An HTTP-Based Versioning Mechanism for Linked Data

Herbert Van de Sompel
Los Alamos National Laboratory, NM, USA
herbertv@lanl.gov

Robert Sanderson
Los Alamos National Laboratory, NM, USA
rsanderson@lanl.gov

Michael L. Nelson
Old Dominion University, Norfolk, VA, USA
mln@cs.odu.edu

Lyudmila L. Balakireva
Los Alamos National Laboratory, NM, USA
ludab@lanl.gov

Harihar Shankar
Los Alamos National Laboratory, NM, USA
harihar@lanl.gov

Scott Ainsworth
Old Dominion University, Norfolk, VA, USA
sainswor@cs.odu.edu

arXiv:1003.3661v1 [cs.DL] 18 Mar 2010

ABSTRACT

Dereferencing a URI returns a representation of the current state of the resource identified by that URI. But, on the Web representations of prior states of a resource are also available, for example, as resource versions in Content Management Systems or archival resources in Web Archives such as the Internet Archive. This paper introduces a resource versioning mechanism that is fully based on HTTP and uses datetime as a global version indicator. The approach allows "follow your nose" style navigation both from the current *time-generic* resource to associated *time-specific* version resources as well as among version resources. The proposed versioning mechanism is congruent with the Architecture of the World Wide Web, and is based on the Memento framework that extends HTTP with transparent content negotiation in the datetime dimension. The paper shows how the versioning approach applies to Linked Data, and by means of a demonstrator built for DBpedia, it also illustrates how it can be used to conduct a time-series analysis across versions of Linked Data descriptions.

Categories and Subject Descriptors

H.3.5 [**Information Storage and Retrieval**]: Online Information Services

General Terms

Design, Experimentation, Standardization

Keywords

Web Architecture, HTTP, Linked Data, Resource Versioning, Web Archiving, Temporal Applications

1. INTRODUCTION

The Architecture of the World Wide Web [11] states that dereferencing a URI yields a representation of the (current) state of the resource identified by that URI, and highlights the impracticality of keeping prior states accessible at their own distinct URIs:

> Resource state may evolve over time. Requiring a URI owner to publish a new URI for each change in resource state would lead to a significant number of broken references. For robustness, Web architecture promotes independence between an identifier and the state of the identified resource.

Nevertheless, use cases abound that require the availability of (representations of) distinct prior states of resources. Resource versioning is of crucial importance in areas as diverse as community-driven content creation, open government, and scientific communication. Also, as more data becomes available in the Linked Data cloud, the need to version them will increase if only to allow efficient update of stores that leverage the data, and to trace their provenance. Web archives and content management systems possess significant amounts of prior versions of resources, but these prior versions are largely disconnected from current versions and discoverable only in an ad-hoc manner. Given this state of Web resource versioning, we consider these challenges:

1. *Given the current version of a resource, how can "follow your nose" style navigation to prior versions of the resource be achieved?*
2. *Given any version of a resource and a particular timestamp, how can "follow your nose" style navigation towards another version that matches the timestamp be achieved?*

This paper is concerned with versioning mechanisms that are **machine-actionable**, have **global scope**, and are **independent of media-type**. Hence, approaches that are mainly beneficial to human users such as untyped hyperlinks in HTML with anchor text that provides navigational guidance (e.g. "previous/next version"), or version semantics expressed in metadata-carrying URIs [13] are not considered. Also, mechanisms that have version indicators specific to a certain server such as the deprecated "Content-Version" header field from RFC 2068 [6] are not considered. Similarly, versioning mechanisms that are specific to media-types such as the *link* element combined with the "prev" and "next" relationships as used in HTML [17] or Atom [15] are not considered.

Our contributions are a resource versioning mechanism based on the global notion of time and an HTTP-based mechanism to navigate across versions. Furthermore, we demonstrate how these contributions can be applied for time series analysis across resource versions, and illustrate this using a linked data example.

Copyright is held by the author/owner(s).
LDOW2010, April 27, 2010, Raleigh, USA.

The remainder of this paper is structured as follows: Section 2 discusses and illustrates characteristics of resource versioning approaches; Section 3 provides an introduction to the Memento framework that extends HTTP with transparent datetime content negotiation capabilities; Section 4 shows how the Memento framework suggests an elegant resource versioning approach that is fully based on HTTP; Section 5 shows how the Memento versioning ideas apply to Linked Data; Section 6 describes the demonstrator we built for the DBpedia environment to illustrate how the proposed versioning mechanism can be used to access prior descriptions of DBpedia concepts via their existing DBpedia URIs. In the same section, we also show how this mechanism was used to conduct a time-series analysis of Gross Domestic Product values for several countries across DBpedia versions. Section 7 reviews some related work, and Section 8 holds our conclusions.

2. RESOURCE VERSIONING

This Section introduces core characteristics of versioning approaches, discusses these characteristics for a typical resource versioning approach, and evaluates how that versioning approach can meet the challenges (1) and (2) from the Introduction.

2.1 Versioning Characteristics

The following four core characteristics of versioning approaches are considered:

1. *Identification*: By which means are different versions identified?
2. *Versioning Strategy*: What is the approach used to assign identifiers to versions, e.g. do new versions receive a new identifier, do they inherit a prior identifier, etc.?
3. *Version Relationships*: How are version relationships between resources expressed?
4. *Version Timestamping*: How is the datetime associated with versions conveyed?

2.2 A Typical Resource Versioning Approach

The quote from the Architecture of the World Wide Web implicitly suggests a versioning approach as depicted at the top of Figure 1. This approach is described in terms of the aforementioned core characteristics:

1. *Identification*: HTTP URIs are used to identify versions of resources; each version has its own URI.
2. *Versioning Strategy*: A new URI is minted for each for each new version. When a use case requires that a resource URI-R_0 that started its existence at t_0, but at time t_1 changes state in such a way that a distinct identity is needed, a new resource with URI-R_1 is minted. And, if consecutively at time t_2 a change in state of URI-R_1 again requires a new identity, a resource URI-R_2 is created (top of Figure 1). URI-R_0, URI-R_1, and URI-R_2 co-exist and, in terms of [2] and its associated ontology[1] are *time-specific* resources. They represent the evolving state of a not explicitly defined, abstract resource and are interlinked by the `http://www.w3.org/2006/gen/ont#sameWorkAs` property.

