A Uniform Resource Name (URN) Namespace for the
Data Documentation Initiative (DDI)
draft-urn-ddi-00.txt

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 [RFC5378] and BCP 79 [RFC3979].

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/lid-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on August 25, 2011.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 [RFC5378] and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.
Abstract

This document describes a Uniform Resource Name (URN) namespace for the Data Documentation Initiative (DDI) for naming persistent resources that are structured according to the DDI specification.

Table of Contents

1. Introduction...................................................3
2. Conventions used in this document..............................4
3. Specification..................................................4
  3.1. Namespace ID..............................................4
  3.2. Registration Information..................................4
  3.3. Declared Registrant of the Namespace......................4
  3.4. Declaration of Syntactic Structure........................5
     3.4.1. Description..........................................5
     3.4.2. ABNF Grammar.........................................6
     3.4.3. Regular Expression...................................7
     3.4.4. Examples.............................................9
  3.5. Relevant Ancillary Documentation..........................9
  3.6. Identifier Uniqueness Considerations.....................10
  3.7. Identifier Persistence Considerations....................10
  3.8. Process of Identifier Assignment.........................10
  3.9. Process for Identifier Resolution.........................10
  3.10. Rules for Lexical Equivalence...........................11
  3.11. Conformance with URN Syntax.............................12
  3.12. Validation Mechanism....................................12
  3.13. Scope...................................................12
4. Namespace Considerations......................................12
  4.1. URN Assignment Procedures................................12
  4.2. URN Resolution/Delegation................................12
  4.3. Type of Resources to be Identified.......................12
  4.4. Type of Services.........................................12
5. Community Considerations......................................13
  5.1. Open Assignment and Use of Identifiers...................13
  5.2. Open Operation of Resolution Servers.....................13
  5.3. Creation of Software for Service Discovery..............13
6. IANA Considerations...........................................14
7. Security Considerations.......................................14
8. References....................................................14
  8.1. Normative References.....................................14
  8.2. Informative References...................................15
9. Acknowledgments...............................................17
Appendix A. Example DNS Records..................................18
  A.1. Delegation of the URN Namespace "ddi"....................18
  A.2. Delegation of DDI Agencies................................18
  A.3. DDI Services.............................................18
Appendix B. Algorithm for DDI Service Discovery

1. Introduction

This document registers a formal namespace identifier (NID) for Uniform Resource Names [RFC2141] associated with DDI objects in accordance with the process defined in [RFC3406].

The Data Documentation Initiative (DDI) is an international effort to establish a standard for the compilation, presentation, and exchange of documentation in the social, behavioral, and economic sciences. DDI supports a rich and structured set of metadata elements that informs a potential data analyst about a given dataset and facilitates computer processing of the data. Data producers can, by adopting the DDI standard, produce better and more complete documentation i.e. as a natural step in designing and fielding computer-assisted interviewing. DDI embraces the full life cycle of the data from conception, through development of the data collection instrument, collection and cleaning of data, production of data products, distribution, preservation, and analysis or reuse of the data. DDI is designed to facilitate sharing schemes for concepts, questions, coding, and variables within organizations or throughout the research community.

The current version of DDI is 3.1 and is expressed in XML Schema [DDI31PIV]. DDI is aligned with other metadata standards: with Dublin Core Metadata Initiative [DUBLINCO], with Statistical Data and Metadata Exchange [SDMX] for exchanging aggregate data, with ISO/IEC 11179 [ISO11179] for building metadata registries such as question, variable, and concept banks, and with ISO 19115 [ISO19115] for supporting geographic information systems.

DDI URNs support reusability of DDI objects inside a single DDI instance and in a distributed network of DDI instances. The DDI URN is a globally unique identifier which is a compound of the DDI agency identifier, the identifier of the maintainable object, and possibly the identifier of the identifiable object ([DDI31PI] section 3.3.1).

The DDI specification is developed and maintained by the DDI Alliance [DDIALLIA]. The DDI Alliance is a self-sustaining membership organization whose over 30 member institutions have a voice in the development of the DDI specification.
2. Conventions used in this document

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].

