

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: June 10, 2013

E. Wilde  
C. Davis  
EMC Corporation  
Y. Liu  
UC Berkeley  
December 7, 2012

URI Template Descriptions  
draft-wilde-template-desc-00

Abstract

Interactions with many resources on the Web are driven by and/or can be described by URI Templates. [RFC 6570](#) says that "a URI Template is a compact sequence of characters for describing a range of Uniform Resource Identifiers through variable expansion." This specification defines Template Descriptions, a schema and a media type that can be used to document and describe a URI Template by supporting descriptive markup that allows variables to be associated with documentation (human-oriented) and/or descriptions (machine-oriented). Template Descriptions can be used by media types (by inclusion or by reference) that seek to make URI Templates self-describing, without having to create their own representation.

Note to Readers

Please discuss this draft on the rest-discuss mailing list [[9](#)].

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

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."

This Internet-Draft will expire on June 10, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) 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. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1.](#) Introduction . . . . . [3](#)
- [1.1.](#) URI Template . . . . . [3](#)
- [1.2.](#) Describing Variables . . . . . [4](#)
- [2.](#) Terminology . . . . . [4](#)
- [3.](#) Template Descriptions . . . . . [5](#)
- [3.1.](#) General Concepts . . . . . [5](#)
- [3.2.](#) Template Description Structure . . . . . [5](#)
- [3.3.](#) Variable Description Structure . . . . . [5](#)
- [4.](#) IANA Considerations . . . . . [5](#)
- [4.1.](#) Media Type . . . . . [5](#)
- [4.2.](#) Link Relation . . . . . [6](#)
- [5.](#) Security Considerations . . . . . [7](#)
- [6.](#) Examples . . . . . [7](#)
- [6.1.](#) OpenSearch . . . . . [7](#)
- [6.2.](#) Atom Feeds . . . . . [7](#)
- [7.](#) Open Issues . . . . . [7](#)
- [8.](#) References . . . . . [8](#)
- [8.1.](#) Normative References . . . . . [8](#)
- [8.2.](#) Informative References . . . . . [8](#)
- Authors' Addresses . . . . . [8](#)

## 1. Introduction

Simple interactions with resources on the Web often are driven by simply following links, but in many other cases, interactions with resources require clients to provide information in addition to just using a fixed URI in a request. In these cases, information can be provided in any way supported by the interaction protocol, and in case of HTTP, this often means that information is either embedded in the URI, and/or in the body of the request. For the first case, the "URI Template" standard [Section 1.1](#) provides a standard that allows servers and clients to exchange information about the URIs that a service accepts.

### 1.1. URI Template

The "URI Template" standard [RFC 6570 \[1\]](#) specifies "a compact sequence of characters for describing a range of Uniform Resource Identifiers through variable expansion." It allows servers to publish their expectation how a URI should be created by substituting variables with values. Consider the following URI Template:  
`http://www.example.com/feed{?pagesize,page}`

This URI Template allows clients to expand two variables to end up with a concrete URI such as the following:  
`http://www.example.com/feed?pagesize=10&page=42`

URI Templates cover the aspect of starting with a template with variables in it, assigning values to these variables, and then expanding the template into a URI that can be used for sending a request. URI Templates make no assumptions or statements about the value range of the variables, except for those aspects which are required to cover the process of expanding the template. In particular, for the example given above, there is no indication that the values are supposed to be positive integers (the simple data type), nor is there any indication that the service may apply certain limits such as a maximum page size (which may change depending on which paged resource is being accessed). As a side note, even if this information was known, URI template expansion could still result in URIs that would not yield successful requests, such as when asking for a page that is beyond the number of pages that a feed has (in a given page size).

The goal of Templates Descriptions as defined here is to allow servers to expose a description resource that provides support both at development time (when a developer looks at a media type that uses URI Templates) and at runtime (when a client wants to use a URI template as part of its application flow).

## [1.2.](#) Describing Variables

Variables can be described in a variety of ways when using Template Descriptions. For each variable in the URI Template, it is possible to use the following description methods:

**Domain Concept:** It is possible to associate a variable with a domain concept, so that media types and applications can make an association between the concepts they are defining/exposing, and where they are represented in URI Templates. Concepts can be referred to by URI, or by using a QName [\[2\]](#).

**Datatype:** Variables can be described in terms of using certain datatypes, and the datatype vocabulary is that of XML Schema Part 2 [\[3\]](#), plus all of the applicable facets of those datatypes. This allows Template Descriptions to constrain the set of allowed values.

**Documentation:** Documentation constructs can be associated with variables, which allows Template Descriptions to attach human-readable information to individual variables. The documentation constructs use the documentation design of XML Schema Part 1 [\[4\]](#).

**Application Information:** Application information constructs can be associated with variables, which allows Template Descriptions to attach machine-readable information to individual variables. The application information constructs use the application information design of XML Schema Part 1 [\[4\]](#).

For the purpose of this specification, the term "description" should be interpreted loosely. Some aspects of descriptions can be formal, such as the datatypes of variables. Thus, such a description can be used to drive general-purpose logic that knows nothing but this specification. However, for most other description aspects (domain concepts, documentation, and application information), this specification does not prescribe a description framework; it simply provides a structure how to deliver these descriptions.

## [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 [RFC 2119](#) [\[5\]](#).

### [3.](#) Template Descriptions

Template Descriptions are based on a URI Template, and add descriptive elements that allow publishers of URI Templates to describe the URI Template as a whole, and to add individual descriptions of all variables in the template.