[1]Ontology for Relating Generic and Specific Information Resources `http://www.w3.org/2006/gen/ont`

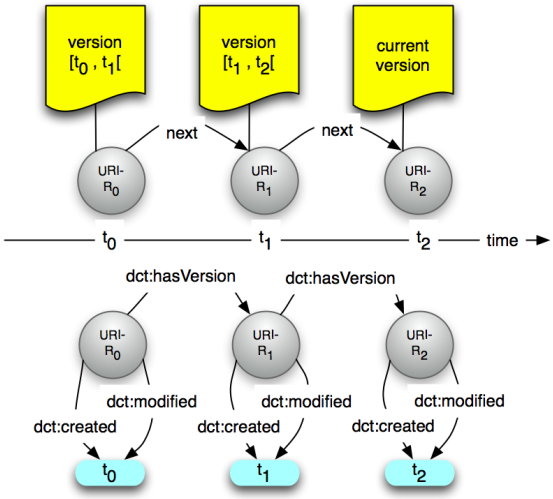

Figure 1: A resource versioning strategy, and its expression in RDF.

3. *Version Relationships*: Can be made available as RDF metadata published about and linked from the related resources. The common Dublin Core Terms[2] *hasVersion* and *isVersionOf* predicates (bottom of Figure 1) can be used. Alternatively, the machine-processable and media-independent *HTTP Link* header [14] can be used in combination with the registered *prev* and *next* relationships to express version-relationship semantics. In both cases, it should be noted that the semantics of the relationships do not strictly only apply to time-based version relationships: *hasVersion* is used in cases where "a related resource is a version, edition, or adaptation of the described resource", whereas *next* refers to "the next resource in an ordered series of resources".
4. *Version Timestamping*: In case of the aforementioned RDF approach, additional triples can be introduced to express version timestamps. For example, at the bottom of Figure 1, the Dublin Core Terms predicates *created* and *modified* are used in accordance to the description for time-specific resource given in the aforementioned ontology as being a resource for which "the dates of creation and of last modification are the same". It is unclear how a version timestamp can appropriately be expressed when using the HTTP Link element: conveying such information is not specified for the *prev* and *next* relationships, the HTTP Last-Modified header does not provide reliable version semantics, and use of metadata embedded in the linked resource yields an approach that is dependent on media-type.

The above characterization reveals the technological substrates that are used in the considered versioning approaches:

[2]`http://dublincore.org/documents/dcmi-terms/`

for the RDF approach, URIs, HTTP, RDF, and an appropriate RDF vocabulary; for the HTTP Link approach, URIs, HTTP, HTTP Link, and registered link relationships. For both approaches, applications such as browser plug- ins, can be created to support the navigation described in question (1) above, whereby the starting point would be the current resource URI-R_2. Also (2) can be achieved for the RDF approach, although a processor would need to traverse versions until a matching datetime is found. Lacking appropriate version datetime information, (2) can not reliably be achieved in case of the HTTP Link header.

To summarize, both (1) and (2) can be achieved for the RDF approach, however: (a) Two technological substrates, HTTP and RDF, must be combined (b) Version datetime can not be used as a primary entry point; rather resource versions must be traversed until a version with a matching datetime is found (3) The common predicates used to express version relationships do not necessarily imply time-based version relations.

3. THE MEMENTO FRAMEWORK

The basic motivation for the Memento[3] work [19] is achieving a tighter integration between the current and the past Web. Remnants of the past Web exist both in version-aware servers such as Content Management Systems (CMS, e.g. Wikipedia) and Version Control Systems, and in special-purpose Web Archives such as the Internet Archive[4] and the on-demand WebCite[5] archive. Whereas a current representation of a resource is available from its URI-R, prior representations - if they exist - are available from distinct resources URI-M_i (i=1..n) that encapsulate the state URI-R had at times t_i, with t_i prior to the current time. In the Memento framework, the resource that provides the current representation is named the *Original Resource*, whereas resources that provide prior representations are named *Mementos*. More formally, a Memento for a resource URI-R (as it existed) at time t_i is a resource URI-M_i[URI-R@t_i] for which a representation at any moment past its creation time t_c is the same as a representation that was available from URI-R at time t_i, with $t_c \geq t_i$. Implicit in this definition is the notion that, once created, a Memento always keeps the same representation.

From a HTTP perspective, URI-R and URI-M_i are disconnected in that HTTP provides no means to navigate towards a URI-M_i via its original URI-R. The Memento framework introduces this missing capability as follows (Figure 2):

- Inspired by Transparent Content Negotiation for HTTP (*conneg* from now on) specified in RFC 2295 [10] that allows HTTP clients to negotiate with HTTP servers in four dimensions (media type, language, character set, compression), Memento introduces conneg in a fifth dimension: datetime. RFC 2295 introduces the notion of a transparently negotiable resource as the resource that is the target of conneg, and variant resources that vary according to the aforementioned negotiable dimensions. Similarly, Memento introduces the notion of a TimeGate URI-G as a resource that supports conneg in the datetime dimension, and Mementos URI-M_i[URI-R@t_i] as the resources that vary according to the datetime dimension. In a manner symmetrical to the way RFC 2295 introduces the *Accept-Language* request header to express the client's language preferences, and the *Content-Language* response header to express the language returned by the server, Memento introduces the *Accept-Datetime* and *Content-Datetime* headers to express the client's preferred datetime for a Memento, and the datetime of the Memento returned by its hosting server, respectively. It can be noted that, although RFC 2295 did not specify datetime conneg, its desirability is at least suggested by Tim Berners-Lee's Generic Resources Statement [2] as all other dimensions of genericity described in it (language, media-type, target-medium) are covered by RFC 2295.