In this document, these words will appear with that interpretation only when in ALL CAPS. Lower case uses of these words are not to be interpreted as carrying RFC 2119 significance.

"DDI": Data Documentation Initiative. The single term is often used as a synonym for the DDI specification.

"DDI agency": An organization which maintains DDI instances.

"DDI Alliance": Alliance for the Data Documentation Initiative (DDI).

"RFC": The basic publication series for the IETF. RFCs are published by the RFC Editor and once published are never modified ([RFC2026] section 2.1).

3. Specification

This section provides the information required to register a formal namespace according to the registration procedure defined in [RFC3406]. The URNs conform to the syntax defined in [RFC2141].

3.1. Namespace ID

The Namespace Identifier (NID) "ddi" is requested.

3.2. Registration Information

Version 1
Date: February 25, 2011

3.3. Declared Registrant of the Namespace

Name: Alliance for the Data Documentation Initiative (DDI)
Address: University of Michigan, ICPSR
        PO Box 1248
        Ann Arbor, MI 48106-1248
        USA
Contact: Joachim Wackerow
        E-mail: joachim.wackerow@gesis.org,
                secretariat@ddialliance.org
3.4. Declaration of Syntactic Structure

3.4.1. Description

Details of the DDI URN are described in DDI 3.1 [DDI31PI] section 3.3.1. The upcoming version 3.2 of the DDI specification has some changes regarding separators and version structure. The actual syntax is here declared. This syntax isn’t final not until DDI 3.2 is published.

The Namespace Specific String (NSS) of all URNs that use the "ddi" NID have the following structure:

<agency-id>:<maintainable-object-id>:<identifiable-object-id>

agency-id is the identifier of a DDI agency that maintains DDI objects (other terms are instance or resource). This identifier basically follows the rules of reversed domain names and is case-insensitive. This way, the DNS resolution of DDI agency identifiers is supported. The hierarchy of domains descends from the left to the right label in the name; each label to the right specifies a subdivision, or subdomain of the domain to the left. The left-most label of agency-id conveys the top-level domain. It SHOULD be a country code corresponding to ISO 3166-1-alpha-2 codes [ISO3166]. The next subdomain identifies the agency within that top-level domain. Further optional subdomains can follow.

maintainable-object-id is the identifier of a maintainable object within the scope of the relevant DDI agency. A maintainable object is a container for identifiable objects.

identifiable-object-id is the identifier of an identifiable object within the scope of the relevant maintainable object. identifiable-object-id is only required if a DDI URN identifies an identifiable object. Therefore it is optional.

Both maintainable-object-id and identifiable-object-id have the following structure and are case-sensitive:

<ddi-object-name>:<object-id>:<object-version>

ddi-object-name is a valid object name according to the DDI specification (maintainable, versionable, or identifiable element).

object-id is an identifier within a local scope.
object-version is a hierarchical version number and used in object identification. Example:

4.34.7 major version 4, minor version 34, sub-minor version 7

object-version-in-ref replaces object-version in the case of a reference to an object. To supplement the structure of object-version, late-binding can be specified by the indicator "l". It is possible to request the latest version of an object or the latest version within a specific level. Example:

4.L major version 4, highest sub-minor version

3.4.2. ABNF Grammar

The following syntax specification for the complete URN uses the augmented Backus-Naur Form (ABNF) as described in [RFC5234].

; Rules are case-sensitive, if not stated otherwise.
ddi-urn = urn
  major-separator ddi
  major-separator agency-id
  major-separator maintainable-object-id
  [ major-separator identifiable-object-id ]

urn = "urn"
  ; urn is case-insensitive.
  ; See [RFC2141] section 2.1.

ddi = "ddi"
  ; ddi is case-insensitive.
  ; See [RFC2141] section 2.1.
  ; ddi is the URN namespace identifier.

agency-id = 1*255(
  top-level-domain
  minor-separator agency-name
  *(minor-separator sub-agency-name)
)
  ; agency-id is case-insensitive.

