

CLEARINGHOUSE



TMCH Automated interface (API 2)



TABLE OF CONTENTS

1. INTRODUCTION	4
● 1.1. Documents included	4
● 1.2. Objects	4
● 1.3. Session control	5
● 1.4. Enumerations	5
2. XSD SCHEMA	6
3. CREATING A CONNECTION	20
● 3.1. First block	20
● 3.2. Second block	20
● 3.3. Labels	21
● 3.3.1. Algorithm	21
4. SESSION COMMANDS	23
● 4.1. LOGIN	23
● 4.2. LOGOUT	23
● 4.3. HELLO	24
5. CREATE MARK	25
● 5.1. Create registered trademark	25
● 5.2. Create court validated mark	27
● 5.3. Create statute or treaty	28
● 5.3.1. Create mark response	30
6. INFO MARK	31
● 6.1. Info mark command	31
● 6.2. Info mark smd	32
● 6.3. Info mark encoded smd	34
● 6.4. Info mark file	37
● 6.5. Info court validated mark	39
● 6.6. Info statute or treaty	40
7. UPDATE MARK	42
● 7.1. Update mark settings	42
● 7.2. Adding a label	43
● 7.3. Changing label flags	43
8. UDRP OR COURT CASE	45
● 8.1. Adding a udrp case	45
● 8.2. Adding a court case	45
● 8.3. Managing labels linked to a case	46
● 8.4. Managing documents linked to a case	46
● 8.5. Changing a case	47
● 8.6. Retrieve case data	47
9. RENEW	50
10. CHECK MARK	51
11. POLL	52
● 11.1. Poll request	52
● 11.1.1. Poll response (no msg)	52
● 11.1.2. Poll response (msg pending)	52
● 11.1.3. Poll response (not a mark)	53



● 11.2.	Poll ack (dequeue)	53
● 11.2.1.	Poll ack (response last message)	53
● 11.2.2.	Poll ack (response more message)	54
12. TRANSFER		55
● 12.1.	Transfer – request to execute	55
● 12.1.1.	Transfer response	55
13. ERROR RESPONSE		56
14. APPENDIX 1		57
● 14.1.	Country codes	57
● 14.2.	Jurisdictions	62
● 14.3.	Languages	66
15. APPENDIX 2		69
● 15.1.	Poll codes	69
● 15.2.	API error codes	69
16. APPENDIX 3		72
● 16.1.	Transliteration of & and @ per country and per jurisdiction.	72



1. INTRODUCTION

The TMCH project provides an automated interface to allow users to interact in an automated fashion with the data stored in the Clearinghouse database (TMCH database). XML has been widely used by registries and registrars to provide an automated interfaces (the API) for their data exchange. This API provides similar functionality and will be provided using TLS to be as close to the current EPP standard as possible. The API may only be used by Trademark Agents that have contracted with the Clearinghouse.

This document contains some sample of commands and responses based on the XSD that should help building implementations to use the API provided to get direct access to the various objects in the TMCH database in an automated fashion.

The complete XSD scheme can be found under the section XSD schemes. It extends the Signed Mark Data (SMD) XSD description draft-lozano-smd-01.txt that was provided by ICANN. The structure of the mark block is intentionally included and not modified to prevent misunderstanding, make the mapping easy and prevent misinterpretation of the various fields.

1.1. Documents included

- XSD schemes
- Examples
- Appendixes with various lists

The provided schema allows the user of the automated interface to verify the syntactical correctness of the request sent to TMCH and the response returned. It details to a certain extent the possible options that are allowed in some fields while determining the format of other fields based on regular expressions.

This schema does not provide a full policy description nor does it specify complex intricacies that exists between certain types and certain required fields.

1.2. Objects

There are 3 basic objects

1. Mark (a complex object which can take 3 forms: Registered Trademarks/Court Validated Marks/Marks protected by statute or treaty)
2. Label (the domain name without its TLD extension)
3. Uploaded document (implicitly linked to mark and as such not directly accessible using the automated interface)

We allow the 'C(reate)R(ead)U(pdate) schema for the Mark and Label object, using create, info and update/change commands to insert, retrieve and maintain these objects. This should provide an adequate means of creating a mechanism to synchronize your own local database with the TMCH database.



1.3. Session control

To create a new session the **login command** can be used and to terminate the session the **logout command** should be used. To verify if the connection with the server is still up and running the **hello command** is provided. The **hello command** can also be used to keep an open session alive preventing the server disconnecting after the session has been idle for a 5 minutes. Due to the nature of certain mark characteristics some objects will have extra policy determined constraints. These will be described in detail later. A few examples are listed below

- The 'Description of goods and services class' is comprised of a list of comma separated numbers for 1 to 45 or contains the value 46 when not applicable.
- A Registered trademark will need proof of use to be able to toggle the 'sunrise notification' on.
- A court validated mark needs a 'Court order' document attached when it is created

1.4. Enumerations

Currently the following enumerations can be found in the XSD schema

entitlement

- Owner
- Assignee
- Licensee

fileType

- pdf
- jpg

docClassType

- Trademark - Licensee declaration
- Trademark - Assignee declaration
- Trademark - Other
- Declaration of Proof of use including one single sample
- Proof of use - Other
- Copy of the court order

Other list of possible values such as country codes and jurisdictions can be found in Appendix