- In order to support discovery of a TimeGate URI-G for a resource URI-R, a relationship type of *timegate* is introduced for the HTTP Link response header [14]. In case of servers that have internal versioning/archiving support (such as CMS) a TimeGate URI-G for URI-R can typically be exposed by the server of URI-R itself. In cases whereby servers rely on third parties for their versioning/archiving (for example by being recurrently crawled by the Internet Archive), URI-R and URI-G will reside on different servers. In addition, in order to allow discovering the Original Resource associated with a Memento, another special-purpose HTTP Link header, this time with a relationship type of *original* is introduced.

- Memento also introduces the notion of a TimeBundle resource via which an overview is available of all Mementos that a server hosts for a given (internal or external) URI-R. A TimeBundle is a non-information resource [18] modeled as an ORE Aggregation [12] in which all Aggregated Resources share a temporal relationship with the Original Resource. A TimeBundle is described by a TimeMap, which is a specialization of an ORE Resource Map. A TimeMap lists all URI-M_i for a given URI-R as well as their associated metadata including timestamp. It also lists the Original Resource URI-R and its TimeGate URI-G. Appendix A shows an example RDF/XML TimeMap; other serializations such as Turtle and Atom are possible. Discovery of TimeBundles is supported by the rel value *timebundle* in the HTTP Link response header.

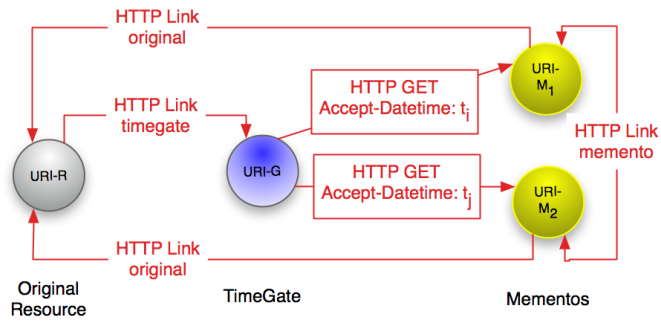

Figure 2: The Memento Framework.

[3] http://mementoweb.org/
[4] http://archive.org/
[5] http://webcitation.org/

Three aspects of the Memento architecture ensure that the globally deployed HTTP caching infrastructure can be leveraged. First, the Original Resource URI-R and its TimeGate URI-G are always separate resources: URI-R is a conventional resource and URI-G is dedicated to datetime conneg. This eliminates caching problems that would be caused by transitioning URI-R between non-negotiable and negotiable if URI-R and URI-G were to coincide. Second, the initial Memento architecture [19], required the Original Resource URI-R to 302 redirect to its TimeGate URI-G; as a result, cached versions of URI-R could not be leveraged. By using the *timegate* HTTP Link for discovery of URI-G, Memento clients work with caches instead of against them. Third, URI-G and URI-M are never the same resource so the Mementos (URI-M) can be cached as well.

A detailed overview of HTTP request/response scenarios is available in the Memento HTTP Transactions Guide[6]. Here, we highlight certain aspects related to HTTP interactions with the TimeGate URI-G. A choice was made to handle cases in which URI-G is dereferenced without the Accept-Datetime header, by issuing a "302 Found" redirect to the most recent Memento, as opposed to offering a list of choices to the client. While a list would be feasible for a top-level resource (say, an HTML page), it would be cumbersome for the potentially many embedded resources (say, the images in the HTML page). URI-G will only return HTTP response code "300 Multiple Choices" if explicitly requested with a "Negotiate: 1.0" request header or when there are multiple Mementos with the same Content-Datetime[7]. URI-G will return HTTP response code "406 Not Acceptable" when the Accept-Datetime is outside of the datetime range of known Mementos. For further technical details about the Memento framework, we refer to the original paper [19], and the more recent overview of the evolved solution [20] that has resulted from feedback to the original ideas provided by both the Linked Data and Web Archiving communities.

Since its publication, Memento has received significant attention. Major Web Archives have started implementing support[8], and work is ongoing to develop support for common CMS platforms such as MediaWiki and Drupal. Also, the establishment of a Memento-track at the JISC Developer Days (Dev8D)[9] organized by the UK's Joint Information Systems Committee is an early indication of interest by both funders and implementers.

As an illustration, Figure 3 shows a Memento HTTP flow whereby a client requests a November 8 2009 version of the Wikipedia page for DJ Shadow, by interacting with its current URI `http://en.wikipedia.org/wiki/DJ_Shadow`; the client is pointed by `http://en.wikipedia.org/wiki/DJ_Shadow` to a TimeGate at Wikipedia; via that TimeGate the client successfully retrieves a Memento that meets its datetime preferences (only headers crucial to convey an understanding of datetime conneg are shown). We should point out that Wikipedia has not (yet) implemented such Memento HTTP flows, but a MediaWiki plug-in that adds Memento support is available[10].

[6] `http://www.mementoweb.org/guide/http/`

[7] This may occur as HTTP only supports second-level time granularity

[8] See Agenda of First Memento Implementation Meeting at `http://mementoweb.org/events/IA201002/`

[9] `http://wiki.2010.dev8d.org/w/Talk_6`

[10] Memento MediaWiki plug-in `http://www.mediawiki.org/wiki/Extension:Memento`

In Figure 3, note the use of the HTTP Link header to express the very first and most recent Mementos available from Wikipedia (*rel="first-memento"* and *rel="last-memento"*, respectively) as well as the Mementos that are closest in time (*rel="prev-memento"* and *rel="next-memento"*) to the one that is returned. Note also the use of a HTTP Link header to point back to the Original Resource (*rel="original"*).

