Variant Extension Mapping for the Extensible Provisioning Protocol (EPP)

Author

I. Treadwell

J. Khallouf

Status of This Document

Draft - AusRegistry Pty Ltd

Copyright Notice

Copyright (C) 2010 AusRegistry Pty Ltd

Abstract

This document describes an Extensible Provisioning Protocol (EPP) extension mapping for the provisioning and management of Internationalised Domain Names in Applications (IDNA) domain names stored in a shared central repository. Specified in XML, this mapping extends the EPP domain name mapping to provide additional features required for the specifying IDNA domain variants, as described in the AusRegistry document, ' Variants Protocol Extension for the Extensible Provisioning Protocol '.

Table of Contents

[1. Introduction 3](#_Toc252380739)

[1.1 Conventions used in this document 3](#_Toc252380740)

[2 Object Attributes 3](#_Toc252380741)

[2.1 IDNA Domain Names 3](#_Toc252380742)

[3 EPP Command Mapping 3](#_Toc252380743)

[3.1 EPP Query Commands 3](#_Toc252380744)

[3.1.1 EPP <check> Command 4](#_Toc252380745)

[3.1.2 EPP <info> Command 4](#_Toc252380746)

[3.1.3 EPP <variantInfo> Command 5](#_Toc252380747)

[3.2 EPP Transform Commands 6](#_Toc252380748)

[3.2.1 EPP <create> Command 7](#_Toc252380749)

[3.2.2 EPP <delete> Command 7](#_Toc252380750)

[3.2.3 EPP <renew> Command 7](#_Toc252380751)

[3.2.4 EPP <transfer> Command 8](#_Toc252380752)

[3.2.5 EPP <update> Command 8](#_Toc252380753)

[4. Formal Syntax 9](#_Toc252380754)

[5. Internationalization Considerations 11](#_Toc252380755)

[6. Security Considerations 11](#_Toc252380756)

[7. References 11](#_Toc252380757)

[7.1 Normative References 11](#_Toc252380758)

[7.2 Informative References 12](#_Toc252380759)

# Introduction

This document describes a Variants domain mapping for version 1.0 of the Extensible Provisioning Protocol (EPP) and the AusRegistry Variant protocol extension [3]. This mapping, an extension of the domain name mapping described in [1], is specified using the Extensible Markup Language (XML) 1.0, as described in [5], and XML Schema notation, as described in [6] and [7].

The EPP core protocol specification [2] and the AusRegistry Variant protocol extension specifications [3] provide a complete description of EPP command and response structures. A thorough understanding of the base protocol and specification relevant extensions is necessary to understand the mapping described in this document.

## 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 RFC-2119 [4].

In examples, "C:" and "S:" indicate lines sent by the client and server respectively. Indentation and white space in examples is provided only to illustrate element relationships and is not a mandatory feature of this protocol.

XML is case sensitive. Unless stated otherwise, XML specifications and examples provided in this document MUST be interpreted in the character case presented to develop a conforming implementation.

# Object Attributes

This extension adds elements to the EPP domain name mapping [1]. Only new element descriptions are described here.

## IDNA Domain Names

Internationalizing Domain Names in Applications (IDNA) is a mechanism for handling Internationalised Domain Names (IDN) in a standard fashion. To provide interoperability with legacy systems, IDNA defines the ASCII Compatible Encoding (ACE) for non-ASCII labels. This extension makes the distinction between the two representations of domain names; the ACE-encoded form and native presentation form, denoted by the terms 'DNS form' and 'user form' respectively.

# EPP Command Mapping

A detailed description of the EPP syntax and semantics can be found in the EPP core protocol specification [2]. The command mappings described here are specifically for use in implementing IDNA domain provisioning processes via EPP.

## EPP Query Commands

EPP provides three commands to retrieve object information: <check> to determine if an object is known to the server, <info> to retrieve detailed information associated with an object, and <transfer> to retrieve object transfer status information. In addition, the AusRegistry Variant protocol extension [3] specifies an additional command, <variantInfo>.

### EPP <check> Command

This extension does not add any elements to the EPP <check> command or <check> response described in the EPP domain mapping [1].

### EPP <info> Command

This extension adds elements to the EPP <info> command as well as the EPP <info> response, both described in the EPP domain mapping [1]. The response to this command MAY vary depending on the identity of the querying client, use of authorization information, and server policy towards unauthorized clients.

In addition to the EPP command elements described in the EPP domain mapping [1], the <info> command MUST contain an EPP <extension> element, which MUST contain an <info> element that identifies the variant namespace. The <info> element contains the following:

* An optional "variants" attribute that indicates whether the response to the info command should list all variants of the domain. Allowable values for this attribute are "all", or "none", with the default value being "all". If "none" is specified, then no variants are listed in the response. If "all" is specified, or the attribute is not present, then all variants are listed in the response.

