

# XML Mark Data Extension for the CORE Registration System

CORE Association

# 1. XML Mark Data Extension for specifying augmented trademark information during Launch Phases

The CORE Registration System fully supports the standard launch phase EPP extension (as defined at <http://tools.ietf.org/html/rfc8334.txt>) for the submission of launch phase and trademark information during a registry's launch phases. However, while this standard extension does offer sufficient support for ordinary landrush, TMCH sunrise and TMCH claims based launch phases, it does not provide means to collect any additional information from registrars/registrants if the need arises. For this purpose, the CORE Registration System provides support for "augmented" mark data via custom XML element usable as mark data within the standard launch phase extension. Details about this augmented mark data are provided in this document.

## 1.1 Introduction

Some registries may require registrants (and, in turn, their registrars) to provide additional information during certain launch phases and submit this information along with standard domain data in an EPP request. The launch phase EPP extension (specified at <http://tools.ietf.org/html/rfc8334.txt>) provides standardized support for the most common requirements, i.e. for specifying the launch phase itself, plus (signed or unsigned) trademark data. In some situations, this may not be sufficient. For example, a registry for a TLD representing a certain geographical region may choose to devise a launch phase during which local public authorities shall get a chance to register certain names; such a launch phase may require the specification of additional information needed to validate a domain application, such as e.g. a description of the public authority, a URL where the public authority may be validated, or a reference URL at which an existing public online presence of the authority may be found.

To accommodate such requirements, the CORE Registration System supports an extension of the mark data definitions used by the standard launch phase extension that allows the specification of arbitrary key-value data (plus free text information) alongside standard mark data. This document describes how this mark data extension is represented in EPP/XML.

## 1.2 Implementation Details

In the following, the XML name space prefixes "mark" and "smd" shall refer to the "urn:ietf:params:xml:ns:mark-1.0" and "urn:ietf:params:xml:ns:signed-Mark-1.0" name spaces, both as specified at <http://tools.ietf.org/html/rfc7848>.

This extension for augmented mark data utilizes the extension mechanism devised by the standard launch phase extension, which is based on XSD substi-

tution groups. Specifically, the `<augmentedmark>` element defined by this extension belongs to the substitution group "smd:abstractSignedMark" and uses an XML schema type that extends the "smd:abstractSignedMarkType" type defined by the launch phase extension. Consequently, the `<augmentedMark>` element may substitute, i.e. occur in place of (non-augmented) standard mark data within the launch phase extension. Practically, this means that a registrar must simply use an `<augmentedMark>` element instead of standard mark within the launch phase EPP extension whenever a launch phase requires mandatory (or allows optional) augmented data; the actual mark data may be used (unaltered) as an inner element within the `<augmentedMark>` element (see below).

Note: this means that the augmented mark data is not implemented via a "top-level" EPP extension, but is merely utilizing an extension mechanism that allows extending the types of mark data accepted by the existing launch phase extension. This approach should facilitate the adoption by registrars using EPP.

The content of the `<augmentedMark>` element is structured as follows:

- First, an (optional) element containing the actual mark data may be present; this may either be a `<mark:abstractMark>`, `<smd:abstractSignedMark>` or `<smd:encodedSignedMark>` element. This element is optional, allowing the augmented mark to be used as a pure key-value based data set if a launch phase does not require actual mark data.
- Second, one or more `<applicationInfo>` elements must follow which contain the additional information (the value). An optional "type" element may indicate the key for which the value supplied data. If the "type" element is omitted, the content of the `<applicationInfo>` element represents free-text information with no specific semantics. Note that the same value for the "type" element may not occur more than once; also, no more than one `<applicationInfo>` with the "type" element omitted may occur. The order in which the `<applicationInfo>` elements occur is irrelevant.

Note: the `<augmentedMark>` element provides a generic way to express free-text and key/value based data. The actual types (keys) and values required/allowed in `<applicationInfo>` elements within an `<augmentedMark>` element, as well as the requirement for mark data, are defined by registry policy and may vary between different launch phases of the same registry. Please refer to the respective registry's policy documentation for more information on the required/optional `<applicationInfo>` type attributes and respective expected element content.

### 1.3 Examples

The following `<augmentedMark>` example specifies standard encoded mark data (as received from the TMCH) and augments it with a free-text `<applicationInfo>` plus two `<applicationInfo>` elements (with types "example-type-1" and "example-type-2"):

```

<ext:augmentedMark xmlns:ext="http://xmlns.corenic.net/epp/mark-ext-1.0">
  <smd:encodedSignedMark xmlns:smd="urn:iETF:params:xml:ns:signedMark-1.0">
    PD94bWwgdMvYc2lVbJ0iMS4wIiB1bWVZG1uZz0iVVRGLTgiPz4KPHNtZDpzaWduZWRNYXJrIHht
    bG5zOnNtZD0idXJ0m1ldGY6cGFyYw1zOnhtbDpuc2pzaWduZWRNYXJrLTEuMCIgaWQ9I182YzMw
    ...
    VnFhUT08L2Rz01g1MD1DZXJ0aWZpY2F0ZT48L2Rz01g1MD1EYXRhPjwvZHM6S2V5SW5mbz48L2Rz
    01NpZ25hdHVyZT48L3NtZDpzaWduZWRNYXJrPg==
  </smd:encodedSignedMark>
  <ext:applicationInfo>Arbitrary text goes here.</ext:applicationInfo>
  <ext:applicationInfo type="example-type-1">Example Content</ext:applicationInfo>
  <ext:applicationInfo type="example-type-2">other example content</ext:applicationInfo>
</ext:augmentedMark>
  