4. MEMENTO RESOURCE VERSIONING

The Memento framework suggests a versioning mechanism that is fully based on HTTP (see Figure 4). These are its core characteristics:

- *Identification*: HTTP URIs are used to identify versions of resources.
- *Versioning Strategy*: The top of Figure 4 shows URI-R as the resource from which at any point in time the current representation is served, and URI-M$_i$ as resources that provide access to representations that were previously available from URI-R. In terms of [2] and its associated ontology[11], URI-R is a *time-generic* resource, whereas all URI-M$_i$ are *time-specific* resources. This strategy is different than the one shown in Figure 1, yet aligned with the stable URI principle of Cool URIs [3, 18]: instead of minting a new URI for every **new** version, keep the URI stable and mint new URIs for **old** versions. This approach has become rather widespread; for example, `http://cnn.com` and `http://en.wikipedia.org/wiki/DJ_Shadow` are such URI-R, whereas `http://web.archive.org/web/20010911203610/http://www.cnn.com` and `http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=337446696` are examples of respective URI-M$_i$.
- *Version Relationships*: A *timegate* HTTP Link header provided in response to GET/HEAD requests issued against the stable URI-R points at a TimeGate. And, an *original* HTTP Link header provided in response to GET/HEAD requests issued at Mementos URI-M$_i$ points at URI-R of the Original Resource. As described, a TimeGate supports datetime conneg based on the content of the *Accept-Datetime* header. It is a *time-travel* resource that acts as a gateway between the time-generic URI-R and its associated time-specific Mementos URI-M$_i$. The result of the datetime conneg is a Memento that meets the expressed datetime preference. Also, the *prev-memento* and *next-memento* relationships may be used in the HTTP Link header to point at Mementos that are adjacent in time to the returned one.
- *Version Timestamping*: Versioning of URI-R is not required as it always is the current version. Mementos URI-M$_i$ are timestamped by means of the *Content-Datetime* response header.

The technology substrate used by the Memento versioning approach is fully centered on HTTP: URI, HTTP, HTTP Link with to-be-registered link relationships, HTTP datetime conneg. The challenges formulated in the Introduction can be addressed as follows (bottom of Figure 4):

[11] `http://www.w3.org/2006/gen/ont`

```
HEAD http://en.wikipedia.org/wiki/DJ_Shadow HTTP/1.1
Accept-Datetime: Sun, 08 Nov 2009 00:00:00 GMT

HTTP/1.1 200 OK
Link: <http://en.wikipedia.org/Special:TimeGate/http://en.wikipedia.org/wiki/DJ_Shadow>
       ; rel="timegate"

GET http://en.wikipedia.org/Special:TimeGate/http://en.wikipedia.org/wiki/DJ_Shadow
     HTTP/1.1
Accept-Datetime: Sun, 08 Nov 2009 00:00:00 GMT

HTTP/1.1 302 Found
TCN: choice
Vary: negotiate, Accept-Datetime
Location: http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=324178040
Link: <http://en.wikipedia.org/wiki/DJ_Shadow>; rel="original"
Alternates:
 {"http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=1493688" 0.01
  {type text/html} {language en} {dt "Sun, 28 Sep 2003 01:42:00 GMT" first}},
 {"http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=337446696" 0.01
  {type text/html} {language en} {dt "Tue, 12 Jan 2010 19:55:00 GMT" last}},
 {"http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=324178040" 1.0
  {type text/html} {language en} {dt Thu, 05 Nov 2009 23:41:00 GMT}},
 {"http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=322586071" 0.8
  {type text/html} {language en} {dt "Wed, 28 Oct 2009 14:307:00 GMT" prev}},
 {"http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=326164283" 0.7
  {type text/html} {language en} {dt "Thu, 26 Nov 2009 23:50:00 GMT" next}}

GET http://en.wikipedia.org/w/index.php?title=DJ_Shadow&oldid=324178040
Accept-Datetime: Sun, 08 Nov 2009 00:00:00 GMT

HTTP/1.1 200 OK
Content-Datetime: Thu, 05 Nov 2009 23:41:00 GMT
Link: <http://en.wikipedia.org/wiki/DJ_Shadow>; rel="original"
Alternates: ...
```

Figure 3: Memento HTTP Request/Response Cycle.

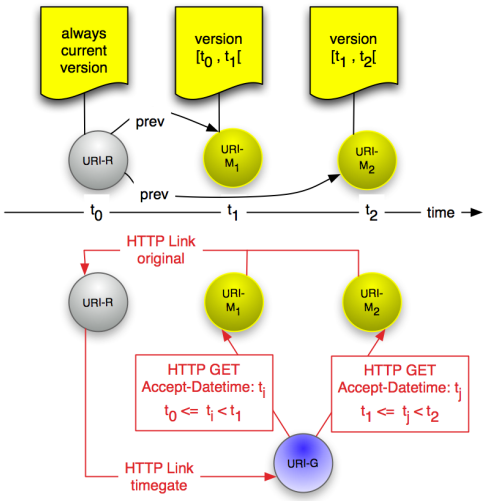

Figure 4: Memento Resource Versioning.

1. *Given the current version of a resource, how can "follow your nose" style navigation to prior versions of the resource be achieved?* The current version is the stable URI-R. A client application can follow its nose to a TimeGate for URI-R by using the URI that is expressed in the *timegate* HTTP Link header returned by URI-R. The TimeGate supports datetime conneg allowing the client to obtain various versions (Mementos) by varying the content of its *Accept-Datetime* request header. In addition, in cases where the *prev-memento* and *next-memento* relationships are available in the Alternates header provided in TimeGate responses, a client can engage in version-to-version navigation with a certainty that the version-relationships are time-based.

2. *Given any version of a resource and a particular timestamp, how can "follow your nose" style navigation towards another version that matches the timestamp be achieved?* A version URI-M$_i$ provides an *original* HTTP Link header pointing at the stable URI-R. From thereon, this scenario is the same as described in the previous point; the timestamp is used as the content of the *Accept-Datetime* request header. The *Content-Datetime* provides the earliest datetime at which the returned version became available; that version was still the then-current one at the datetime that was expressed in the *Accept-Datetime* header.