C: <?xml version="1.0" encoding="UTF-8"?>

C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">

C: <command>

C: <info>

C: <info xmlns="urn:ietf:params:xml:ns:domain-1.0">

C: <name>xn--kxaefcf8b.idn</name>

C: </info>

C: </info>

C: <extension>

C: <info xmlns="urn:X-ar:params:xml:ns:variant-1.0"

C: variants="all"/>

C: </extension>  
C: <clTRID>ABC-12345</clTRID>

C: </command>

C: </epp>

When an <info> command has been successfully processed, the EPP <resData> element in the <info> response MUST contain child elements as described in the EPP domain mapping [1]. In addition, the <info> response MUST contain an EPP <extension> element, which MUST contain an <infData> element that identifies the variant namespace. The <infData> element contains the following child elements:

* One or more <variant> elements containing the fully qualified name of any variants of a domain name, in DNS form. Each <variant> element MUST have a "userForm" attribute indicating the native presentation form of the variant name.

Example <info> response:

S: <?xml version="1.0" encoding="UTF-8" standalone="no"?>

S: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">

S: <response>

S: <result code="1000">

S: <msg lang="en">Command completed successfully</msg>

S: </result>

S: <resData>

S: <infData xmlns="urn:ietf:params:xml:ns:domain-1.0">

S: <name>xn--kxaefcf8b.idn</name>

S: <roid>DA25CB8C51B42CA2A84B6B4876AFC512B-AR</roid>

S: <status s="ok"/>

S: <registrant>CON-1</registrant>

S: <clID>EPP</clID>

S: <crID>EPP</crID>

S: <crDate>2008-09-22T02:40:12.0Z</crDate>

S: <exDate>2010-09-22T02:40:12.0Z</exDate>

S: <authInfo>

S: <pw>2fooBAR</pw>

S: </authInfo>

S: </infData>

S: </resData>

S: <extension>

S: <infData xmlns="urn:X-ar:params:xml:ns:variant-1.0">

S: <variant userForm="δεiγua.idn">xn--ua-48bqdg.idn</variant>

S: </infData>

S: </extension>

S: <trID>

S: <clTRID>ABC-12345</clTRID>

S: <svTRID>54321-XYZ</svTRID>

S: </trID>

S: </response>

S: </epp>

An EPP error response MUST be returned if an extended <info> command cannot be processed for any reason.

### EPP <variantInfo> Command

This extension provides a new object mapping for the EPP <variantInfo> command and EPP <variantInfo> response, described in the AusRegistry Variant protocol extension specification [3]. This command is used to determine which domains would be provisioned in a Registry if a specified domain was created. The behaviour and syntax of this command is unaffected by whether the specified domain has already been created or not.

The <variantInfo> command MUST contain a <variantInfo> element that identifies the variant namespace. The <variantInfo> element contains the following elements:

* One <name> element that contains the fully-qualified name of the domain to be queried. An OPTIONAL "language" attribute MAY be present to indicate the language of the name.

Example <variantInfo> command:

C: <?xml version="1.0" encoding="UTF-8" standalone="no"?>

C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">

C: <extension>

C: <command xmlns="urn:X-ar:params:xml:ns:viext-1.0">

C: <variantInfo>

C: <variantInfo xmlns="urn:X-ar:params:xml:ns:variant-1.0">

C: <name language="gr">δείγμα.idn</name>

C: </variantInfo>

C: </variantInfo>

C: <clTRID>ABC-12345</clTRID>

C: </command>

C: </extension>

C: </epp>

When a <variantInfo> command has been completed successfully, the EPP <resData> element in the <variantInfo> response MUST contain a child <varInfData> element that identifies the variant namespace. The <varInfData> element contains the following elements:

* One or more <variant> elements containing the fully qualified name of any variants of a domain name, in DNS form. Each <variant> element MUST have a "userForm" attribute indicating the native presentation form of the variant name.

Example <variantInfo> response:

S: <?xml version="1.0" encoding="UTF-8" standalone="no"?>

S: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">

S: <response>

S: <result code="1000">

S: <msg lang="en">Command completed successfully</msg>

S: </result>

S: <resData>

S: <varInfData xmlns="urn:X-ar:params:xml:ns:variant-1.0">

S: <name userForm="δείγμα.idn">xn--kxaefcf8b.idn</name>

S: <name userForm="δείγua.idn">xn--ua-48bqdg.idn</name>

S: </varInfData>

S: </resData>

S: <trID>

S: <clTRID>ABC-12345</clTRID>

S: <svTRID>54322-XYZ</svTRID>

S: </trID>

S: </response>

S: </epp>

An EPP error response MUST be returned if a <variantInfo> command cannot be processed for any reason.

