1 Abstract

This document describes an Extensible Provisioning Protocol (EPP) extension mapping for the provisioning and management of Domain Name extensions for domain objects stored in a shared central repository. Specified in XML, the mapping extends the EPP domain name mapping to provide additional features required for the control of the DNServices Registry Domain Objects.

Contents

2 Introduction

This extension provides additional functionality to the Domain object as described in RFC 5731. The additional functionality is listed below:

1. Auto Renew

2. Cancel Pending Action

3 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 BCP 14, RFC 2119.
In examples, "C:" represents lines sent by a protocol client, and "S:" represents lines returned by a protocol server. "////" is used to note element values that have been shortened to better fit page boundaries. 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 in order to develop a conforming implementation.

gtldd is used as an abbreviation for http://co.za/epp/extensions/gtlddomain-1-0.

4 Object Attributes

This extension adds an Auto Renew attribute to a domain name object.

4.1 Auto Renew

The auto renew flag is a boolean flag used to control the renew functionality around a domain upon expiry. If the flag is set to TRUE then the domain will be automatically renewed in the Registry assuming:

1. There are sufficient funds

2. There are subordinate host dependencies on the domain

5 EPP Command Mapping

6 EPP Query Commands

6.1 EPP `<check>` Command

This extension does not add any elements to the EPP `<check>` command or `<check>` response described in the EPP domain mapping RFC 5731.

6.2 EPP `<info>` Command

This extension does not add any elements to the EPP `<info>` command described in the EPP domain mapping RFC 5731. However, additional elements are defined for the `<info>` response.

When an `<info>` command has been processed successfully, the EPP `<resData>` element MUST contain child elements as described in the EPP domain mapping RFC 5731. In addition, the EPP `<extension>` element MAY contain a child `<gtldd:infData>` element that identifies the extension namespace if the domain object has data associated with this extension and based on server policy. The `<gtldd:infData>` element contains the following child elements:

- An OPTIONAL `<gtldd:autoenew>` element that indicates the domain object preference for automatic renewal

Example `<info>` Response for Auto Renew:

```
S:<epp:epp xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
S: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
S: xmlns:gtldd="http://co.za/epp/extensions/gtlddomain-1-0">
S: <epp:response>
S:   <epp:result code="1000">
S:     <epp:msg>Domain Info Command completed successfully</epp:msg>
S:   </epp:result>
S:   <epp:resData>
S:     <domain:infData>
S:       <domain:name>exampledomain.gtld</domain:name>
S:       <domain:roid>DOM_2W-COZA</domain:roid>
S:       <domain:status s="ok">Domain Creation</domain:status>
S:       <domain:registrant>testCont</domain:registrant>
S:       <domain:ns>
```

```
S:            <domain:hostAttr>
S:              <domain:hostName>ns1.otherdomain.gtld</domain:hostName>
S:            </domain:hostAttr>
S:            <domain:hostAttr>
S:              <domain:hostName>ns2.otherdomain.gtld</domain:hostName>
S:            </domain:hostAttr>
S:          </domain:ns>
S:          <domain:clID>testrar1</domain:clID>
S:          <domain:crID>testrar1</domain:crID>
S:          <domain:crDate>2011-02-23T14:43:12Z</domain:crDate>
S:          <domain:upID>testrar1</domain:upID>
S:          <domain:upDate>2011-02-23T14:46:18Z</domain:upDate>
S:          <domain:exDate>2013-02-22T14:43:12Z</domain:exDate>
S:        </domain:infData>
S:      </epp:resData>
S:      <epp:extension>
S:        <gtldd:infData>
S:          <gtldd:autorenew>false</gtldd:autorenew>
S:        </gtldd:infData>
S:      </epp:extension>
S:      <epp:trID>
S:        <epp:clTRID>CLTRID-12984723857-97L2</epp:clTRID>
S:        <epp:svTRID>DNS-EPP-12E52FC3CEB-A80EF</epp:svTRID>
S:      </epp:trID>
S:    </epp:response>
S:</epp:epp>
```

6.3 EPP `<transfer>` Command

This extension does not add any elements to the EPP `<transfer>` command
or `<transfer>` response described in the EPP domain mapping RFC 5731.

7 EPP Transform Commands

7.1 EPP `<create>` Command

This extension defines additional elements for the EPP `<create>` command
described in the EPP domain mapping RFC 5731. The additional auto re-
new elements are defined for the EPP `<create>` response as follows.
The EPP `<create>` command provides a transform operation that allows a
client to create a domain object. In addition to the EPP command elements

described in the EPP domain mapping RFC 5731, the command MAY contain an `<extension>` element, and the `<extension>` element MAY contain a child `<gtldd:create>` element that identifies the extension namespace if the client wants to associate data defined in this extension to the domain object.