2. XSD SCHEMA

This XSD schema uses the external XSD schemas

1. mark-1.0 which can be found at <http://tools.ietf.org/html/draft-lozano-tmch-smd-02> section 3.2
2. signedMark-1.0 which can be found at <http://tools.ietf.org/html/draft-lozano-tmch-smd-02> section 3.1

```
<?xml version="1.0" encoding="utf-8"?>
<schema targetNamespace="urn:ietf:params:xml:ns:tmch-1.1"
  xmlns:mark="urn:ietf:params:xml:ns:mark-1.0"
  xmlns:smd="urn:ietf:params:xml:ns:signedMark-1.0"
  xmlns:tmch="urn:ietf:params:xml:ns:tmch-1.1"
  xmlns="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified">

  <annotation>
    <documentation>
      Trademark Clearing House Protocol v1.1 schema - DRAFT.
    </documentation>
  </annotation>

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

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

  <!--
  Every TMCH XML instance must begin with this element.
  -->
  <element name="tmch" type="tmch:tmchType"/>

  <!--
  An TMCH XML instance must contain a greeting, hello, command, or
  response.
  -->
  <complexType name="tmchType">
    <choice>
      <element name="greeting" type="tmch:greetingType"/>
      <element name="hello"/>
      <element name="command" type="tmch:commandType"/>
      <element name="response" type="tmch:responseType"/>
    </choice>
  </complexType>

  <!--
  A greeting is sent by a server in response to a client connection
  or <hello>.
  -->
  <complexType name="greetingType">
    <sequence>
      <element name="svID" type="tmch:svIDType"/>
      <element name="svDate" type="dateTime"/>
    </sequence>
  </complexType>
```



```

<!--
Server IDs are strings with minimum and maximum length restrictions.
-->
<simpleType name="svIDType">
  <restriction base="normalizedString">
    <minLength value="3"/>
    <maxLength value="64"/>
  </restriction>
</simpleType>

<complexType name="commandType">
  <sequence>
    <choice>
      <element name="check" type="tmch:mIDType"/>
      <element name="create" type="tmch:createType"/>
      <element name="info" type="tmch:infoType"/>
      <element name="login" type="tmch:loginType"/>
      <element name="logout"/>
      <element name="poll" type="tmch:pollType"/>
      <element name="renew" type="tmch:renewType"/>
      <element name="transfer" type="tmch:transferType"/>
      <element name="update" type="tmch:updateType"/>
    </choice>
    <element name="clTRID" type="tmch:trIDStringType"
      minOccurs="0"/>
  </sequence>
</complexType>

<!--
Child element of the <check> command
-->
<complexType name="mIDType">
  <sequence>
    <element name="id" type="mark:idType"
      maxOccurs="unbounded"/>
  </sequence>
</complexType>

<!--
Child elements of the <create> command.
-->
<complexType name="createType">
  <sequence>
    <element ref="mark:abstractMark"/>
    <element name="period" type="tmch:periodType"
      minOccurs="0"/>
    <element name="document" minOccurs="0" maxOccurs="unbounded"
      type="tmch:documentType"/>
    <element name="label" minOccurs="0" maxOccurs="unbounded"
      type="tmch:labelType"/>
  </sequence>
</complexType>

<!--
TMCH period type.
-->
<complexType name="periodType">
  <simpleContent>
    <extension base="tmch:pLimitType">
      <attribute name="unit" type="tmch:pUnitType"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>

```



```
<simpleType name="pLimitType">
  <restriction base="unsignedShort">
    <minInclusive value="1"/>
    <maxInclusive value="5"/>
  </restriction>
</simpleType>

<simpleType name="pUnitType">
  <restriction base="token">
    <enumeration value="y"/>
  </restriction>
</simpleType>

<!--
TMCH document type.
-->
<complexType name="documentType">
  <sequence>
    <element name="docType" type="tmch:docClassType"/>
    <element name="fileName" minOccurs="0" type="tmch:optionalStringType"/>
    <element name="fileType" type="tmch:fileType"/>
    <element name="fileContent" type="base64Binary"/>
  </sequence>
</complexType>

<simpleType name="docClassType">
  <restriction base="token">
    <enumeration value="tmLicenseeDecl"/>
    <enumeration value="tmAssigneeDecl"/>
    <enumeration value="tmOther"/>
    <enumeration value="declProofOfUseOneSample"/>
    <enumeration value="proofOfUseOther"/>
    <enumeration value="copyOfCourtOrder"/>
  </restriction>
</simpleType>

<simpleType name="fileType">
  <restriction base="token">
    <enumeration value="pdf"/>
    <enumeration value="jpg"/>
  </restriction>
</simpleType>

<simpleType name="requiredStringType">
  <restriction base="token">
    <minLength value="1"/>
    <maxLength value="255"/>
  </restriction>
</simpleType>

<simpleType name="optionalStringType">
  <restriction base="token">
  </restriction>
</simpleType>

<!--
Child elements of the <info> command.
-->
<complexType name="infoType">
  <sequence>
    <element name="id" type="mark:idType"/>
  </sequence>
  <attribute name="type" type="tmch:infoTypeType"/>
</complexType>
```




```

        default="info"/>
</complexType>

<simpleType name="infoTypeType">
  <restriction base="token">
    <enumeration value="enc"/> <!-- encodedSignedMark -->
    <enumeration value="file"/> <!-- like in web interface -->
    <enumeration value="info"/> <!-- info -->
    <enumeration value="smd"/> <!-- signedMark -->
  </restriction>
</simpleType>
<!--
The <login> command.
-->
<complexType name="loginType">
  <sequence>
    <element name="clID" type="tmch:clIDType"/>
    <element name="pw" type="tmch:pwType"/>
  </sequence>
</complexType>

<!--
The <poll> command.
-->
<complexType name="pollType">
  <attribute name="op" type="tmch:pollOpType"
    use="required"/>
  <attribute name="msgID" type="token"/>
</complexType>

<simpleType name="pollOpType">
  <restriction base="token">
    <enumeration value="ack"/>
    <enumeration value="req"/>
  </restriction>
</simpleType>

<!--
Child elements of the <renew> command.
-->
<complexType name="renewType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="curExpDate" type="date"/>
    <element name="period" type="tmch:periodType"
      minOccurs="0"/>
  </sequence>
</complexType>

<!--
Child elements of the <update> command.
-->
<complexType name="updateType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="case" type="tmch:caseType"
      minOccurs="0"/>
    <element name="add" type="tmch:addType"
      minOccurs="0"/>
    <element name="rem" type="tmch:remType"
      minOccurs="0"/>
    <element name="chg" type="tmch:chgType"
      minOccurs="0"/>
  </sequence>

```



```
</complexType>

<!--
Data elements that can be added or removed.
-->
<complexType name="addType">
  <sequence>
    <element name="document" type="tmch:documentType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="label" type="tmch:labelType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="case" type="tmch:caseType"
      minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="remType">
  <sequence>
    <element name="label" type="tmch:labelType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="case" type="tmch:caseType"
      minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<!--
Data elements that can be changed.
-->
<complexType name="chgType">
  <sequence>
    <element ref="mark:abstractMark"
      minOccurs="0"/>
    <element name="label" type="tmch:labelType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="case" type="tmch:caseType"
      minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="labelType">
  <sequence>
    <element name="aLabel" type="mark:labelType"/>
    <element name="smdInclusion" type="tmch:notifyType"
      minOccurs="0"/>
    <element name="claimsNotify" type="tmch:notifyType"
      minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="notifyType">
  <simpleContent>
    <extension base="normalizedString">
      <attribute name="enable" type="boolean" use="required"/>
    </extension>
  </simpleContent>
</complexType>

<complexType name="caseType">
  <sequence>
    <element name="id" type="tmch:caseIdType"/>
    <element name="udrp" type="tmch:udrpType"
      minOccurs="0"/>
    <element name="court" type="tmch:courtType"
      minOccurs="0"/>
  </sequence>
</complexType>
```



```

    <element name="document" type="tmch:caseDocumentType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="label" type="tmch:caseLabelType"
      minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="udrpType">
  <sequence>
    <element name="caseNo" type="token"/>
    <element name="udrpProvider" type="token"/>
    <element name="caseLang" type="tmch:requiredStringType"/>
  </sequence>
</complexType>

<complexType name="courtType">
  <sequence>
    <element name="refNum" type="token"/>
    <element name="cc" type="mark:ccType"/>
    <element name="region" type="token" minOccurs="0"
      maxOccurs="unbounded"/>
    <element name="courtName" type="token"/>
    <element name="caseLang" type="tmch:requiredStringType"/>
  </sequence>
</complexType>

<complexType name="caseDocumentType">
  <sequence>
    <element name="docType" type="tmch:caseDocClassType"/>
    <element name="fileName" minOccurs="0" type="tmch:optionalStringType"/>
    <element name="fileType" type="tmch:fileType"/>
    <element name="fileContent" type="base64Binary"/>
  </sequence>
</complexType>

<simpleType name="caseDocClassType">
  <restriction base="token">
    <enumeration value="courtCaseDocument"/>
    <enumeration value="tmOther"/>
  </restriction>
</simpleType>

<complexType name="caseLabelType">
  <sequence>
    <element name="aLabel" type="mark:labelType"/>
  </sequence>
</complexType>

<complexType name="transferType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="authCode" type="normalizedString"
      minOccurs="0"/>
  </sequence>
  <attribute name="op" type="tmch:transferOpType"
    use="required"/>
</complexType>

<simpleType name="transferOpType">
  <restriction base="token">
    <enumeration value="initiate"/>
    <enumeration value="execute"/>
  </restriction>
</simpleType>

