of linked data (as previously stated, add two linked data networks together and you get a bigger network) has the impetus for a snowball effect, wherein each project lowers the entry bar for future endeavors. Linked data enables not just taking previous projects as a model, but also extending the resources produced by the project in new directions. As SNAC demonstrates, linked data breaks down the walls of data silos and derives added value from the wealth of archival metadata that already exists.

Conclusion: Linked Data and the Future of Archives

What makes linked data more than just the latest technological buzzword is not just its potential applications, but also its consonance with current tonal shifts in archival theory and practice. Linked data speaks to many subjects of rising interest in the profession: considering the archive as a polyvalent confluence of multiple voices, perspectives, and sources, not all of which have been well-served by traditional archival procedures; creating interoperable metadata that connects information across the borders of collections, repositories, and nations; coping with diminished funding by leveraging existing resources for added value; improving the usability of metadata for audiences beyond the trained specialist; and re-envisioning the archivist as a steward of data as well as of documents. Linked data is not a new problem for archivists to solve, but rather a tool that archivists may apply toward the resolution of existing concerns. Archivists are highly skilled at forging productive connections, whether between documents, between collections, between institutions, or between people and information. Linked data offers a potentially powerful means for expanding, enhancing, and extending connectivity in multiple diverse forms, within, across, and beyond the archive, to the benefit of archivists, archives, and users.