maintainable-object-id= object-complete-id
identifiable-object-id= object-complete-id
top-level-domain = dns-label
agency-name = dns-label
sub-agency-name = dns-label
dns-label = 1*63(
  (ALPHA / DIGIT)
  [ *(ALPHA / DIGIT / hyphen)
    (ALPHA / DIGIT) ]
)
object-complete-id = ddi-object-name
    major-separator object-id
    major-separator (object-version /
        object-version-in-ref)
    ; object-version-in-ref is only
    ; allowed in an object reference.

ddi-object-name       = 1*ALPHA
object-id             = 1*(ALPHA / DIGIT / underscore / hyphen)
object-version        = 1*DIGIT *(minor-separator 1*DIGIT)
object-version-in-ref = ( 1*DIGIT
    *(minor-separator 1*DIGIT)
    *1(minor-separator late-bound)
) / late-bound

major-separator       = %x3A       ; colon (":")
minor-separator       = %x2E       ; dot (".")
hyphen                = %x2D       ; hyphen ("-")
underscore            = %x5F       ; underscore ("_")
late-bound            = %x4C       ; uppercase l ("L")
ALPHA                 = %x41-5A /  ; uppercase letters
    %x61-7A  ; lowercase letters
DIGIT                 = %x30-39    ; digits

; ALPHA and DIGIT are actually defined in the ABNF specification.
; They are here declared for convenience purposes.

3.4.3. Regular Expression

The following regular expression syntax uses components (component
names indicated by angle brackets, i.e. <component>) and is written
in free-spacing mode for easier reading (the XML Schema flavor does
not support that). These regular expressions implicitly anchor at the
head and tail (as with the XML Schema flavor). For other regular
expression language flavors, it is necessary to anchor the pattern by
"^" at the head and by "$" at the tail. The XML Schema flavor
supports only non-capturing groups. If backreferences are not
required in the replacement string, the regular expressions in other
flavors can be optimized by using non-capturing groups. For example:

    (?::L|\d+(?::\.\d+)*(?::\.L)?) instead of (L|\d+(\.\d+)*(\.L)?)

The regular expressions are valid for popular regular expression
languages and are tested for the flavors XML Schema, Java, and C#.
ddi-urn:=
  [Uu][Rr][Nn]
  : [Dd][Dd][Ii]
  : <agency-id>
  : <maintainable-object-id>
  ( : <identifiable-object-id> )?

agency-id :=
  ( <top-level-domain>
    \. <agency-name>
    (\. <sub-agency-name>)*
  ){1,255}

top-level-domain, agency-name, sub-agency-name :=
  <dns-label>

dns-label :=
  ( [0-9A-Za-z] ([0-9A-Za-z\-]* [0-9A-Za-z])? [0-9A-Za-z])? ){1,63}

maintainable-object-id, identifiable-object-id :=
  <object-complete-id>

object-complete-id :=
  <ddi-object-name>
  : <object-id>
  : ( <object-version> | <object-version-in-ref> )

ddi-object-name :=
  [A-Za-z]+

object-id :=
  [0-9A-Za-z\-__]+

object-version-in-ref :=
  L|[0-9]+(\.[0-9]+)\* (\.L)?

object-version :=
  [0-9]+(\.[0-9]+)\*
3.4.4. Examples

URN of a Maintainable Object

urn:ddi:us.mpc:QuestionScheme:lp239:2.1

The DDI question scheme identified by "lp239" with the version "2.1" of the DDI agency "mpc" located in the United States.

URN of an Identifiable Object

urn:ddi:gb.ukda:VariableScheme:vs1786:4.2.3:Variable:v07:1.7

The DDI variable identified by "v07" with the version "1.7" and maintained in the variable scheme identified by "vs1786" with the version "4.2.3" of the DDI agency "ukda" located in the United Kingdom.

Late-bound Reference to an Identifiable Object

urn:ddi:de.gesis:CategoryScheme:047ia7:1:Category:a975d7:2.L

The DDI category identified by "a975d7" with the major version 2, the latest minor version and maintained in the category scheme identified by "047ia7" with the version "1" of the DDI agency "gesis" located in Germany.