```



```
<!--
<check> response types.
-->
<complexType name="reasonType">
  <simpleContent>
    <extension base="tmch:reasonBaseType">
      <attribute name="lang" type="language" default="en"/>
    </extension>
  </simpleContent>
</complexType>

<simpleType name="reasonBaseType">
  <restriction base="token">
    <minLength value="1"/>
    <maxLength value="32"/>
  </restriction>
</simpleType>

<!--
Abstract client and object identifier type.
-->
<simpleType name="clIDType">
  <restriction base="token">
    <minLength value="3"/>
    <maxLength value="16"/>
  </restriction>
</simpleType>

<simpleType name="pwType">
  <restriction base="token">
    <minLength value="6"/>
    <maxLength value="16"/>
  </restriction>
</simpleType>

<complexType name="trIDType">
  <sequence>
    <element name="clTRID" type="tmch:trIDStringType"
      minOccurs="0"/>
    <element name="svTRID" type="tmch:trIDStringType"/>
  </sequence>
</complexType>

<simpleType name="trIDStringType">
  <restriction base="token">
    <minLength value="3"/>
    <maxLength value="64"/>
  </restriction>
</simpleType>

<!--
Response types.
-->
<complexType name="responseType">
  <sequence>
    <element name="result" type="tmch:resultType"
      maxOccurs="unbounded"/>
    <element name="msgQ" type="tmch:msgQType"
      minOccurs="0"/>
    <element name="resData" type="tmch:resDataType"
      minOccurs="0"/>
    <element name="trID" type="tmch:trIDType"/>
  </sequence>
```



```
</complexType>

<complexType name="resultType">
  <sequence>
    <element name="msg" type="tmch:msgType"/>
    <choice minOccurs="0" maxOccurs="unbounded">
      <element name="value" type="tmch:errValueType"/>
      <element name="extValue" type="tmch:extErrValueType"/>
    </choice>
  </sequence>
  <attribute name="code" type="tmch:resultCodeType"
    use="required"/>
</complexType>

<complexType name="errValueType" mixed="true">
  <sequence>
    <any namespace="##any" processContents="skip"/>
  </sequence>
  <anyAttribute namespace="##any" processContents="skip"/>
</complexType>

<complexType name="extErrValueType">
  <sequence>
    <element name="value" type="tmch:errValueType"/>
    <element name="reason" type="tmch:msgType"/>
  </sequence>
</complexType>

<complexType name="msgQType">
  <sequence>
    <element name="qDate" type="dateTime"
      minOccurs="0"/>
    <element name="msg" type="tmch:mixedMsgType"
      minOccurs="0"/>
  </sequence>
  <attribute name="count" type="unsignedLong"
    use="required"/>
  <attribute name="id" type="tmch:minTokenType"
    use="required"/>
</complexType>

<complexType name="mixedMsgType" mixed="true">
  <sequence>
    <any processContents="skip"
      minOccurs="0" maxOccurs="unbounded"/>
  </sequence>
  <attribute name="lang" type="language"
    default="en"/>
</complexType>

<!--
Human-readable text may be expressed in languages other than English.
-->
<complexType name="msgType">
  <simpleContent>
    <extension base="normalizedString">
      <attribute name="lang" type="language" default="en"/>
    </extension>
  </simpleContent>
</complexType>

<!--
Non-empty token type.
-->
```



```
<simpleType name="minTokenType">
  <restriction base="token">
    <minLength value="1"/>
  </restriction>
</simpleType>

<!--
TMCH result codes.
-->
<simpleType name="resultCodeType">
  <restriction base="unsignedShort">
    <enumeration value="1000"/> <!-- Success -->
    <enumeration value="1300"/> <!-- Poll : no messages -->
    <enumeration value="1301"/> <!-- Poll : messages (use ack) -->
    <enumeration value="1500"/> <!-- Success & closing connection -->
    <enumeration value="2001"/> <!-- Syntax error -->
    <enumeration value="2201"/> <!-- Not logged in -->
    <enumeration value="2202"/> <!-- Login failed -->
    <enumeration value="2303"/> <!-- Object does not exist -->
    <enumeration value="2306"/> <!-- Policy error -->
    <enumeration value="2400"/> <!-- Command failed (internal server error) -->
  </restriction>
</simpleType>

<!--
TMCH resData type
-->
<complexType name="resDataType">
  <choice>
    <element name="chkData" type="tmch:chkDataType"/>
    <element name="creData" type="tmch:creDataType"/>
    <element name="infData" type="tmch:infDataType"/>
    <element name="renData" type="tmch:renDataType"/>
    <element name="trnData" type="tmch:trnDataType"/>
  </choice>
</complexType>

<!--
<check> response elements.
-->
<complexType name="chkDataType">
  <sequence>
    <element name="cd" type="tmch:checkType"
      maxOccurs="unbounded"/>
  </sequence>
</complexType>

<complexType name="checkType">
  <sequence>
    <element name="id" type="tmch:checkIdType"/>
    <element name="reason" type="tmch:reasonType"
      minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="checkIdType">
  <simpleContent>
    <extension base="mark:idType">
      <attribute name="avail" type="boolean"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>
```



```
<!--
<create> response elements.
-->
<complexType name="creDataType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="crDate" type="dateTime"/>
    <element name="balance" type="tmch:balanceType"/>
  </sequence>
</complexType>

<complexType name="balanceType">
  <sequence>
    <element name="amount" type="tmch:currencyAmountType"/>
    <element name="statusPoints" type="integer"/>
  </sequence>
</complexType>

<complexType name="currencyAmountType">
  <simpleContent>
    <extension base="tmch:amountType">
      <attribute name="currency" type="tmch:currencyCodeType"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>

<simpleType name="amountType">
  <restriction base="decimal">
    <fractionDigits value="2"/>
  </restriction>
</simpleType>

<simpleType name="currencyCodeType">
  <restriction base="string">
    <minLength value="3"/>
    <maxLength value="3"/>
    <pattern value="[A-Z]{3}"/>
  </restriction>
</simpleType>

<!--
<info> response elements.
-->
<complexType name="infDataType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="status" type="tmch:statusType"/>
    <element name="pouStatus" type="tmch:pouStatusType" minOccurs="0"/>
    <element name="smdId" type="mark:idType" minOccurs="0"/>
    <element ref="mark:abstractMark" minOccurs="0"/>
    <element ref="smd:signedMark"
      minOccurs="0"/>
    <element ref="smd:encodedSignedMark"
      minOccurs="0"/>
    <element name="encFile" type="string"
      minOccurs="0"/>
    <element name="document" type="tmch:infDocumentType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="label" type="tmch:infLabelType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="comment" type="tmch:optionalStringType"
      minOccurs="0"/>
    <element name="case" type="tmch:infMarkCaseType"/>
```



```

        minOccurs="0" maxOccurs="unbounded"/>
    <element name="crDate" type="dateTime"
        minOccurs="0"/>
    <element name="upDate" type="dateTime"
        minOccurs="0"/>
    <element name="exDate" type="dateTime"
        minOccurs="0"/>
    <element name="pouExDate" type="dateTime"
        minOccurs="0"/>
    <element name="correctBefore" type="dateTime"
        minOccurs="0"/>
    <element name="authCode" type="normalizedString"
        minOccurs="0"/>
</sequence>
</complexType>

<complexType name="statusType">
    <simpleContent>
        <extension base="normalizedString">
            <attribute name="s" type="tmch:statusValueType"
                use="required"/>
        </extension>
    </simpleContent>
</complexType>

<simpleType name="statusValueType">
    <restriction base="token">
        <enumeration value="new"/>
        <enumeration value="verified"/>
        <enumeration value="incorrect"/>
        <enumeration value="corrected"/>
        <enumeration value="invalid"/>
        <enumeration value="expired"/>
        <enumeration value="deactivated"/>
    </restriction>
</simpleType>

<complexType name="pouStatusType">
    <simpleContent>
        <extension base="normalizedString">
            <attribute name="s" type="tmch:pouStatusValueType"
                use="required"/>
        </extension>
    </simpleContent>
</complexType>

<simpleType name="pouStatusValueType">
    <restriction base="token">
        <enumeration value="notSet"/>
        <enumeration value="valid"/>
        <enumeration value="invalid"/>
        <enumeration value="expired"/>
        <enumeration value="na"/>
        <enumeration value="new"/>
        <enumeration value="incorrect"/>
        <enumeration value="corrected"/>
    </restriction>
</simpleType>

<complexType name="infDocumentType">
    <sequence>
        <element name="id" type="token"/>
        <element name="docType" type="tmch:docClassType"/>
        <element name="fileName" minOccurs="0" type="tmch:optionalStringType"/>
    </sequence>
</complexType>