#### [3.1.](#) General Concepts

As mentioned in [Section 1.1](#), most of the descriptions in this spec do not prescribe a specific description framework. While variables ([Section 3.3](#)) can be described with a built-in vocabulary of datatypes, most other descriptions are either for human consumption, or do rely on some external description framework. To attach these descriptions to both the template as a whole, and individual variables, this specification reuses the "appinfo" and "documentation" elements from XML Schema Part 1 [\[4\]](#). These elements carry a "source" attribute which is used (quoting from [\[4\]](#)) "to supplement the local information."

#### [3.2.](#) Template Description Structure

A template is described by including the URI Template itself, and optionally adding documentation and/or appinfo elements to add human- or machine-readable descriptions.

#### [3.3.](#) Variable Description Structure

A variable is described by specifying the variable name. Variables can refer to a "concept" associated with a variable, which can be identified by URI, or by a QName (at most one of these SHALL be present). This specification makes no provision how such a concept is defined or described, but it allows consumers of a Template Description to match their understanding of certain concepts to those identifiers, which then establishes a binding between the concept, and the variable it has been bound to.

Variable descriptions can optionally add documentation and/or appinfo elements to add human- or machine-readable descriptions.

## [4.](#) IANA Considerations

### [4.1.](#) Media Type

The Internet media type for a Template Description document is application/td+xml.

Type name: application

Subtype name: td+xml

Required parameters: none

Optional parameters: none

Encoding considerations: binary

Security considerations: See Security Considerations in [Section 5](#).

Interoperability considerations: N/A

Published specification: [[ This document ]]

Applications that use this media type: Applications that publish descriptions of URI Templates.

Additional information:

Magic number(s): N/A

File extension(s): .tdx

Macintosh file type code(s): TEXT

Person & email address to contact for further information: Erik Wilde <erik.wilde@emc.com>

Intended usage: COMMON

Restrictions on usage: none

Author: Erik Wilde <erik.wilde@emc.com>

Change controller: IETF

#### [4.2.](#) Link Relation

The link relation type below will be registered by IANA per [Section 6.2.1 of RFC 5988](#) [6]:

Relation Name: query-template

Description: Linking to a resource that can be used as a template description for creating a request URI for the context resource.

Reference: [[ This document ]]

Notes: Template Descriptions can be used in all scenarios where clients want to create requests that represent a query into the context resource. The media type of the context resource and the media type of the template description resource are not constrained by this specification.

## [5.](#) Security Considerations

...

## [6.](#) Examples

...

### [6.1.](#) OpenSearch

...

### [6.2.](#) Atom Feeds

...

"Feed Paging and Archiving" specified in [RFC 5005](#) [8]

Maybe suggest to use profiles?

## [7.](#) Open Issues

If and how to use profiles (example in [Section 6](#)); if profile use is recommended, define a suggested profile URI for other specs to use?

Should we support other ways how URIs could be generated, most importantly the application/x-www-form-urlencoded method of HTML forms?

How to handle variables in Level 4 templates that are supposed to have composite values?

Should there be some way how default values can be exposed/ documented?

If a template is refined in an incremental process, does it make sense to be able to add a "back" link and/or "home" link, so that clients can find the "most general" version easily?

## [8.](#) References

### [8.1.](#) Normative References

- [1] Gregorio, J., Fielding, R., Hadley, M., Nottingham, M., and D. Orchard, "URI Template", [RFC 6570](#), March 2012.
- [2] Hollander, D., Layman, A., Thompson, H., Tobin, R., and T. Bray, "Namespaces in XML 1.0 (Third Edition)", World Wide Web Consortium Recommendation REC-xml-names-20091208, December 2009, <<http://www.w3.org/TR/2009/REC-xml-names-20091208>>.
- [3] Malhotra, A. and P. Biron, "XML Schema Part 2: Datatypes Second Edition", World Wide Web Consortium Recommendation REC-xmlschema-2-20041028, October 2004, <<http://www.w3.org/TR/2004/REC-xmlschema-2-20041028>>.
- [4] Thompson, H., Beech, D., Mendelsohn, N., and M. Maloney, "XML Schema Part 1: Structures Second Edition", World Wide Web Consortium Recommendation REC-xmlschema-1-20041028, October 2004, <<http://www.w3.org/TR/2004/REC-xmlschema-1-20041028>>.
- [5] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](#), March 1997.
- [6] Nottingham, M., "Web Linking", [RFC 5988](#), October 2010.
- [7] Nottingham, M., Ed. and R. Sayre, Ed., "The Atom Syndication Format", [RFC 4287](#), December 2005.

### [8.2.](#) Informative References

- [8] Nottingham, M., "Feed Paging and Archiving", [RFC 5005](#), September 2007.

#### URIs

- [9] <<http://tech.groups.yahoo.com/group/rest-discuss/>>



Authors' Addresses

Erik Wilde  
EMC Corporation  
6801 Koll Center Parkway  
Pleasanton, CA 94566  
U.S.A.

Phone: +1-925-6006244  
Email: [erik.wilde@emc.com](mailto:erik.wilde@emc.com)  
URI: <http://dret.net/netdret/>

Cornelia Davis  
EMC Corporation

Email: [cornelia.davis@emc.com](mailto:cornelia.davis@emc.com)

Yiming Liu  
UC Berkeley  
South Hall  
Berkeley, CA 94720  
U.S.A.

Email: [yliu@ischool.berkeley.edu](mailto:yliu@ischool.berkeley.edu)