3.5. Relevant Ancillary Documentation

An introductory article can be found at [DDIINTRO].

Information on the DDI specification can be found in the Overview of the Technical Specification of DDI 3.1 [DDI31PI].

Information on domain names can be found in the relevant RFCs.

- For an overview, see [RFC1034].
- Regarding case insensitivity, see [RFC1035] section 2.3.3.
- Regarding syntax, see [RFC952] section "Grammatical Host Table Specification" B. and [RFC1123] section 2.1.
- Regarding size limits, see [RFC1123] section 2.1 and [RFC1035] section 2.3.4.
3.6. Identifier Uniqueness Considerations

Assignment of identifiers for DDI agencies in the requested namespace will be managed by the DDI Alliance, which will ensure that the assigned DDI agency identifiers are consistent with the directives for unique identification of DDI agencies.

Assignment of URNs for objects of a DDI agency in the requested namespace will be managed by the respective DDI agency, which will ensure that the assigned URNs are unique for scope of the agency.

3.7. Identifier Persistence Considerations

Persistence of identifiers is dependent upon suitable delegation of resolution at the level of the DDI agencies, and persistence of DDI agency assignment. The persistence of the referenced resource is also the responsibility of the DDI agency.

3.8. Process of Identifier Assignment

Assignment of identifiers for DDI agencies in the requested namespace will be managed by the DDI Alliance.

Assignment of URNs for objects of a DDI agency and sub-agencies of a DDI agency in the requested namespace will be managed by the respective DDI agency.

3.9. Process for Identifier Resolution

The DDI Alliance will promote a service discovery system for identifying available services connected to DDI agencies using the Domain Name System (DNS). A DNS request for a DDI agency within the domain ddi.urn.arpa is delegated by the DNS servers of the DDI Alliance to the DNS servers of the relevant DDI agency. The response is a list of available DDI services for the agency identifier under which the agency has assigned URNs. The approach is based on the Dynamic Delegation Discovery System (DDDS) [RFC3401] and especially the straightforward URI-enabled NAPTR (U-NAPTR) [RFC4848].

The DDI Alliance is responsible for operating or delegating resolution requests to the resolution servers of the relevant DDI agencies. DDI agencies are responsible for operating or delegating resolution servers for the agency identifier under which they have assigned URNs.
Sample Sequence Diagram

<table>
<thead>
<tr>
<th>Client</th>
<th>NS for urn.arpa</th>
<th>NS for ddialliance.org</th>
<th>NS for pop.umn.edu</th>
<th>DDI services for us.mpc</th>
</tr>
</thead>
</table>
1 | -------> | | | |
2 | <------- | | | |
3 | | <----------------------> | | |
4 | | | <----------------------| | |
5 | | | | ------------------------------------> |
6 | | | | <-------------------------------------| |
7 | | | | | -----------------------------------------------------------------|
8 | | | | | <----------------------------------------------------------------|

1. The name server (NS) of IANA for the domain "urn.arpa." is reached with the request "mpc.us.ddi.urn.arpa." for the DDI agency "us.mpc".

2./3. The request is delegated to the name server for "ddialliance.org".

4./5. The request is delegated to the name server for "pop.umn.edu" (domain of the DDI agency "us.mpc").

6. The server responds with a list of NAPTR records [RFC3403] pointing to available DDI services for the DDI agency "us.mpc".

7. The client selects an appropriate DDI service and sends a request for a DDI URN to this service.

8. The DDI service responds for example with a DDI object identified by the requested DDI URN.

See Appendix A for examples of name server records.

3.10. Rules for Lexical Equivalence

The DDI agency identifier basically follows the rules of domain names. Domain names are case-insensitive. Thus, the portion of the URN

    urn:ddi:<agency-id>:

is case-insensitive for matches. The remainder of the identifier MUST be considered case-sensitive.
3.11. Conformance with URN Syntax

The Namespace Specific String uses a subset of the allowed characters in [RFC2141]: letters, digits, hyphen, dot, colon, and underscore. Escaping is not used.