The `<gtldd:create>` element contains the following child elements:

- An OPTIONAL `<gtldd:autorenew>` element that indicates a child's preference to automatically renew this domain object upon expiration.

Example `<create>` Command for autorenew false:

```
C:<epp:epp xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
C: xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
C: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
C: xmlns:gtldd="http://co.za/epp/extensions/gtlddomain-1-0"
C: xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
C: <epp:command>
C:   <epp:create>
C:     <domain:create
C:       xsi:schemaLocation="urn:ietf:params:xml:ns:domain-1.0
C:       domain-1.0.xsd">
C:       <domain:name>exampledomain.gtld</domain:name>
C:       <domain:ns>
C:         <domain:hostAttr>
C:           <domain:hostName>ns1.exampledomain.gtld</domain:hostName>
C:           <domain:hostAddr ip="v4">160.124.24.57</domain:hostAddr>
C:         </domain:hostAttr>
C:         <domain:hostAttr>
C:           <domain:hostName>ns2.exampledomain.gtld</domain:hostName>
C:           <domain:hostAddr ip="v4">160.124.24.58</domain:hostAddr>
C:         </domain:hostAttr>
C:       </domain:ns>
C:       <domain:registrant>rant1</domain:registrant>
C:       <domain:authInfo>
C:         <domain:pw>coza</domain:pw>
C:       </domain:authInfo>
C:     </domain:create>
C:   </epp:create>
C:   <epp:extension>
C:     <gtldd:create>
C:       <gtldd:autorenew>false</gtldd:autorenew>
C:     </gtldd:create>
```

```
C:    </epp:extension>
C: </epp:command>
C:</epp:epp>
```

When a `<create>` command has been processed successfully, the EPP response is as described in the EPP domain mapping RFC 5731 with the extension element as follows:

```
S:  <epp:extension>
S:    <gtldd:gtldData>
S:      <gtldd:detail result="success">
S:          AutoRenew 'False' successful
S:      </gtldd:detail>
S:    </gtldd:gtldData>
S:  </epp:extension>
```

7.2 EPP `<delete>` Command

This extension does not add any elements to the EPP `<delete>` command or `<delete>` response described in the EPP domain mapping RFC 5731.

7.3 EPP `<renew>` Command

Although this extension does not add any elements to the EPP `<renew>` command or `<renew>` response described in the EPP domain mapping RFC 5731 it does extend the Registry's handling of the domain object upon expiry.

7.4 EPP `<transfer>` Command

This extension does not add any elements to the EPP `<transfer>` command or `<transfer>` response described in the EPP domain mapping RFC 5731.

7.5 EPP `<update>` Command

This extension defines additional elements for the EPP `<update>` command described in the EPP domain mapping RFC 5731. The additional elements and attributes are defined for the EPP `<update>` response as follows.
The EPP `<update>` command provides a transform operation that allows a client to modify the attributes of a domain object. In addition to the EPP command elements described in the EPP domain mapping, the command MAY contain an `<extension>` element, and the `<extension>` element MAY

contain a child `<gtldd:update>` element that identifies the extension namespace if the client wants to update the domain object with data defined in this extension. The `<gtldd:update>` element MAY contain a `<gtldd:chg>` element. The `<gtldd:chg>` element contains a `<gtldd:autorenew>` element to adjust the automatic renewal status of a domain object.

The `<gtldd:update>` element also contains an OPTIONAL "cancelPendingAction" attribute that a client can use to ask the server operator to cancel a predefined action as provided by the Registry software. This attribute accepts XML token values meaning standard text without leading or trailing whitespace.

The `<gtldd:update>` element contains the following child elements:

- **An OPTIONAL `<gtldd:chg>` element** that contains a `<gtldd:autorenew>` element that is used to adjust the auto renew flag on the domain object.

- **An OPTIONAL cancelPendingAction attribute** that contains the predefined action name as provided by the server.