```




```

    <element name="fileType" type="tmch:fileType"/>
    <element name="status" minOccurs="0" type="tmch:docStatusType"/>
    <element name="comment" minOccurs="0" type="tmch:optionalStringType"/>
  </sequence>
</complexType>

<complexType name="docStatusType">
  <simpleContent>
    <extension base="normalizedString">
      <attribute name="s" type="tmch:docStatusValueType"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>

<simpleType name="docStatusValueType">
  <restriction base="token">
    <enumeration value="new"/>
    <enumeration value="verified"/>
    <enumeration value="invalid"/>
    <enumeration value="expired"/>
  </restriction>
</simpleType>

<complexType name="infLabelType">
  <sequence>
    <element name="aLabel" type="mark:labelType"/>
    <element name="uLabel" type="tmch:requiredStringType"/>
    <element name="smdInclusion" minOccurs="0" type="tmch:notifyType"/>
    <element name="claimsNotify" minOccurs="0" type="tmch:notifyType"/>
  </sequence>
</complexType>

<complexType name="infMarkCaseType">
  <sequence>
    <element name="id" type="tmch:caseIdType"/>
    <element name="udrp" type="tmch:udrpType"
      minOccurs="0"/>
    <element name="court" type="tmch:courtType"
      minOccurs="0"/>
    <element name="status" type="tmch:markStatusType"/>
    <element name="document" type="tmch:infCaseDocumentType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="label" type="tmch:infCaseLabelType"
      minOccurs="0" maxOccurs="unbounded"/>
    <element name="comment" type="tmch:optionalStringType"
      minOccurs="0"/>
    <element name="upDate" type="dateTime"
      minOccurs="0"/>
  </sequence>
</complexType>

<simpleType name="caseIdType">
  <restriction base="token">
    <pattern value="case-\d+"/>
  </restriction>
</simpleType>

<complexType name="markStatusType">
  <simpleContent>
    <extension base="normalizedString">
      <attribute name="s" type="tmch:markStatusValueType"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>

```



```
</simpleContent>
</complexType>

<simpleType name="markStatusValueType">
  <restriction base="token">
    <enumeration value="new"/>
    <enumeration value="verified"/>
    <enumeration value="incorrect"/>
    <enumeration value="corrected"/>
    <enumeration value="invalid"/>
  </restriction>
</simpleType>

<complexType name="infCaseLabelType">
  <sequence>
    <element name="aLabel" type="mark:labelType"/>
    <element name="status" type="tmch:labelStatusType"
      minOccurs="0"/>
  </sequence>
</complexType>

<complexType name="labelStatusType">
  <simpleContent>
    <extension base="normalizedString">
      <attribute name="s" type="tmch:labelStatusValueType"
        use="required"/>
    </extension>
  </simpleContent>
</complexType>

<simpleType name="labelStatusValueType">
  <restriction base="token">
    <enumeration value="new"/>
    <enumeration value="verified"/>
    <enumeration value="incorrect"/>
    <enumeration value="corrected"/>
    <enumeration value="invalid"/>
    <enumeration value="expired"/>
  </restriction>
</simpleType>

<complexType name="infCaseDocumentType">
  <sequence>
    <element name="id" type="token"/>
    <element name="docType" type="tmch:caseDocClassType"/>
    <element name="fileName" minOccurs="0" type="tmch:optionalStringType"/>
    <element name="fileType" type="tmch:fileType"/>
    <element name="status" minOccurs="0" type="tmch:docStatusType"/>
    <element name="comment" minOccurs="0" type="tmch:optionalStringType"/>
  </sequence>
</complexType>

<!--
<renew> response elements.
-->
<complexType name="renDataType">
  <sequence>
    <element name="id" type="mark:idType"/>
    <element name="exDate" type="dateTime"
      minOccurs="0"/>
    <element name="balance" type="tmch:balanceType"/>
  </sequence>
</complexType>
```



```
<complexType name="trnDataType">
  <sequence>
    <element name="newId" type="mark:idType"/>
    <element name="trnDate" type="dateTime"
      minOccurs="0"/>
    <element name="balance" type="tmch:balanceType"/>
  </sequence>
</complexType>

<!--
End of schema.
-->
</schema>
```



3. CREATING A CONNECTION

This API is intended for use in diverse operating environments where transport and security requirements vary greatly. A Transport Layer Security (TLS) mechanism over standard TCP/IP sockets is used. This transport mechanism is based on the reference document RFC 5734 “Extensible Provisioning Protocol (EPP) Transport over TCP”. Further security considerations are resolved by the TLS transport layer and are beyond the scope of this document.