3.12. Validation Mechanism

The DDI Alliance will promote development of software for validation purposes.

3.13. Scope

The scope is global.

4. Namespace Considerations

There is no available namespace that will allow one to uniquely identify and access DDI resources.

4.1. URN Assignment Procedures

See also above section on Process of Identifier Assignment.

4.2. URN Resolution/Delegation

See also above section on Process for Identifier Resolution.

It is RECOMMENDED to use sub-agencies for flexible administration. For example, delegation of URNs of a sub-agency to different servers would be easily possible.

If necessary, the delegation process can be extended by an additional step on the country level.

4.3. Type of Resources to be Identified

The DDI specification has more than 800 objects, from which 120 objects can be identified by a DDI URN. The identifiable objects are contained in over 30 maintainable container objects.

4.4. Type of Services

Examples of potential services are listed below. The services and appropriate service tags need to be defined in future. The mentioned service tags are from [RFC2169].
5. Community Considerations

5.1. Open Assignment and Use of Identifiers

DDI agency identifiers can be registered at the DDI Alliance. The DDI Alliance will maintain a registry of the assigned values for the DDI agency identifier used in the NSS. Information may be obtained from the following address: secretariat@ddialliance.org.

DDI agencies assign URNs and potential sub-agencies within the scope of the assigned DDI agency identifiers.

See also above section on Identifier Uniqueness Considerations.

5.2. Open Operation of Resolution Servers

The DDI Alliance operates publicly accessible name servers for the delegation of DNS requests within the domain ddi.urn.arpa to DNS servers of DDI agencies.

5.3. Creation of Software for Service Discovery

The DDI Alliance will promote software for the resolution of DDI agency identifiers and service discovery. See also Appendix B for the algorithm.
6. IANA Considerations

This document defines a URN NID registration of "ddi". IANA is asked to register "ddi" in the URN Namespaces registry.

The URN.ARPA registration for the URN namespace "ddi" is planned. Requests for the domain ddi.urn.arpa will be delegated to the name servers of the DDI Alliance.

7. Security Considerations

This document introduces no additional security considerations beyond those associated with the use and resolution of URNs in general.

The security of the DNS-based resolution of DDI agency identifiers is only as good as the security of DNS queries in general. A full discussion of the security threats pertaining to DNS and possible solutions can be found in [RFC3833]. Further information on security considerations regarding U-NAPTR can be found in [RFC4848] section 6.

8. References

8.1. Normative References


8.2. Informative References

[ABNF2REG] abnf2regex from Martin Thomson, 
https://github.com/martinthomson/abnf2regex


[ABNFPFEN] ABNF Parser from Bill Fenner, 
http://tools.ietf.org/tools/bap/abnf.cgi

[ABNFPTHO] ABNF Parser Generator from Lowell D. Thomas, 
http://www.coasttocoastresearch.com/interactiveapq


[DUBLINCO] Dublin Core Metadata Initiative, 
http://www.dublincore.org/

[ISO3166] ISO 3166-1-alpha-2 code elements, 
http://www.iso.org/iso/english_country_names_and_code_elements.


9. Acknowledgments

Many thanks to Arofan Gregory, Dan Smith, and Wendy Thomas from the DDI Technical Implementation Committee, and Peter Koch from DENIC (German Network Information Center) for discussion and input that led to this document.

The following software tools have been helpful in evaluating the ABNF grammar and the regular expressions: two ABNF parsers [ABNFPFEN] [ABNFPTHO], a tool that creates regular expressions from an ABNF grammar [ABNF2REG], and a tool that generates random strings that match an ABNF grammar [ABNFGEN].

This document was prepared using the Word template 2-Word-v2.0.template.dot [RFC5385].
Appendix A. Example DNS Records

The examples use NAPTR [RFC3403] and SRV [RFC2782] records. The values for the services and flags fields of the NAPTR records will be determined by the DDI application ([RFC3403] section 9.). The DDI Alliance will establish a mechanism for the description and registration of DDI services.

