Data Documentation Initiative

Best Practice

Subject

DDI 3.0 URNs and Entity Resolution (2009-03-21)

Document identifier:

DDIBestPractices_URNsAndEntityResolution.doc.PDF

Location:

http://www.ddialliance.org/bp/DDIBestPractices_URNsAndEntityResolution.doc.PDF

Authors:

Nikos Askitas, Janet Eisenhauer, Arofan Gregory, Rob Grim, Pascal Heus, Maarten Hoogerwerf, Wolfgang Zenk-Möltgen

Editors:

Nikos Askitas, Maarten Hoogerwerf

Abstract:

This document is not quite a best practice but rather a recommendation about appropriate architecture for the effective resolution of DDI URNs[1]. The recommended architecture is based on standard and tested technologies put together in order to facilitate URN resolution needs. Along the way we describe the consequences for the various parties involved and the relationship of DDI URNs to other resolution mechanisms. This document does not deal with the latter in depth, as that would be the subject of a white paper in its own right.

Status:

This document is updated periodically on no particular schedule. Send comments to editor: ddi-bp-editors@icpsr.umich.edu

[1] All URNs in this document are assumed to be DDI 3.0 URNs. Unless otherwise stated, we simply say “URN” and mean “DDI 3.0 URN”.

Introduction

This best practice lays out an architecture and solution to the issue of providing unique persistent identifiers for DDI entities down to the variable level.

1.1 Problem statement

A key feature of DDI 3.0 is the use of URN-based references for the purpose of identification, discovery, and reusability. To enable this to the right granularity, many URNs need to be assigned. This implies scale, mass, and volume issues that need to be addressed and solved. To answer these issues effectively, an architectural infrastructure must have the following core properties:

1. High availability, reliability, durability
2. Scalability, manageability on all levels (DDI Alliance, institution, etc.)
3. Non-invasive, low barrier, compatibility, and interoperability
4. Reliance on standard and tested general purpose technologies
5. Community-based (archival, library, academic, quantitative data)

1.2 Terminology

The key words *must, must not, required, shall, shall not, should, should not, recommended, may,* and *optional* in this document are to be interpreted as described in **[RFC2119].** Additional DDI standard terminology and definitions are found in http://www.ddialliance.org/definitions/.

2 Best Practice Solution

2.1 Definitions

DNS: The Domain Name System (DNS) translates Internet domain and host names to IP addresses. It translates domain names meaningful to humans into the numerical (binary) identifiers associated with networking equipment for the purpose of locating and addressing these devices world-wide. An often used analogy to explain the Domain Name System is that it serves as the "phone book" for the Internet by translating human-friendly computer hostnames into IP addresses. For example, www.example.com translates to 208.77.188.166.

HTTP: Short for HyperText Transfer Protocol, the underlying protocol used by the World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands. For example, when you enter a URL in your browser, this actually sends an HTTP command to the Web server directing it to fetch and transmit the requested Web page.

HTTPS: HTTPS stands for HyperText Transfer Protocol over SSL (Secure Socket Layer). It is a TCP/IP protocol used by Web servers to transfer and display Web content securely. The data transferred are encrypted so that they cannot be read by anyone except the recipient.

IP: An Internet Protocol (IP) address is a numerical identification (logical address) that is assigned to devices participating in a computer network utilizing the Internet Protocol for communication between its nodes. Although IP addresses are stored as binary numbers, they are usually displayed in human-readable notations, such as 208.77.188.166 (for IPv4).

URL: A URL (Uniform Resource Locator, previously Universal Resource Locator) is the unique address for a file that is accessible on the Internet. A common way to get to a Web site is to enter the URL of its home page file in a Web browser’s address line. However, any file within that Web site can also be specified with a URL.

URN: A URN (Uniform Resource Name) is an Internet resource with a name that, unlike a URL, has persistent significance -- that is, the owner of the URN can expect that someone else (or a program) will always be able to find the resource.

TTL: Short for Time to Live, a field in the Internet Protocol (IP) that specifies how many more hops a packet can travel before being discarded or returned.

2.2 Best Practice behavior

General description

The proposed solution will be used for resolution of a great number of DDI URNs[2]. Combined with the requirements mentioned in paragraph 1.1 “Problem statement,” this means delegation of both responsibility and resolution. An existing system that implements such delegation and is well proven is DNS. Rather than rebuilding or copying DNS, the proposed infrastructure will use it to meet the requirements of the solution.