Currently the API uses a self-signed CA certificate. You can download it via the website in case your EPP client requires you to verify the SSL certificate of the EPP server. The API is to be used in synchronous mode. A response to a command must be received by the client before sending another command. The API can be layered over multiple transport protocols.

The API can only be used in a connection-oriented mode. A connection-less mode can be simulated by sending a login/command/logout combination in one set. There is a timeout on the session. If a session has not been used for a period longer than 5 minutes, the system will enforce a logout command and close the connection. Any other requirements are further described in the standard.

The number of simultaneously open connections per trademark agent account is set at two connections. Attempting to open more sessions will be blocked.

XML is case sensitive. Unless stated otherwise, XML specifications and examples provided in this document must be interpreted in the character case presented. Any data unit sent to the server or received from the server contains two sections, a block describing the message length and an XML message.

3.1. First block

A 32-bit header describing the total length of the data unit including the length of the first block. The header information (4 bytes) is to be sent in network order (also known as BigEndian).

3.2. Second block

This block contains the XML API command or response. It is an XML structure that complies with the above XSD description. The XML block has to use a supported character set such as UTF-8 and an XML version of 1.0. Please note that UTF-8 is the RECOMMENDED character encoding for use with this API.

All API commands are enclosed within in `<tmch> ... </tmch>` tags.

After setting up a session the first message will be sent by the server and will contain a ‘hello’ response XML document. Error messages are compliant with the EPP standard. A description of the various messages can be found in appendix 2.



During a session it is always possible to add a cTRID tag. This block will always be mimicked by the server in the trID block enabling the client to track the command and response send by the server. The server will add an extra svTRID block to the trID block uniquely identifying the server response code. This reference can be useful to trace problems and might be asked or referred to in case of technical problems.

3.3. Labels

The following method can be used to verify if a label matches a name of a mark

1. a dot '.' is not allowed in a mark name – all mark names (and labels) with a dot will be rejected.
2. determine which characters that can be represented in an IDN label determine which characters that can be represented in an IDN domain name (refer to IDN 2008 guideline)
3. all other IDN non permissible characters can be omitted or replaced by a hyphen '-'
4. in case the character is the ampersand (&) or the atpersand (@) it can be omitted, replaced by a hyphen '-' or by a transliteration of 'and' or 'at' in the language linked to the jurisdiction or the country where the mark is registered. (see appendix 3)
5. Verify if the label is conform the general domain name rules
 - ⇒ name cannot start with a hyphen ('-')
 - ⇒ name cannot end in a hyphen ('-')
 - ⇒ name with a dash on the 3rd and 4th position is rejected
 - ⇒ name and its ascii representation (a-label) cannot be longer than 63 characters

If using the 5 above steps allow to convert the mark name in a label will be accepted.

Examples Mark name Labels

ab & c	Allowed: ab-c, ab-c, abandc, ab-and-c (if jurisdiction is US), ...
	Not allowed: ab---c (dash on 3rd and 4th position),
	Not allowed: a-and-c (b cannot be omitted), ...
a+b	Allowed: ab, a-b
+abc+	Allowed: abc
	Not allowed: -abc (starts with dash), abc- (ends with a dash), ...

3.3.1. Algorithm

1. We will assume the trademark name TM is according to ICANN specifications
2. Convert all characters to the lower case equivalent
3. create an empty list of lists A
4. for all characters C in the name TM check if the character C can be represented in IDN2008
 - ⇒ yes : create a list L with a single element, the character C
 - ⇒ no: create a list L with 2 elements : an empty string and the character '-'
 - ⇒ if the character is '&' or '@' add additional transliterations to the list L (based on country or jurisdiction – see appendix 3)



- ➔ add the list L to the list of lists A
- 5. create a complete set S of labels by calculation all unique combination's of exactly one choice for all lists L in the list A (this can easily be done by recursion or by straight forward looping over all combination's)
- 6. eliminate all labels from the set S that start or end in a '-', have a '-' in 3rd and 4th position or are longer then 63 characters
- 7. S now contains the list of all possible labels linked to the mark name TM



4. SESSION COMMANDS

Session commands are intended to set up new sessions and to keep the session alive to process various CREATE/UPDATE/READ/DELETE (CRUD) commands.

There are 3 basic session commands

- login
- logout
- hello (to keep the session alive)

4.1. LOGIN

Example of a login command

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:iETF:params:xml:ns:tmch-1.1">
  <command>
    <login>
      <clID>u777</clID>
      <pw>ReplaceWithPass</pw>
    </login>
    <clTRID>client-id-123</clTRID>
  </command>
</tmch>
```

The response in case of a positive result confirms user identification

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:iETF:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <trID>
      <clTRID>ref-to-which-login</clTRID>
      <svTRID>login-1390895916</svTRID>
    </trID>
  </response>
</tmch>
```

After logging in successfully you have an authenticated session. You will be able to retrieve information on all objects in your portfolio/account.

4.2. LOGOUT

The logout command is intended to terminate a session.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:iETF:params:xml:ns:tmch-1.1">
  <command>
    <logout/>
  </command>
</tmch>
```



```
</command>  
</tmch>
```

The result will be the following XML message after which the server will close the connection.

```
<?xml version="1.0" encoding="utf-8"?>  
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">  
  <response>  
    <result code="1000">  
      <msg>Command completed successfully</msg>  
    </result>  
    <msgQ count="76" id="17" />  
    <trID>  
      <svTRID>logout-1390896163</svTRID>  
    </trID>  
  </response>  
</tmch>
```

Notice that is a connection has not been used for some time, the server will send this message to the client (auto-logout) and the connection will be closed.

4.3. HELLO

The hello command can be sent at all moments. It can be used to verify if the connection is still open or to keep the session alive. It can also be used if a connection is established since it does not require an authenticated session.

Command to send

```
<?xml version="1.0" encoding="utf-8"?>  
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">  
  <hello/>  
</tmch>
```

The response is