5. MEMENTO RESOURCE VERSIONING AND LINKED DATA

Figure 5 shows how Memento integrates in the Linked Data environment. In this case, URI-R is a cool URI for a non-information resource [18], and the current description

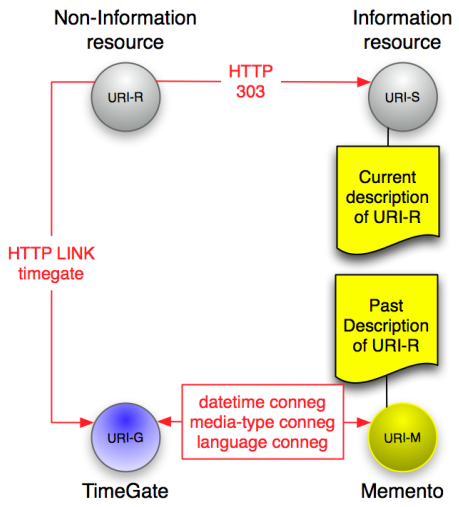

Figure 5: Memento Resource Versioning and Linked Data.

Table 1: DBpedia Demonstrator Database Table.

id	subject	start	end	triples
integer, auto_increment, not null, PK	varchar(256)	datetime	datetime	blob

of URI-R is available at URI-S. A TimeGate URI-G is introduced for URI-R, and to support its discovery a *timegate* HTTP Link header is provided in responses to GET/HEAD requests to URI-R. When a Linked Data client is in need of prior descriptions of URI-R, it follows its nose to URI-G, where it can use datetime conneg to arrive at a description of URI-R as it existed at some time in the past. Note that the conneg with URI-G can include dimensions other than datetime. The media-type dimension that is commonly used in Linked Data to allow a choice of descriptions expressed in RDF serializations or HTML can also be supported. Similarly, negotiation on language can be supported.

6. THE DBPEDIA DEMONSTRATOR

6.1 Demonstrator Set-Up

We have implemented the architecture depicted in Figure 5 in the DBpedia context. We first downloaded the five prior English-language versions of DBpedia (2.0 through 3.3)[12] in NT format. Using a python script, the approximately 600 Million triples were loaded into a MySQL table (Table 1). Loading took approximately 15 hours, and resulted in a MyISAM table of 81 GB.

For each DBpedia subject URI-R, we exposed a TimeGate to support content negotiation in the datetime and media-type dimensions. For example, our TimeGate for DBpedia's France resource `http://dbpedia.org/resource/France` is `http://mementoarchive.lanl.gov/dbpedia/timegate/http://dbpedia.org/resource/France`. The datetime functionality was implemented by retrieving all distinct start/end combinations for the requested subject URI-R. This approach readily supports providing the *first-memento*, *last-memento*, *next-memento* and *prev-memento* relationships in the HTTP Link header provided in responses. The returned Memento is the one with the start/end interval that covers the datetime requested via conneg. For conneg requests with a datetime value that is in the range of the current DBpedia version, the TimeGate issues an HTTP 302 redirect to the Original Resource URI-R at `dbpedia.org`. Mementos are available both in HTML and RDF/XML. For example, the Memento for DBpedia 3.3's France resource in HTML is `http://mementoarchive.lanl.gov/dbpedia/memento/20090701/http://dbpedia.org/page/France`.

Colleagues at DBpedia kindly implemented the *timegate* HTTP Link header pointing at our TimeGates. This required approximately one hour and consisted of adding a stored procedure in the OpenLink Virtuoso engine to add the appropriate HTTP Link header.

Figure 6 shows a Memento HTTP flow whereby a client requests the description of France that was available from DBpedia on March 20, 2008. Only headers crucial to convey an understanding of the conneg are shown.

To illustrate full Memento compliance, we also implemented TimeBundle/TimeGate support for our DBpedia version archive. This functionality was not used to achieve the time-series analysis described below. The Appendix shows our TimeMap for the DBpedia resource `http://dbpedia.org/resource/France`; the content should be self-explanatory.

Although DBpedia currently operates under a regime of recurrent discreet updates, both the proposed Memento approach and our specific database design support a possible future regime in which DBpedia is updated on an ongoing basis. In this case, an archiving mechanism would need to be added to ensure that versions of distinct DBpedia descriptions are pushed/pulled into the version archive as they change.

6.2 Time-Series Analysis using Memento Resource Versioning

To illustrate the power of the proposed approach, we implemented a simple time-series analysis using both past and current DBpedia data. We set out to trace the evolution over time of the Gross Domestic Product Per Capita for various countries, leveraging the `http://dbpedia.org/property/gdpPppPerCapita` property.

The straightforward data-time-traveling algorithm used to construct the time-series data-set is described by the below pseudo code. It must be noted that the actual script must rely on some ad-hoc heuristics to deal with diverging data formats used for GDP values.

```
resources := [list of country description TimeGate URIs]
times := [list of date times, one per version including current]
prop := "http://dbpedia.org/property/gdpPppPerCapita"
values := {}

foreach r in resources:
   values[r] := []
   foreach t in times:
       data := fetch(URI-TG/r, Accept-Datetime: t, Accept:
"application/rdf+xml")
       graph := parse(data)
       value := graph.sparql(SELECT val WHERE { r prop ?val . })
       value := normalize(value)
       values[r].push(value)
```