Example `<update>` Command for autorenew false:

```
C:<epp:epp xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
C: xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
C: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
C: xmlns:gtldd="http://co.za/epp/extensions/gtlddomain-1-0"
C: xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
C: <epp:command>
C:   <epp:update>
C:     <domain:update
C:       xsi:schemaLocation="urn:ietf:params:xml:ns:domain-1.0
C:       domain-1.0.xsd">
C:       <domain:name>exampledomain.gtld</domain:name>
C:     </domain:update>
C:   </epp:update>
C:   <epp:extension>
C:     <gtldd:update>
C:       <gtldd:chg>
C:         <gtldd:autorenew>false</gtldd:autorenew>
C:       </gtldd:chg>
C:     </gtldd:update>
C:   </epp:extension>
C:  </epp:command>
C:</epp:epp>
```

When the `<update>` command has been processed successfully, the EPP response is as described in the EPP domain mapping RFC 5731 with the extension element as follows:

```
S:<epp:epp xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
S: xmlns:gtldd="http://co.za/epp/extensions/gtlddomain-1-0">
S: <epp:response>
S:   <epp:result code="1001">
S:     <epp:msg>Domain action 'PendingUpdate' pending</epp:msg>
S:   </epp:result>
S:   <epp:extension>
S:     <gtldd:gtldData>
S:       <gtldd:detail result="success">
S:         AutoRenew 'False' successful
S:       </gtldd:detail>
S:     </gtldd:gtldData>
S:   </epp:extension>
S:   <epp:trID>
S:     <epp:clTRID>CLTRID-12984717630-F490</epp:clTRID>
S:     <epp:svTRID>DNS-EPP-12E52F2BC78-8AC51</epp:svTRID>
S:   </epp:trID>
S: </epp:response>
S: </epp:epp>
```

If a domain object enters a deletion process through expiry or command then the action MAY be cancelled.
Example `<update>` Command for cancelling a pending action:

```
C:<epp:epp xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
C: xmlns:epp="urn:ietf:params:xml:ns:epp-1.0"
C: xmlns:domain="urn:ietf:params:xml:ns:domain-1.0"
C: xmlns:gtldd="http://co.za/epp/extensions/gtlddomain-1-0"
C: xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
C: <epp:command>
C:   <epp:update>
C:     <domain:update
C:       xsi:schemaLocation="urn:ietf:params:xml:ns:domain-1.0
C:       domain-1.0.xsd">
C:       <domain:name>exampledomain.gtld</domain:name>
C:     </domain:update>
C:   </epp:update>
C:   <epp:extension>
C:     <gtldd:update cancelPendingAction="PendingDeletion"/>
```

```
C:    </epp:extension>
C: </epp:command>
C:</epp:epp>
```

When the `<update>` command has been processed successfully, the EPP
response is as described in the EPP domain mapping RFC 5731. However
the action that was specified MUST be cancelled and any status effects on
that domain object removed. If the action is not pending or does not exist
then an appropriate message is returned to the client.

8   Formal Syntax

An EPP object 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="http://co.za/epp/extensions/gtlddomain-1-0"
  xmlns:gtldd="http://co.za/epp/extensions/gtlddomain-1-0"
  xmlns="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <annotation>
    <documentation>
      Extensible Provisioning Protocol v1.0 domain command extension ////
      schema for gTLD required extensions </documentation>
  </annotation>

  <element name="create" type="gtldd:createType"/>
  <element name="update" type="gtldd:updateType"/>
  <element name="infData" type="gtldd:infoResponseType"/>
  <element name="gtldData" type="gtldd:gtldDataType"/>

  <complexType name="chgType">
    <sequence>
      <element name="autorenew" type="gtldd:autoRenewType" minOccurs="0"/>
    </sequence>
  </complexType>
```

```xml
<complexType name="updateType">
  <sequence>
    <element name="chg" type="gtldd:chgType" minOccurs="0"/>
  </sequence>
  <attribute name="cancelPendingAction" type="string" use="optional"/>
</complexType>

<complexType name="createType">
  <sequence>
    <element name="autorenew" type="gtldd:autoRenewType" minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="infoResponseType">
  <sequence>
    <element name="autorenew" type="gtldd:autoRenewType" minOccurs="0"/>
  </sequence>
</complexType>
<complexType name="gtldDataType">
  <sequence>
    <element name="detail">
      <complexType>
        <simpleContent>
          <extension base="string">
            <attribute name="result" type="gtldd:resultType" use="required"/>
          </extension>
        </simpleContent>
      </complexType>
    </element>
  </sequence>
</complexType>

<simpleType name="resultType">
  <restriction base="NMTOKEN">
    <enumeration value="success"/>
    <enumeration value="failure"/>
  </restriction>
</simpleType>

<simpleType name="autoRenewType">
  <restriction base="boolean">
  </restriction>
</simpleType>
```

```
</schema>
END
```