```
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">  
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">  
  <greeting>  
    <svID>TMCH server v1.1</svID>  
    <svDate>2014-01-28T08:01:16Z</svDate>  
  </greeting>  
</tmch>
```




5. CREATE MARK

With the create command you will be able to create the various mark types. There are 3 different types of marks that can be registered

1. registered trademark
2. court validated mark
3. statute or treaty

5.1. Create registered trademark

The following command can be used to create a registered trademark:

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <create>
      <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
        <trademark>
          <id>0000061234-1</id>
          <markName>Example One</markName>
          <holder entitlement="owner">
            <org>Example Inc.</org>
            <addr>
              <street>123 Example Dr.</street>
              <street>Suite 100</street>
              <city>Reston</city>
              <sp>VA</sp>
              <pc>20190</pc>
              <cc>US</cc>
            </addr>
          </holder>
          <jurisdiction>US</jurisdiction>
          <class>35</class>
          <class>36</class>
          <goodsAndServices>Dirigendas et eiusmodi
            featuring infringo in airfare et cartam servicia.
          </goodsAndServices>
          <regNum>234235</regNum>
          <regDate>2009-08-16T09:00:00.0Z</regDate>
          <exDate>2015-08-16T09:00:00.0Z</exDate>
        </trademark>
      </mark>
      <period unit="y">5</period>
      <document>
        <docType>Other</docType>
        <fileName>C:\\ddafs\\file.png</fileName>
        <fileType>jpg</fileType>
        <fileContent>/9j/4AAQSkZJRgABAQEASABIAAD/2wBDAP////////////////////
        //////////////////////2wBD Af////////////////////
        //////////////////////wAARCAABAEDASIA
        AhEBAxEB/8QAFQABAQAAAAAAAAAAAAAAAAAAP/xAAUEAEAAAAAAAAAAAAAAAAAA
        AAAAAAAAAAAAAAAAAAAAP/EABQRAQAAAAAAAAAAAAAAAAAAD/2gAMAwEAAhEDEQA/
        AKAA/9k=
        </fileContent>
      </document>
      <label>
        <aLabel>exampleone</aLabel>
      </label>
    </create>
  </command>
</tmch>
```



```

        <smdInclusion enable="1"/>
        <claimsNotify enable="1"/>
    </label>
    <label>
        <aLabel>example-one</aLabel>
        <smdInclusion enable="0"/>
        <claimsNotify enable="1"/>
    </label>
</create>
</command>
</tmch>

```

Remarks

Trademark	Means registered trademark
id	Unique reference that should start with the agents uid formatted in 6 digits left padded with zeros and ending in -1
markName	Name of the mark
Holder entitlement	Owner/Licensee/Assignee
sp	State/Province
ORG	Name of the organization that is the trademark holder
ADDR	Address consisting out of :
Street	Street
City	City
sp	State
pc	Postal code
cc	Country code (for the complete list see appendix 1)
class	Nice classification number (1-45 or 46 for N/A)
Jurisdiction	for the complete list see appendix 1)
goodsAndServices	Full copy of the complete description of goods and services information on the registration form
regNum	The registration number of the trademark
regDate	The date of registration of the mark (YYYY-MM-DD)
exDate	The date on which a trademark expires (YYYY-MM-DD)
Period unit	The term of registration for a trademark record (1, 3 or 5 years)
docType	Licensee declaration/ Trademark - Assignee declaration/ Trademark – Other/ Declaration of Proof of use including one single sample/ Proof of use - Other
fileContent	Content of the file, base64 encoded
aLabel	ASCII representation of the label linked to the mark
SmdInclusion	enable Flag for sunrise services: the label will be included in the SMD file to be used in sunrise
claimsNotify	Flag for the claims services: registrants will be made aware of mark existence and you will receive notification on registration during the initial period of general availability



5.2. Create court validated mark

The following command can be used to create a court validated mark

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <create>
      <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
        <court>
          <id>0000061234-1</id>
          <markName>Example One</markName>
          <holder entitlement="owner">
            <org>Example Inc.</org>
            <addr>
              <street>123 Example Dr.</street>
              <street>Suite 100</street>
              <city>Reston</city>
              <sp>VA</sp>
              <pc>20190</pc>
              <cc>US</cc>
            </addr>
          </holder>
          <goodsAndServices>Dirigendas et eiusmodi
            featuring infringo in airfare et cartam servicia.
          </goodsAndServices>
          <refNum>234235</refNum>
          <proDate>2009-08-16T09:00:00.0Z</proDate>
          <cc>US</cc>
          <courtName>P.R. supreme court</courtName>
        </court>
      </mark>
      <period unit="y">5</period>
      <document>
        <docType>Other</docType>
        <fileName>C:\\ddafs\\file.png</fileName>
        <fileType>jpg</fileType>
        <fileContent>YnJvbAo=</fileContent>
      </document>
      <label>
        <aLabel>exampleone</aLabel>
        <smdInclusion enable="1"/>
        <claimsNotify enable="1"/>
      </label>
      <label>
        <aLabel>example-one</aLabel>
        <smdInclusion enable="0"/>
        <claimsNotify enable="1"/>
      </label>
    </create>
  </command>
</tmch>
```

Remarks

court	This means that you want to enter a court validated mark
id	Unique reference that should start with the agents uid formatted in 6 digits left padded with zeros and ending in -1
markName	Name of the mark as on the registration form
Holder entitlement	Owner/Licensee/Assignee



org	Name of the organization that is the trademark holder
addr	Address consisting out of :
street	Street
city	City
sp	State/Province
pc	Postal code
cc	Country code (for the complete list see appendix 1)
goodsAndServices	Copy of the complete description of goods and services
refNum	Reference number of the court order
proDate	The date of protection of the mark (YYYY-MM-DD)
CourtName	Name of the court
Period unit	The term of registration for a trademark record (1, 3 or 5 years)
docType	Trademark - Licensee declaration/ Trademark - Assignee declaration/Trademark - Copy of the court order
fileContent	Content of the file, base64 encoded
aLabel	ASCII representation of the label linked to the mark
SmdInclusion	Flag for sunrise services : the label will be included in the SMD file to be used in sunrise
claimsNotify	Flag for the claims services : registrants will be made aware of mark existence and you will receive notification on registration during the initial period of general availability

5.3. Create statute or treaty

The following command can be used to create a statute or treaty mark

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <create>
      <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
        <treatyOrStatute>
          <id>00000712423-1</id>
          <markName>Example One</markName>
          <holder entitlement="owner">
            <org>Example Inc.</org>
            <addr>
              <street>123 Example Dr.</street>
              <street>Suite 100</street>
              <city>Reston</city>
              <sp>VA</sp>
              <pc>20190</pc>
              <cc>US</cc>
            </addr>
          </holder>
          <protection ruling="US">
            <cc>US</cc>
          </protection>
          <goodsAndServices>Dirigendas et eiusmodi
            featuring infringo in airfare et cartam servicia.
          </goodsAndServices>
          <refNum>234235</refNum>
          <proDate>2009-08-16T09:00:00.0Z</proDate>
        </mark>
      </create>
    </command>
  </tmch>