[12] `http://wiki.dbpedia.org/Downloads34`

```
HEAD http://dbpedia.org/resource/Paris HTTP/1.1
Accept: application/rdf+xml
Accept-Datetime: Thu, 20 Mar 2008 18:00:00 GMT

HTTP/1.1 200 OK
Link: <http://mementoarchive.lanl.gov/dbpedia/timegate/http://dbpedia.org/resource/France>
  ; rel="timegate"

GET http://mementoarchive.lanl.gov/dbpedia/timegate/http://dbpedia.org/resource/France HTTP/1.1
Accept: application/rdf+xml
Accept-Datetime: Thu, 20 Mar 2008 18:00:00 GMT

HTTP/1.1 302 Found
TCN: choice
Vary: negotiate, accept, accept-datetime
Location: http://mementoarchive.lanl.gov/dbpedia/20080201/http://dbpedia.org/resource/France
Link: <http://dbpedia.org/resource/France>; rel="original",
 <http://mementoarchive.lanl.gov/dbpedia/memento/20070901/http://dbpedia.org/data/France>;
  datetime="Sat, 01 Sep 2007 00:00:00 GMT"; rel="first-memento",
 <http://mementoarchive.lanl.gov/dbpedia/memento/20091101/http://dbpedia.org/data/France>;
  datetime="Sun, 01 Nov 2009 00:00:00 GMT"; rel="last-memento",
 <http://mementoarchive.lanl.gov/dbpedia/memento/20070901/http://dbpedia.org/data/France;
  datetime="Sat, 01 Sep 2007 00:00:00 GMT"; rel="prev-memento",
 <http://mementoarchive.lanl.gov/dbpedia/memento/20000801/http://dbpedia.org/data/France>;
  datetime="Fri, 01 Aug 2008 00:00:00 GMT"; rel="next-memento",
 <http://mementoarchive.lanl.gov/dbpedia/timebundle/http://dbpedia.org/resource/France>;
  rel="timebundle"

GET http://mementoarchive.lanl.gov/dbpedia/20080201/http://dbpedia.org/resource/France HTTP/1.1
Accept: application/rdf+xml
Accept-Datetime: Thu, 20 Mar 2008 18:00:00 GMT

HTTP/1.1 200 OK
Content-Datetime: Fri, 01 Feb 2008 00:00:00 GMT
Content-Type: application:rdf+xml
Link: <http://dbpedia.org/resource/France>; rel="original",
 <http://mementoarchive.lanl.gov/dbpedia/memento/20070901/http://dbpedia.org/data/France>;
  datetime="Sat, 01 Sep 2007 00:00:00 GMT"; rel="first-memento", ...
```

Figure 6: Memento DBpedia HTTP Request/Response Cycle.

The collected data were then turned into a chart (Figure 7) using the Google Chart API[13].

7. RELATED WORK

Little research has explored a protocol-based solution to augment the Web with time travel capabilities. TTApache [5] introduced an ad-hoc RPC-style mechanism to access archived representations given the URI of their original, e.g. "page.html?02-Nov-2009". This approach reveals the local scope of the problem addressed by TTApache, as opposed to the global perspective taken by the Memento datetime conneg framework. Indeed, the query components are issued against a specific server, and are not maintained when a client moves to another server as is the case with the Accept-Datetime header of datetime conneg. TTApache also allowed addressing archived representations using version numbers in query components rather than datetimes. This capability is similar to the deprecated "Content-Version" header field from RFC 2068 [6] and other, similar expired proposals (e.g., [16]). Such versioning features have not found wide-spread adoption, presumably because their address space is tied to a specific resource or server, and not universal like datetime. TTApache also provided support for reserved terms as query components such as "page.html?now". This capability is similar to link relationship types such as "latest-version", "predecessor-version", "successor-version", and "working-copy-of" proposed in [4] to allow simple version navigation between Web resources. The focus of this proposal that emerged from the AtomPub [7] context, however, is clearly on editorial version control (cf. WebDav, Java Content Repository). Also, it provides no means to navigate versions based on datetime information.

There is a relationship between the described work and efforts that research the problem of provenance of Linked Data, specifically those provenance aspects concerned with the time intervals in which specific data is valid. For example, [8], is concerned with provenance graphs that allow expressing such validity information, whereas [9] focuses on applications to support preserving link integrity over time. Our proposal introduces a native HTTP approach that allows leveraging the results of these efforts at Web scale.

8. CONCLUSIONS

URIs like `http://weather.example.com/oaxaca` used in [11] have gained significant functionality in the Linked Data context as they start providing access not only to HTML intended for human consumption, but also to data expressed in some RDF serialization intended for machine processing. When publishing data in accordance to Memento's HTTP-based versioning mechanism proposed in this paper, their

[13] `http://code.google.com/apis/charttools/index.html`

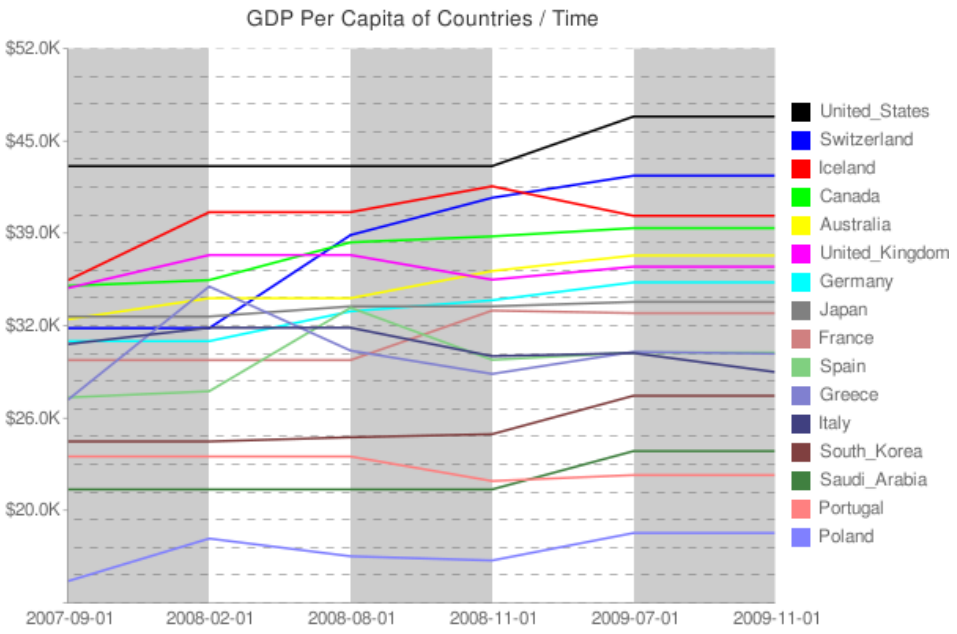

Figure 7: A time-series analysis conducted across DBpedia versions using Memento's HTTP-based versioning approach.

value further increases as they become entry points to both current and past versions of data. The time-series analysis described in Section 6.2 is an admittedly simple demonstration of a subtle and powerful change in the utility of Linked Data URIs.