In addition to DNS, the proposed architecture will draw upon highly used techniques such as (secured) HTTP, URLs, and query strings. Reuse of this technology acknowledges the concepts of DDI.

The solution is explained here using a schematic overview. This overview demonstrates both the components of the solution and the resolution process. The overview uses ICPSR as an example, but in general one can substitute any DDI Agency wherever “ICPSR” appears.

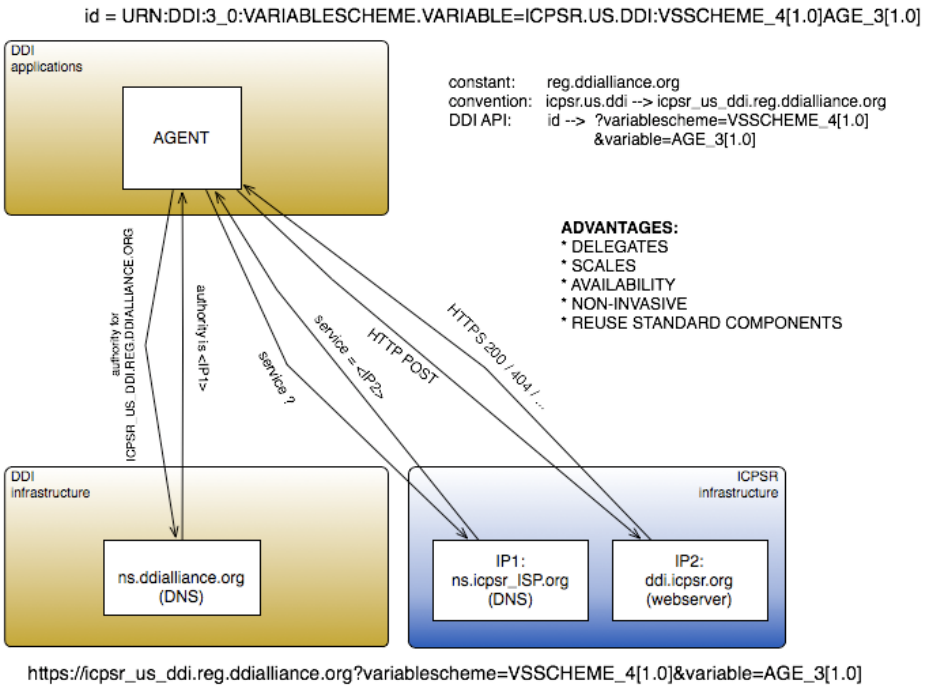

Figure 1: Overview of DNS-based resolution

Description of required infrastructure

Figure 1 shows an infrastructure consisting of three components described below:

[2] An estimation for the amount of resolvable URNs is 1.500.000.000 (500 agencies each having 500 studies that each contain 2000 variables that each have 3 referred schemes)

- DDI applications. A typical situation is that a service or application wants to retrieve the referred material to show it to an end user. All DDI aware applications should have a central agent that takes care of resolution of the URNs.

- DDI Infrastructure. The DDI Alliance will need control over a name server and a domain name where it can register various sub domains for each registered DDI Agency: `<agency_id.reg>.ddialliance.org`. Redundancy or load balancing of this infrastructure can require agreements with certain agencies.[3]

- Agency Infrastructure. Each agency needs to be able to register some DNS records in the name server of their ISP. In addition, they need their DDI-exposing application to implement an API that specifies the requests for certain concepts, studies, variables, questions, versions, etc. The API should also specify functionality like authentication and a TTL mechanism.

Description of resolution process

Assume that the agent from a researcher's client-application needs to resolve the following URN: `URN:DDI:3_0:VARIABLESCHEME.VARIABLE=ICPSR.US.DDI`[4]`:VSSCHEME_4[1.0]AGE_3[1.0]`

The following steps will result in the researcher's application's agent retrieving the metadata identified by this URN .

1. The DDI application recognizes ICPSR.US.DDI as the DDI Agency and queries DNS for ICPSR_US_DDI.REG.DDIALLIANCE.ORG, where it can resolve this URN.

2. The name server of the DDI Alliance responds with the name of the name server of ICPSR, which is NS.ICPSR.UMICH.EDU.

3. The name server of ICPSR maintains a list of services and the locations of these. The agent DDI application now requests the DDI resolution service and receives one or more IP address of the machines that host this service.

4. The agent now translates the URN into a query string that will request the identified data from the ICPSR DDI Registry. In a nutshell, this transformation will result in "?variablescheme=VSSCHEME_4[1.0]&variable=AGE_[1.0]". The details about this transformation can be found in section 2.5 "Discussion".