```



```

        <title>My Mark Title</title>
        <execDate>2010-01-05T09:00:00.0Z</execDate>
    </treatyOrStatute>
</mark>
<period unit="y">5</period>
<document>
    <docType>Other</docType>
    <fileName>C:\\ddafs\\file.png</fileName>
    <fileType>jpg</fileType>
<fileContent>/9j/4AAQSkZJRgABAQEASABIAAD/2wBDAP////////////////////////
/////////////////////////2wBDaf////////////////////////
/////////////////////////wAARCAABAEDASIA
AhEBAxEB/8QAFQABAQAAAAAAAAAAAAAAAAAAP/xAAUEAEAAAAAAAAAAAAAAAAAAA/8QAFaEB
AAAAAAAAAAAAAAAAAAAAAAAAAP/EABQRAQAAAAAAAAAAAAAAAAAAD/2gAMAwEAAhEDEQA/AKAA/9k=
</fileContent>
    </document>
    <label>
        <aLabel>exampleone</aLabel>
        <smdInclusion enable="1"/>
        <claimsNotify enable="1"/>
    </label>
    <label>
        <aLabel>example-one</aLabel>
        <smdInclusion enable="0"/>
        <claimsNotify enable="1"/>
    </label>
</create>
</command>
</tmch>

```

treatyorstatue	This means that you want to enter a mark protected under statue of treaty
id	Unique reference that should start with the agents uid formatted in 6 digits left padded with zeros and ending in -1
markName	Name of the mark as on the registration form
Holder entitlement	Owner/Licensee/Assignee
org	Name of the organization that is the trademark holder
addr	Address consisting out of :
street	Street
city	City
sp	State/Province
pc	Postal code
cc	Country code (for the complete list see appendix 1)
goodsAndServices	Copy of the complete description of goods and services
refNum	Reference number of the mark
proDate	The date of protection of the mark (YYYY-MM-DD)
Title	Title of the statute or treaty
Period unit	The term of registration for a trademark record (1, 3 or 5 years)
execdate	Date of execution of the statue or treaty (YYYY-MM-DD)
docType	Trademark - Licensee declaration/ Trademark - Assignee declaration/Trademark – Other
fileContent	Content of the file, base64 encoded
aLabel	ASCII representation of the label linked to the mark



SmdInclusion	enable Flag for sunrise services : the label will be included in the SMD file to be used in sunrise
claimsNotify	Flag for the claims services : registrants will be made aware of mark existence and you will receive notification on registration during the initial period of general availability

5.3.1. *Create mark response*

When a mark is successfully created the following XML messages will be returned by the server

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <resData>
      <creData>
        <id>00000100010001390896163-1</id>
        <crDate>2014-01-28T09:02:43.7Z</crDate>
        <balance>
          <amount currency="USD">57830.2500</amount>
          <statusPoints>1407</statusPoints>
        </balance>
      </creData>
    </resData>
    <trID>
      <svTRID>create-1390896163-5266</svTRID>
    </trID>
  </response>
</tmch>
```

Remarks

If the mark is created successfully the resulting response includes

1. The number of messages still pending
2. A balance section that contains the remaining currency and your status points



6. INFO MARK

The info linked to a specific mark can be retrieved using the info command. Depending on the request and the status of the mark the info can be formatted in various different ways.

1. Mimicking the create command
2. In SMD format
3. In base64 encoded SMD format
4. In full SMD compatibility format

6.1. Info mark command

The plain version of the info command will mimic the create command

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <info>
      <id>00000712423-1</id>
    </info>
  </command>
</tmch>
```

The result is a complete set of info linked to a mark with the given handle. The following example relates to a registered trademark.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <resData>
      <infData>
        <id>00071234-992</id>
        <status s="new" />
        <pouStatus s="notSet" />
        <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
          <court>
            <id>00071234-992</id>
            <markName>My name</markName>
            <holder entitlement="assignee">
              <name>Mr. Rep</name>
              <org>Owner INC</org>
            <addr>
              <street>Jacobs Street</street>
              <city>Bruges</city>
              <sp>VA</sp>
              <pc>8100</pc>
              <cc>BE</cc>
            </addr>
            <voice x="1234">+1.32235523</voice>
            <fax>+1.32235523</fax>
            <email>j@bla</email>
```



```

    </holder>
    <goodsAndServices>BX-1421312-IRUNF-134</goodsAndServices>
    <refNum>12345678</refNum>
    <proDate>2009-08-16T00:00:00Z</proDate>
    <cc>BE</cc>
    <courtName>Court of Brussels</courtName>
  </court>
</mark>
<label>
  <aLabel>my-name</aLabel>
  <uLabel>my-name</uLabel>
  <smdInclusion enable="0" />
  <claimsNotify enable="0" />
</label>
<label>
  <aLabel>myname</aLabel>
  <uLabel>myname</uLabel>
  <smdInclusion enable="0" />
  <claimsNotify enable="0" />
</label>
<crDate>2013-03-22T10:42:06.0Z</crDate>
<upDate>2013-03-19T00:32:45.9Z</upDate>
<exDate>2013-03-19T00:32:45.9Z</exDate>
</infData>
</resData>
<trID>
  <svTRID>poll-1390900577</svTRID>
</trID>
</response>
</tmch>

```

Please note that when the info command returns a successful result, the response includes 3 blocks

1. the result block showing the command is executed successfully
2. the msgQ block showing the number of pending messages if there are any (if not the block will not be added)
3. the resData block showing the mark info

6.2. Info mark smd

If the mark has been approved you will be able to download the SMD file linked to the mark using the following command

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <info type="smd">
      <id>000009812423-1</id>
    </info>
  </command>
</tmch>