A.1. Delegation of the URN Namespace "ddi"

Records at a.iana-servers.net and other authoritative name servers for the domain urn.arpa.

The empty flag indicates that the lookup is not terminal and the next probe to DNS is for more NAPTR records where the new domain is "dns.ddialliance.org".

; Delegation to name servers of ddialliance.org
; order pref flag service regexp replacement
ddi.urn.arpa.
IN NAPTR 100 10 "" "" "" dns.ddialliance.org.

A.2. Delegation of DDI Agencies

Records at dns.ddialliance.org for ddi.urn.arpa.

The empty flag indicates that the lookup is not terminal and the next probe to DNS is for more NAPTR records where the new domain is the DNS server of the relevant DDI agency.

; Delegation to name servers of subdomains in ddi.urn.arpa, i.e. DDI agencies.
; order pref flag service regexp replacement
gesis.de.ddi.urn.arpa.
IN NAPTR 100 10 "" "" "" dns.gesis.org.
mpc.us.ddi.urn.arpa.
IN NAPTR 100 10 "" "" "" dns.pop.umn.edu.
ukda.gb.ddi.urn.arpa.
IN NAPTR 100 10 "" "" "" dns.data-archive.ac.uk.

A.3. DDI Services

Records at dns.gesis.org for ddi.urn.arpa.

The "u" flag states that the rule is terminal and that the output is a URI which contains the information needed to contact that DDI service. The "s" flag states that the rule is terminal and that the
output of the rewrite will be a domain-name for which an SRV record should be queried. See also [RFC4848] section 4.4.

The service N2R returns one instance of the resource identified by the given URN. That service is a repository of DDI instances available at http://ddirepos.gesis.org/N2R/; possibly a REST-based service. The service N2C returns a description of the resource identified by the given URN. That service is a registry of DDI instances available at registry-udp.gesis.org port 10060.

U-NAPTR permits regular expressions of a form that does a complete replacement of the matched string with a URI, expressed as a constant string. With this limited form of regular expression ([RFC4848] section 2.2.), applications using NAPTR need not implement full regular expression parsers.

gesis.de.ddi.urn.arpa.
;         order pref flag
  IN NAPTR 100 10 "u" "N2R+http" (; service
                      "!.*!http://ddirepos.gesis.org/N2R/!"; regex
                      ; replacement
                     )
  IN NAPTR 100 10 "s" "N2C+udp" (; service
                      "" ; regex
                      _registry._udp.gesis.org. ; replacement
                     )
  ; all subdomains in gesis.de.ddi.urn.arpa.
*._service._protocol.name
; TTL class SRV priority weight port target
  registry. udp.gesis.org
     - 14400 IN  SRV 0 0 10060 registry-udp.gesis.org.
Appendix B. Algorithm for DDI Service Discovery

The description is based on the Dynamic Delegation Discovery System (DDDS) algorithm [RFC3402].

Application Unique String

The Application Unique String is a DDI URN.

First Well Known Rule

1. Extracting the characters between the second and third colon (the agency identifier).
2. Normalizing case of that string.
3. Reversing the order of the substrings separated by dots.
4. Appending the string ".ddi.urn.arpa" to the end to get a domain name.

Valid Databases

The DNS is specified as a DDDS Database for this application which uses the NAPTR DNS resource records to contain the rewrite rules for service discovery.

The DNS is queried for NAPTR records for the domain name which is the output of the First Well Known Rule.

Expected Output

The expected output is the information necessary to connect to authoritative server(s) (host, port, protocol, or URL) for an application service within a given DDI agency. The result is a list of terminal NAPTR records pointing to services available for the relevant DDI agency.

The application selects the appropriate service and contacts the service for the given URN.

The process can be optimized by an application cache for the NAPTR records of already requested DDI agencies.
Author’s Address

Joachim Wackerow
(Vice chair, Technical Implementation Committee, DDI Alliance)
GESIS - Leibniz Institute for the Social Sciences
B2 1
68159 Mannheim
Germany

Email: joachim.wackerow@gesis.org