[3] See Discussion.

[4] The DDI Alliance is working with the community and its partners to establish a global registry of agencies that will be publicly available online -- see http://tools.ddialliance.org/?lvl1=community&lvl2=agencyid. It is crucial for early adopters of DDI 3.0 to immediately start using unique global identifiers. Therefore, there is now an unofficial pre-registration form that will allow organizations to request and reserve an identifier in the future registry. The conventions in the registry are of the form icpsr.us.ddi, odesi.ca.ddi, gesis.de.ddi, etc.

5. The agent makes a request of the DDI registry for the identified material using a HTTP or a HTTPS POST command like:
   http://<ddisvc>.icpsr_us_ddi.reg.ddialliance.org/?variablescheme=VSSCHEME_4[1.0]&variable=AGE_[1.0]

   Additional parameters can be required: authentication to allow authorization, time-to-live to prevent circular references, etc.

6. The DDI registry uses the request parameters and determines whether the requested material is available and whether the requesting agent is authorized to view it. It will send its response via HTTP(S) and contain HTTP response codes and DDI XML (possibly contained within an XML container to facilitate additional parameters).

2.3 Implications

- The ddialliance.org creates a subdomain called reg.ddialliance.org, which maintains a domain name for each registered Agency (this implies modifying the current tentative registry appropriately). For example, this means that the current entry for ICPSR (icpsr.us.ddi) should result in the subdomain icpsr_us_ddi.reg.ddialliance.org to be registered as a record `dns*.ddialliance.org` in the name servers of the DDI Alliance.

- Each DDI provider registers their DDI service(s) in their own (or their Internet provider's) DNS servers. For example ICPSR needs to register at least one SRV record in their name server NS.ICPSR.UMICH.EDU

- The DDI Alliance should encourage the development of reusable libraries that take care of the resolution and/or implement the API for querying DDI providers. This will improve the quality of the resolution and lower the barriers for adopting the DDI 3.0 standard.

2.4 Validity

The choice for DNS allows the use of highly standardized protocols and software. Its reliability is proven by the current Internet infrastructure. It allows redundancy and scalability. It avoids the need to build software and can reuse existing hardware. There is little additional administration required (since the DDI already maintains a registry).

2.5 Discussion

Agency identifiers

Agency identifiers can be reused to register subdomain records in DNS. Currently, the Agency identifiers are structured like ICPSR.US.DDI. This raises two issues:

- Do the dots imply additional hierarchy within DNS? Replacement of the dot (.) by an underscore (_) can help to overcome this problem, either as a system to register an

agency identifier or only as a translation to determine the correct subdomain name: ICPSR_US_DDI.

- How unique are these identifiers? To be complete, these identifiers should be integrated within a standardized domain. This can be done by prefixing them with, for example, URN:DDI:AGENCY:US:ICPSR or INFO:DDI:AGENCY:US:ICPSR. The latter can avoid confusion when these identifiers are not to be resolved via the DDI URN resolution mechanism.

Compatibility with other URN resolution mechanisms

- The proposed resolution mechanism is different from any other resolution mechanisms that are used within the academic / archival community. The main difference is that the resolution requires interpretation of specific elements of the URN and application and that it needs an algorithm to construct the proper URL. Currently, the proposed solution assumes that this will be taken care of by the agents. To be compatible with other mechanisms, this complexity should be taken care of by a reusable service that is available to other (non DDI-aware) applications.

- A problem can arise when DDI resolution will be integrated with existing resolver, because these might not be able or willing to handle the high load.

Service records in DNS

The Agency's name server maintains and advertises a list of machines and services capable of resolving their own URNs. See RFC 27825 for the explanation of the snippet below for example:

```
  $ORIGIN example.com.
@               SOA server.example.com. root.example.com. (
                    1995032001 3600 3600 604800 86400 )
                NS  server.example.com.
                NS  ns1.ip-provider.net.
                NS  ns2.ip-provider.net.
; foobar - use old-slow-box or new-fast-box if either is
; available, make three quarters of the logins go to
; new-fast-box.
_foobar._tcp    SRV 0 1 9 old-slow-box.example.com.
                 SRV 0 3 9 new-fast-box.example.com.
; if neither old-slow-box or new-fast-box is up, switch to
; using the sysdmin's box and the server
                 SRV 1 0 9 sysadmins-box.example.com.
                 SRV 1 0 9 server.example.com.
server           A   172.30.79.10
old-slow-box     A   172.30.79.11
sysadmins-box    A   172.30.79.12
new-fast-box     A   172.30.79.13
; NO other services are supported
```

[5] http://www.ietf.org/rfc/rfc2782.txt

```
*._tcp          SRV  0 0 0 .
*._udp          SRV  0 0 0 .
```

Hosting essential infrastructure

All resolution depends on proper and quick functioning of the name server that hosts the agency identifiers. To ensure availability, DNS allows for replication of these records over the name servers that are hosted by participants of the DDI Alliance and/or other agencies.