The URI `http://weather.example.com/oaxaca` can now be leveraged to obtain an overview of Oaxaca's weather in the past months, merely by issuing HTTP GET requests with varying datetime preferences. Similarly, time-traveling a Dow Jones data URI can result in an overview of the stock market's evolution at any desired granularity. Tracing the evolving state of traffic congestions, implemented in Zoetrope [1] by high-frequency crawls and scraping of a traffic web site could be achieved by dereferencing a single data URI with varying timestamps instead.

While this paper has focused on Linked Data, it should be clear that the proposed versioning mechanism can be applied to Web resources in general. It could, for example, be leveraged to facilitate navigating across issues of Web-based newspapers and magazines, and it can play an important role in better integrating the data-intensive eScience and eHumanities efforts into the Web. Hence, the addition of a time dimension to the Web is not something only digital archaeologists should care about. It is an enabler for a global HTTP-based versioning mechanism that can support a new range of temporal applications for both the document and the data Web. This paper has merely scratched the surface of a new world of possibilities.

9. ACKNOWLEDGMENTS

The Memento research is partly funded by the Library of Congress. Many thanks to Chris Bizer, Kinsley Idehen, and Mitko Iliev for implementing the *timegate* HTTP Link header in DBpedia.

10. REFERENCES

[1] E. Adar, M. Dontcheva, J. Fogarty, and D. S. Weld. Zoetrope: interacting with the ephemeral web. In *UIST '08: Proceedings of the 21st annual ACM symposium on User interface software and technology*, pages 239–248, 2008.

[2] T. Berners-Lee. Web architecture: Generic resources. http://www.w3.org/DesignIssues/Generic.html, 1996.

[3] T. Berners-Lee. Cool URIs don't change. 1998. http://www.w3.org/Provider/Style/URI.html.

[4] A. Brown, G. Clemm, and J. Reschke. Link relation types for simple version navigation between web resources, Internet Draft, 2010.

[5] C. E. Dyreson, H. Lin, and Y. Wang. Managing versions of web documents in a transaction-time web server. In *WWW '04: Proceedings of the 13th international conference on World Wide Web*, pages 422–432, 2004.

[6] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee. Hypertex Transfer Protocol – HTTP/1.1, Internet RFC-2068, 1997.

[7] J. Gregorio and B. de hOra. The Atom publishing protocol, Internet RFC-5023, December 2007.

[8] O. Hartig. Provenance information in the web of data. In *Proceedings of the 2nd Workshop on Linked Data on the Web (LDOW2009)*, 2009.

[9] B. Haslhofer and N. Popitsch. DSNotify–Detecting and Fixing Broken Links in Linked Data Sets. In *Proceedings of 8th International Workshop on Web Semantics*, 2009.

[10] K. Holtman and A. Mutz. Transparent Content Negotiation in HTTP, Internet RFC-2295, 1998.

[11] I. Jacobs and N. Walsh. Architecture of the world wide web, volume one. Technical Report W3C

Recommendation 15 December 2004, W3C, 2004.

[12] C. Lagoze, H. Van de Sompel, P. Johnston, M. L. Nelson, R. Sanderson, and S. Warner. Adding eScience Assets to the Data Web. In *Proceedings of the Linked Data on the Web Workshop (LDOW 2009)*, 2009.

[13] N. Mendelsohn and S. Williams. The use of Metadata in URIs, TAG Finding 2 January 2007, 2007.

[14] M. Nottingham. Web linking, Internet Draft, 2010.

[15] M. Nottingham and R. Sayre. The Atom Syndication Format, Internet RFC-4287, 2005.

[16] K. Ota, K. Takahashi, and K. Sekiya. Version management with meta-level links via HTTP/1.1, Internet Draft draft-ntt-http-version-00, 1996.

[17] D. Raggett, A. L. Hors, and I. Jacobs. HTML 4.01 Specification. Technical Report W3C Recommendation 24 December 1999, 1999.

[18] L. Sauermann and R. Cyganiak. Cool URIs for the semantic web. Technical Report W3C Interest Group Note 31 March 2008, W3C, 2008.

[19] H. Van de Sompel, M. L. Nelson, R. Sanderson, L. L. Balakireva, S. Ainsworth, and H. Shankar. Memento: Time Travel for the Web. Technical Report arXiv:0911.1112, 2009.

[20] H. Van de Sompel, M. L. Nelson, R. Sanderson, L. L. Balakireva, S. Ainsworth, and H. Shankar. Memento: Updated Technical Details (February 2010). http://www.slideshare.net/hvdsomp/memento-updated-technical-details-february-2010, 2010.