## EPP Transform Commands

EPP provides five commands to transform objects: <create> to create an instance of an object, <delete> to delete an instance of an object, <renew> to extend the validity period of an object, <transfer> to manage object sponsorship changes, and <update> to change information associated with an object.

### EPP <create> Command

This extension does not add any elements to the EPP <create> command, however it does add extensions to the EPP <create> response. Both are described in the EPP domain mapping [1].

When a <create> command has been successfully processed, the EPP <resData> element in the <create> response MUST contain child elements as described in the EPP domain mapping [1]. In addition, the <create> response MUST contain an EPP <extension> element, which MUST contain an <creData> element identifying the variant namespace. The <creData> contains the following child elements:

* One or more <variant> elements containing the fully qualified name of any variants of a domain name, in DNS form. Each <variant> element MUST have a "userForm" attribute indicating the native presentation form of the variant name.

Example <create> response:

S: <?xml version="1.0" encoding="UTF-8" standalone="no"?>

S: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">

S: <response>

S: <result code="1000">

S: <msg lang="en">Command completed successfully</msg>

S: </result>

S: <resData>

S: <creData xmlns="urn:ietf:params:xml:ns:domain-1.0">

S: <name>xn--kxaefcf8b.idn</name>

S: <crDate>2008-10-06T23:49:30.0Z</crDate>

S: <exDate>2010-10-06T23:49:30.0Z</exDate>

S: </creData>

S: </resData>

S: <extension>

S: <creData xmlns="urn:X-ar:params:xml:ns:variant-1.0">

S: <variant userForm="δείγua.idn">xn--ua-48bqdg.idn</variant>

S: </creData>

S: </extension>

S: <trID>

S: <clTRID>ABC-12345</clTRID>

S: <svTRID>54321-XYZ</svTRID>

S: </trID>

S: </response>

S: </epp>

### EPP <delete> Command

This extension does not add any elements to the EPP <delete> command or <delete> response described in the EPP domain mapping [1].

### EPP <renew> Command

This extension does not add any elements to the EPP <renew> command or <renew> response described in the EPP domain mapping [1].

### EPP <transfer> Command

This extension does not add any elements to the EPP <transfer> command or <transfer> response described in the EPP domain mapping [1].

### EPP <update> Command

This extension adds elements to the EPP <update> command described in the EPP domain mapping [1]. In addition to the command elements in the domain mapping, the <update> command MUST contain an EPP <extension> element, which MUST contain an <update> element that identifies the variant namespace. The <update> element contains the following elements:

* An OPTIONAL <add> element that contains the domain variants to be activated.
* An OPTIONAL <rem> element that contains the domain variants to be deactivated.

At least one <add> and <rem> element MUST be provided. The <add> and <rem> elements MUST contain the following elements:

* One or more <variant> elements, each containing the fully qualified name of a domain variant, in DNS form. Each <variant> element MUST have a "userForm" attribute indicating the native presentation form of the variant name.

Example <update> command:

C: <?xml version="1.0" encoding="UTF-8"?>

C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">

C: <command>

C: <update>

C: <update xmlns="urn:ietf:params:xml:ns:domain-1.0">

C: <name>xn--kxaefcf8b.idn</name>

C: </update>

C: </update>

C: <extension>

C: <update xmlns="urn:X-ar:params:xml:ns:variant-1.0">

C: <add>

C: <variant userForm="δeiyua.idn">xn--eiyua-u9d.idn</variant>

C: <variant userForm="δείγua.idn">xn--ua-48bqdg.idn</variant>

C: </add>

C: <rem>

C: <variant userForm="δείγμa.idn">xn--a-ulbmcf8b.idn</variant>

C: <variant userForm="δeiγua.idn">xn--eiua-cmdd.idn</variant>

C: </rem>

C: </update>

C: </extension>

C: <clTRID>ABC-12345</clTRID>

C: </command>

C: </epp>

# Formal Syntax

An EPP protocol mapping is specified in XML Schema notation. The formal syntax presented here is a complete schema representation of the object mapping suitable for automated validation of EPP XML instances. The BEGIN and END tags are not part of the schema; they are used to note the beginning and ending of the schema for URI registration purposes.

**BEGIN**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<schema targetNamespace=*"urn:X-ar:params:xml:ns:variant-1.0"*

xmlns:variant=*"urn:X-ar:params:xml:ns:variant-1.0"* xmlns:epp=*"urn:ietf:params:xml:ns:epp-1.0"*

xmlns:eppcom=*"urn:ietf:params:xml:ns:eppcom-1.0"* xmlns=*"http://www.w3.org/2001/XMLSchema"*

elementFormDefault=*"qualified"*>

<!--

Import common EPP element types.

-->

<import namespace=*"urn:ietf:params:xml:ns:epp-1.0"*

schemaLocation=*"epp-1.0.xsd"* />

<import namespace=*"urn:ietf:params:xml:ns:eppcom-1.0"*

schemaLocation=*"eppcom-1.0.xsd"* />

<annotation>

<documentation>Internationalised Domain Name

Extensions to the Extensible Provisioning Protocol v1.0 schema.