If the number of agencies increases, an additional delegation via country codes can be considered.

DDI request API

Use of delegation implies scalability and manageability as well as high availability. The insertions of extra parameters in the DDI query API (like auto-incremented TTL, access KEY, etc.) will prevent infinite circular references as well as grant access rights for the case that not all metadata is accessible for everyone.

Resolution by algorithm

Many persistent identifier experts claim that identifiers should not contain any information in their string. Although maybe not ideal, in this case it is a practical solution. It only implies that both the algorithm and the information should at least be persistent. The integration of versioning should assure this.

Nested DDI objects

When nesting schemes, best practice is to use the object and its parent Maintainable object, which provides a unique ID. The URN for the resource must not change, even though the scheme might be re-used in a different context.

```
urn:ddi_3_0:     DataCollection   .     Coding

=     MPC:       DC_1[3.0]        .     PE_2[1.0]     .  Code_5
```

The definition of DDI URN does not include all object types. The more general case arising from the example in line 623 of DDI 3.0 Part 2 User Guide may contain an equation as follows:

X.Y = A.(B_1...B_n).C

This case is resolved as follows. Introduce in the DDI query API a variable called “da” to stand for disambiguator. The query request will then look like this:

?X=B&da=B_1...B_n&Z=C

The name disambiguator should also now be clear.

Transforming a URN into a query string

The process describes how the DDI-application transforms a URN into a query string. The DDI provider can also do this transformation. This would simplify the API and move the 'complexity' to the more centralized DDI provider. The drawback is that this doesn't allow querying for, e.g., all variables within a variable scheme. The exact API will have to be determined with developers of DDI applications and DDI providers.

3 References

See document for embedded references.

3.1 Normative

[RFC2119] S. Bradner, Key words for use in RFCs to Indicate Requirement Levels, http://www.ietf.org/rfc/rfc2119.txt, IETF RFC 2119, March 1997.

OASIS, Best Practice, http://www.oasis-open.org/committees/uddi-spec/doc/bp/uddi-spec-tc-bp-template.doc, 2003

Appendix A. Acknowledgments

The following individuals were members of the DDI Expert Workshop held 10-14 November 2008 at Schloss Dagstuhl, Leibniz Center for Informatics, in Wadern, Germany.

Nikos Askitas, Institute for the Study of Labor (IZA)

Karl Dinkelmann, University of Michigan

Michelle Edwards, University of Guelph

Janet Eisenhauer, University of Wisconsin

Jane Fry, Carleton University

Peter Granda, Inter-university Consortium for Political and Social Research (ICPSR)

Arofan Gregory, Open Data Foundation

Rob Grim, Tilburg University

Pascal Heus, Open Data Foundation

Maarten Hoogerwerf, Data Archiving and Networked Services (DANS)

Chuck Humphrey, University of Alberta

Jeremy Iverson, Algenta Technology

Jannik Vestergaard Jensen, Danish Data Archive (DDA)

Kirstine Kolsrud, Norwegian Social Science Data Services (NSD)

Stefan Kramer, Yale University

Jenny Linnerud, Statistics Norway

Hans Jørgen Marker, Danish Data Archive (DDA)

Ken Miller, United Kingdom Data Archive (UKDA)

Meinhard Moschner, GESIS - Leibniz Institute for the Social Sciences

Ron Nakao, Stanford University

Sigbjørn Revheim, Norwegian Social Science Data Services (NSD)

Wendy Thomas, University of Minnesota

Mary Vardigan, Inter-university Consortium for Political and Social Research (ICPSR)

282 Joachim Wackerow, GESIS - Leibniz Institute for the Social Sciences

283 Wolfgang Zenk-Möltgen, GESIS - Leibniz Institute for the Social Sciences

284

Appendix B. Revision History

Rev	Date	By Whom	What
0.9	2009-03-21	Stefan Kramer	Began revision history tracking.

Appendix C. Legal Notices

330