APPENDIX

A. AN RDF/XML TIMEMAP

The following is an RDF/XML TimeMap for `http://dbpedia.org/resource/France`.


```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF
  xmlns:foaf='http://xmlns.com/foaf/0.1/'
  xmlns:dcterms='http://purl.org/dc/terms/'
  xmlns:mem='http://www.mementoweb.org/terms/tb/'
  xmlns:dc='http://purl.org/dc/elements/1.1/'
  xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'
  xmlns:ore='http://www.openarchives.org/ore/terms/'
  xmlns:rdfs1='http://www.w3.org/2001/01/rdf-schema#'>
<ore:ResourceMap rdf:about="http://mementoarchive.lanl.gov/dbpedia/timemap/rdf/http://dbpedia.org/resource/France">
<rdf:type rdf:resource="http://www.mementoweb.org/terms/tb/TimeMap"/>
<dcterms:modified>2010-02-17T05:26:27Z</dcterms:modified>
<dcterms:created>2010-02-17T05:26:27Z</dcterms:created>
<dc:format>application/rdf+xml</dc:format>
<dcterms:creator>
<rdf:Description rdf:about="http://foresite-toolkit.googlecode.com/#pythonAgent">
<foaf:mbox>foresite@googlegroups.com</foaf:mbox>
<foaf:name>Foresite Toolkit (Python)</foaf:name>
</rdf:Description>
</dcterms:creator>
<ore:describes>
<ore:Aggregation rdf:about="http://mementoarchive.lanl.gov/dbpedia/timebundle/http://dbpedia.org/resource/France">
<ore:aggregates rdf:resource="http://mementoarchive.lanl.gov/dbpedia/memento/20070901/http://dbpedia.org/data/France"/>
<ore:aggregates rdf:resource="http://mementoarchive.lanl.gov/dbpedia/memento/20080201/http://dbpedia.org/data/France"/>
<ore:aggregates rdf:resource="http://mementoarchive.lanl.gov/dbpedia/memento/20080801/http://dbpedia.org/data/France"/>
<ore:aggregates rdf:resource="http://mementoarchive.lanl.gov/dbpedia/memento/20081101/http://dbpedia.org/data/France"/>
<ore:aggregates rdf:resource="http://mementoarchive.lanl.gov/dbpedia/memento/20090701/http://dbpedia.org/data/France"/>
<ore:aggregates rdf:resource="http://mementoarchive.lanl.gov/dbpedia/timegate/http://dbpedia.org/resource/France"/>
<ore:aggregates rdf:resource="http://dbpedia.org/resource/France"/>
<dc:title>Memento Time Bundle for http://dbpedia.org/resource/France</dc:title>
<rdf:type rdf:resource="http://www.mementoweb.org/terms/tb/TimeBundle"/>
</ore:Aggregation>
</ore:describes>
</ore:ResourceMap>
```

```
  <rdf:Description rdf:about="http://www.openarchives.org/ore/terms/Aggregation">
    <rdfs1:label>Aggregation</rdfs1:label>
    <rdfs1:isDefinedBy rdf:resource="http://www.openarchives.org/ore/terms/"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://www.openarchives.org/ore/terms/ResourceMap">
    <rdfs1:label>ResourceMap</rdfs1:label>
    <rdfs1:isDefinedBy rdf:resource="http://www.openarchives.org/ore/terms/"/>
  </rdf:Description>
  <mem:TimeGate rdf:about="http://mementoarchive.lanl.gov/dbpedia/timegate/http://dbpedia.org/resource/France">
    <mem:timeGateFor rdf:resource="http://dbpedia.org/resource/France"/>
    <mem:covers rdf:nodeID="gtKXRtug21"/>
  </mem:TimeGate>
 <mem:Period rdf:nodeID="gtKXRtug21">
    <mem:end rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2009-11-01T00:00:00+00:00</mem:end>
    <mem:start rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2007-09-01T00:00:00+00:00</mem:start>
  </mem:Period>
  <mem:Memento rdf:about="http://mementoarchive.lanl.gov/dbpedia/memento/20070901/http://dbpedia.org/data/France">
    <mem:validOver>
      <mem:Period>
        <mem:start rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2007-09-01T00:00:00+00:00</mem:start>
        <mem:end rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2008-02-01T00:00:00+00:00</mem:end>
      </mem:Period>
    </mem:validOver>
    <mem:mementoFor rdf:resource="http://dbpedia.org/resource/France"/>
  </mem:Memento>
  <mem:Memento rdf:about="http://mementoarchive.lanl.gov/dbpedia/memento/20080201/http://dbpedia.org/data/France">
    <mem:validOver>
      <mem:Period>
        <mem:start rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2008-02-01T00:00:00+00:00</mem:start>
        <mem:end rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2008-08-01T00:00:00+00:00</mem:end>
      </mem:Period>
    </mem:validOver>
    <mem:mementoFor rdf:resource="http://dbpedia.org/resource/France"/>
  </mem:Memento>
  <mem:Memento rdf:about="http://mementoarchive.lanl.gov/dbpedia/memento/20080801/http://dbpedia.org/data/France">
    <mem:validOver rdf:nodeID="gtKXRtug42"/>
    <mem:mementoFor rdf:resource="http://dbpedia.org/resource/France"/>
  </mem:Memento>
  <mem:Period rdf:nodeID="gtKXRtug42">
    <mem:end rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2008-08-01T00:00:00+00:00</mem:end>
    <mem:start rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2009-11-01T00:00:00+00:00</mem:start>
  </mem:Period>
  <mem:Memento rdf:about="http://mementoarchive.lanl.gov/dbpedia/memento/20081101/http://dbpedia.org/data/France">
    <mem:validOver rdf:nodeID="gtKXRtug49"/>
    <mem:mementoFor>
      <mem:OriginalResource rdf:about="http://dbpedia.org/resource/France"/>
    </mem:mementoFor>
  </mem:Memento>
  <mem:Period rdf:nodeID="gtKXRtug49">
    <mem:end rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2008-11-01T00:00:00+00:00</mem:end>
    <mem:start rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2009-07-01T00:00:00+00:00</mem:start>
  </mem:Period>
  <mem:Memento rdf:about="http://mementoarchive.lanl.gov/dbpedia/memento/20090701/http://dbpedia.org/data/France">
    <mem:validOver rdf:nodeID="gtKXRtug56"/>
    <mem:mementoFor rdf:resource="http://dbpedia.org/resource/France"/>
  </mem:Memento>
  <mem:Period rdf:nodeID="gtKXRtug56">
    <mem:end rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2009-07-01T00:00:00+00:00</mem:end>
    <mem:start rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2009-11-01T00:00:00+00:00</mem:start>
  </mem:Period>
</rdf:RDF>
```