Object mappings to variant-specific commands.</documentation>

</annotation>

<!--

Custom command extensions

-->

<element name=*"info"* type=*"variant:infoType"* />

<element name=*"update"* type=*"variant:updateType"* />

<element name=*"variantInfo"* type=*"variant:nameType"* />

<!--

Custom response extensions

-->

<element name=*"infData"* type=*"variant:resDataType"* />

<element name=*"creData"* type=*"variant:resDataType"* />

<element name=*"varInfData"* type=*"variant:resDataType"* />

<!--

Child elements of the <info> command extension.

-->

<complexType name=*"infoType"*>

<attribute name=*"variants"* type=*"variant:variantsInfoAttributeType"*

default=*"all"* />

</complexType>

<simpleType name=*"variantsInfoAttributeType"*>

<restriction base=*"token"*>

<enumeration value=*"all"* />

<enumeration value=*"none"* />

</restriction>

</simpleType>

<!--

Child elements of the <update> command extension.

-->

<complexType name=*"updateType"*>

<sequence>

<element name=*"add"* type=*"variant:addRemType"* minOccurs=*"0"* maxOccurs=*"1"*/>

<element name=*"rem"* type=*"variant:addRemType"* minOccurs=*"0"* maxOccurs=*"1"* />

</sequence>

</complexType>

<complexType name=*"addRemType"*>

<sequence>

<element name=*"variant"* type=*"variant:variantType"* maxOccurs=*"unbounded"* />

</sequence>

</complexType>

<!--

Child elements of the <variantInfo> command.

-->

<complexType name=*"nameType"*>

<sequence>

<element name=*"name"* type=*"variant:internationalisedLabelType"* />

</sequence>

</complexType>

<complexType name=*"internationalisedLabelType"*>

<simpleContent>

<extension base=*"eppcom:labelType"*>

<attribute name=*"language"* type=*"language"*>

<annotation>

<documentation>

Registration of language with IANA requires the

definition

of a Script or Language Designator

(http://www.iana.org/procedures/idn-repository.html).

The linked

document above notes that Language Designators

are defined in BCP

47

(http://www.rfc-editor.org/rfc/bcp/bcp47.txt), which

satisfies

the requirements of the language datatype and

RFC3066 (BCP 47,

Section 2.2.8).

</documentation>

</annotation>

</attribute>

</extension>

</simpleContent>

</complexType>

<!--

Response extension elements.

-->

<complexType name=*"resDataType"*>

<sequence>

<element name=*"variant"* type=*"variant:variantType"* maxOccurs=*"unbounded"* />

</sequence>

</complexType>

<complexType name=*"variantType"*>

<simpleContent>

<extension base=*"eppcom:labelType"*>

<attribute name=*"userForm"* *type="eppcom:labelType"* use=*"required"* />

</extension>

</simpleContent>

</complexType>

<!--

End of schema.

-->

</schema>

**END**

# Internationalization Considerations

EPP is represented in XML, which provides native support for encoding information using the Unicode character set and its more compact representations, including UTF-8 [8]. Conformant XML processors recognize both UTF-8 and UTF-16 [9]. Though XML includes provisions to identify and use other character encodings through use of an "encoding" attribute in an <?xml?> declaration, use of UTF-8 is RECOMMENDED in environments where parser encoding support incompatibility exists.

As an extension of the EPP domain mapping [1], the elements, element content, attributes, and attribute values described in this document MUST inherit the internationalization conventions used to represent higher-layer domain and core protocol structures present in an XML instance that includes this extension.

# Security Considerations

The mapping extensions described in this document do not provide any security services beyond those described by the EPP core protocol specification [2].

# References

## Normative References

[1] Hollenbeck, S., "Extensible Provisioning Protocol (EPP) Domain Name Mapping", RFC 5731, August 2009

[2] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)", RFC 5730, August 2009.

[3] Khallouf, J., Treadwell, I., “Variant Protocol Extension for the Extensible Provisioning Protocol (EPP)”, August 2009

[4] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

## Informative References

[5] Maler, E., Yergeau, F., Paoli, J., Bray, T., and C. Sperberg-McQueen, "Extensible Markup Language (XML) 1.0 (Third Edition)", World Wide Web Consortium FirstEdition REC-xml-20040204, February 2004, <<http://www.w3.org/TR/2004/REC-xml-20040204>>.

[6] Thompson, H., Maloney, M., Mendelsohn, N., and D. Beech, "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>>.

[7] Biron, P. and A. Malhotra, "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>>.

[8] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

[9] Hoffman, P. and F. Yergeau, "UTF-16, an encoding of ISO 10646", RFC 2781, February 2000.