```

In contrast, the following <augmentedMark> example contains a single, free-text <applicationInfo> element and omits mark data entirely:

```

<ext:augmentedMark xmlns:ext="http://xmlns.corenic.net/epp/mark-ext-1.0">
  <ext:applicationInfo>Arbitrary text goes here.</ext:applicationInfo>
</ext:augmentedMark>
  
```

Finally, the following XML document represents a full <domain:create> EPP command that utilizes the <augmentedMark> from the first example above to submit augmented mark data for an application during the "somelaunch-phase" launch phase:

```

<?xml version="1.0" encoding="UTF-8"?>
<epp xmlns="urn:iETF:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create xmlns:domain="urn:iETF:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:period unit="y">2</domain:period>
        <domain:ns>
          <domain:hostObj>ns1.iroDNS.net</domain:hostObj>
          <domain:hostObj>ns2.iroDNS.net</domain:hostObj>
        </domain:ns>
        <domain:registrant>CONTACT-1</domain:registrant>
        <domain:contact type="admin">CONTACT-1</domain:contact>
        <domain:contact type="tech">CONTACT-1</domain:contact>
        <domain:contact type="billing">CONTACT-1</domain:contact>
        <domain:authInfo>
          <domain:pw>zS2fB#CR[1QgI#S0</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
    <extension>
      <launch:create xmlns:launch="urn:iETF:params:xml:ns:launch-1.0">
        <launch:phase name="somelaunchphase">custom</launch:phase>
        <ext:augmentedMark xmlns:ext="http://xmlns.corenic.net/epp/mark-ext-1.0">
          <smd:encodedSignedMark xmlns:smd="urn:iETF:params:xml:ns:signedMark-1.0">
            PD94bWwgdMvYc2lVbJ0iMS4wIiB1bWVZG1uZz0iVVRGLTgiPz4KPHNtZDpzaWduZWRNYXJrIHht
            bG5zOnNtZD0idXJ0m1ldGY6cGFyYw1zOnhtbDpuc2pzaWduZWRNYXJrLTEuMCIgaWQ9I182YzMw
            ...
            VnFhUT08L2Rz01g1MD1DZXJ0aWZpY2F0ZT48L2Rz01g1MD1EYXRhPjwvZHM6S2V5SW5mbz48L2Rz
            01NpZ25hdHVyZT48L3NtZDpzaWduZWRNYXJrPg==
          </smd:encodedSignedMark>
          <ext:applicationInfo>Arbitrary text goes here.</ext:applicationInfo>
          <ext:applicationInfo type="example-type-1">Example Content</ext:applicationInfo>
          <ext:applicationInfo type="example-type-2">other example content</ext:applicationInfo>
        </ext:augmentedMark>
      </launch:create>
    </extension>
  </command>
</epp>
  
```

## 1.4 Formal Syntax

This XML schema definition provides a formal description of the XML structures defined by this extension:

```

<?xml version="1.0"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://xmlns.corenic.net/epp/mark-ext-1.0"
  xmlns:ext="http://xmlns.corenic.net/epp/mark-ext-1.0"
  xmlns:mark="urn:ietf:params:xml:ns:mark-1.0"
  xmlns:smd="urn:ietf:params:xml:ns:signedMark-1.0"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">

  <annotation>
    <documentation>
      extension schema for the submission of additional
      launch phase/application data if standard mark data
      is not sufficient; the schema defines an "augmentedMark" element
      which type allows it to be used in place of standard mark data
      in the launch phase extension (http://tools.ietf.org/html/rfc7848)
      and to augment mark data with arbitrary, key-value style additional
      information required for evaluating domain name applications
    </documentation>
  </annotation>

  <!-- import mark data definitions -->

  <import namespace="urn:ietf:params:xml:ns:mark-1.0"/>
  <import namespace="urn:ietf:params:xml:ns:signedMark-1.0"/>

  <!-- type for application info values -->

  <simpleType name="applicationInfoValueType">
    <restriction base="normalizedString">
      <minLength value="1"/>
      <maxLength value="2048"/>
    </restriction>
  </simpleType>

  <!-- type for application info types -->

  <simpleType name="applicationInfoTypeType">
    <restriction base="token">
      <minLength value="1"/>
      <maxLength value="64"/>
    </restriction>
  </simpleType>

  <!-- type for application info data -->

  <complexType name="applicationInfoType">
    <simpleContent>
      <extension base="ext:applicationInfoValueType">
        <!-- optional attribute to denote the type of information provided -->
        <attribute name="type" type="ext:applicationInfoTypeType"
          use="optional"/>
      </extension>
    </simpleContent>
  </complexType>

  <!-- sub-type of abstractSignedMarkType suitable for
  use in the launch phase extension -->

  <complexType name="augmentedMarkType">
    <complexContent>
      <extension base="smd:abstractSignedMarkType">
        <sequence>

          <!-- optional embedded standard mark data;
          unsigned, signed or encoded -->
          <choice minOccurs="0">
            <element ref="mark:abstractMark"/>
          </choice>
        </sequence>
      </extension>
    </complexContent>
  </complexType>

```

```
<element ref="smd:abstractSignedMark"/>
<element ref="smd:encodedSignedMark"/>
</choice>

    <!-- additional information for a domain application;
         at least one must be provided -->
    <element name="applicationInfo" type="ext:applicationInfoType"
        maxOccurs="unbounded"/>

</sequence>
</extension>
</complexContent>
</complexType>

<!-- top-level element for augmented mark data -->

<element name="augmentedMark" type="ext:augmentedMarkType"
    substitutionGroup="smd:abstractSignedMark"/>

</schema>
```