```

The result is an encapsulated SMD file

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">

```




```

aWthdC5kay9vY2VzLmNybdAfBgNVHSMEGDAWgBQcmAlHGkw4uRDFBclb8fROgGrMfjAdBgNVHQ4E
FgQUR4fjxTv2jdk9Ztak8CBikXRbWjwwCQYDVR0TBAlwADAZBgkqhkiG9n0HQQAEDDAKGWRWNY4x
AwIDqDANBgkqhkiG9w0BAQUFAAOBQBUBvfdxcBIo8vsxGvtF22WkiWfSkcWj1UhKbGIjcpPfWqS
nZ5iDy5T9iIGY/huT/7SgnJbJnpN6EEa6UqXUBUzdth5OP7qBQkvKnOkIyuGjftcqV8Xja4GwZos
6F6NzqACReC2kwuN44zrzjz3fj7dA26QGpimCJWzV5zRyW3Ng==</ds:X509Certificate><ds:X509Certifi-
cate>MIEXTC8agAwIBAgIEQDYX/DANBgkqhkiG9w0BAQUFADA/MQswCQYDVQQGEwJESzEMMAoGA1UE
ChMDVERDMSIwIAYDVQQDExlUREMgT0NFUyBTeXN0ZW10ZXN0IENBIElJMB4XDTA0MDIyMDEzNTE0
OV0XDTM3MDYyMDE0MjE0OVowPzELMAkGA1UEBhMCRESzDDAKBgNVBAoTA1REQzEiMCAgA1UEAxMZ
VERDIE9DRVMgU3lzdGVtdGVzdCBDQSBjSTCBnZANBgkqhkiG9w0BAQEFAAOBjQAwYkCgYEArawA
NI56sljDsnosDU+Mp4r+rKFys9c5qy8jWZyA+7PYfs4+IZcFxbNuHi8aAcbsFOUJF0PGpNgPetN
c+XAK7p16iawNTyPmkHm2VoInNfWEj/wGmtb4rKDT2a7auGk76q+Xdqno4PRO8e7AKEHw7pN3k
iHmZCI48PTRpRx8CAwEAAOCAMQwgGjgMA8GA1UdEwEB/wQFMAMBAf8wDgYDVR0PAQH/BAQDAgEG
MIIBAwYDVR0gBIH7MIH4MIH1BgkqAQEBAQEBAQEwgcwLwYIKwYBBQUHAgEWI2h0dHA6Ly93d3cu
Y2VydG1maWthdC5kay9yZXVvc210b3J5MIGzBggrBgEFBQcCAjCBpJAKFgNUREMwAwIBARqBl1RE
QyBUZXN0IENlcnRpZmlrYXRlcjBmcmEgZGVubmUgQ0EgdWRzdGVkZXN0Mgk5kZlXIG1EIDeUMS4x
LjEuMS4xLjEuMS4xLjEuIFREYyBUZXN0IENlcnRpZmlrYXRlcjBmcmEgZGVubmUgQ0EgdWRzdGVk
c3VlZCB1bmRlcjBPSUQgMS4xLjEuMS4xLjEuMS4xLjEuMS4wEQQYJYIZIAyB4QgEBBAQDAgAHHIGW
BgNVHR8EgY4wgYswVqBUoFKkUDBOMQswCQYDVQQGEwJESzEMMAoGA1UEChMDVERDMSIwIAYDVQQD
ExlUREMgT0NFUyBTeXN0ZW10ZXN0IENBIElJMQ0wCwYDVRQDEwRDUKwMDGGL6AthitodHRwOi8v
dGVzdC5jcmwub2Nlcj5jZXJ0aWZpa2F0LmRrL29jZXMuY3J5MCA1UEEAQMCKADzIwMDDQwMjIw
MTM1MTQ5W0EPMjAzNzA2MjAxNDIxNDIAMB8GA1UdIwQYMBaAFByYCUcaTDi5EMUEKVvx9E6Aasx+
MB0GA1UdDgQWBQcmAlHGkw4uRDFBclb8fROgGrMfjAdBgkqhkiG9n0HQQAEEEDAOGwhWNi4w0jQu
MAMCBJAwdQYJKoZIhvcNAQEFBQADgYEApyoAjiKq6WK5XaKWUpVskutzohv1VcCke/3JeUVtmB+b
yexJMC171s4RHogcbufcI2ASVWwu84i45MaKg/nx0qoqMyY19/W2wbQFEsdxUCnLa9e9t1Wj0xS/
AaKeUhk2MB0qv+hMdc71j0qc5JN7T2Ba6ZRIY5uXk03IGZ3XUsw=</ds:X509Certificate></ds:X509Data></
ds:KeyInfo></ds:Signature></smd:signedMark>
</infData>
</resData>
<trID>
  <svTRID>info-smd-1390989266</svTRID>
</trID>
</response>
</tmch>

```

6.3. Info mark encoded smd

Alternative format is an encapsulated base64 encode SMD file

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <info type="enc">
      <id>00000715623-1</id>
    </info>
  </command>
</tmch>

```

The result is a complete set of info linked the mark with the given handle

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <resData>
      <infData>
        <id>0000014361387359953082-1</id>
        <status s="verified" />
        <smdId>000000000023234-2</smdId>
        <smd:encodedSignedMark xmlns:smd="urn:ietf:params:xml:ns:smd-1.0">PD94bWw-
gdmVyc2lvdj0iMS4wIiBlbmNvZGluz0iVVRGLTgiPz4KPHNTZDpzaWduZWRYXJrLHht
bG5zOnNtZD0idXJuOmlldGY6cGFyYXl1zOnhtbDpuc2pzaWduZWRYXJrLTEuMCIGaWQ9I18yYTRj

```




WnBhMkYwTG1ScKwyOWpjM0F2YzNSaGRIVnoKTU1JQk53WURWUjBnQklJQkxqQ0NBU293Z2dFbUJn
 b3FnVnkCS1FFQkFRUNNSU1CRmpBdkJnZ3JCZ0VGQ1FjQ0FSWwphSFwY0RvdgpMM2QzZHk1a1pY
 SjbHv1pwYTJGMEExtUnJMM0psY0c5emFYUnZjBmt3Z2VJR0Ndc0dBUVVGQndJQ01JSFZFNQW9XQTFs
 RVF6QURBZ0VCCkdvSEdSbT15SudGdWRtVnVaR1ZzYzJVZ11XWwdZM1Z5ZEdsbWFXdGhkr1YwSudm
 bWJHUmXjaUJQUTBWVE1IWnBiR3ZsY213Z1ExQ1QKSUc5bk1FOURSVk1nUTFBc01HUmxjaUJyWvc0
 Z2FHVnVkr1Z6Sudaev1TQjNkM2N1WTJweWRHbG1hV3RoZEM1a2F5OX1aWEJ2YzJsMApiM0o1TG1C
 Q1pXM21jbxNzSudGME1GUKvREUJsWm5SbGNpQjJhV3hyNVhKbGJtVWdhR0Z5SudWME1HSmxaM0xt
 Ym5ObGRDQmhibk4yC11YSWdhV1owTG1Cd2NtOW1aWE56YVc5dVpXeHNaU0J3WVhKMFpYSXVnQmdH
 Q1dDR1NBR0crRU1CRFFRTEZnbFFaWEp6YjI1WFpXSXcKSUFZRFZSMFJCQmt3RjRFVmMzVndjRz15
 ZEVCalpYSjBhV1pwYTJGMEExtUnJNSUdYQmdOVkhSOEVnWTh3Z113d1Y2Q1ZvRk9rVVRUCUApNuxN3
 Q1FZRFZRUUdFd0pFU3pFTU1Bb0dBMVVFQ2hNRfZfUkRNU013SUFZRFZRUURFeGxVUkVNZ1QwTkZV
 eUJUzVhOMFpXMTBaWE4wCk1FTkJRwXKTVE0d0RBWURWUvFERXdWRfVrd31PVEF4b0MrZ0xZWXJh
 SF1wY0RvdkwzUmXjM1F1WTKc0xtOWpaWE11WTJweWRHbG0KYVd0aGRDNWtheT12WTJWekxtTn1i
 REFMqmdOVkhTtUVHREFXZ0JRY21BbEhHa3c0dVJERkJDbGI4Z1JPZ0dyTWZqQWRCZ05SFEORQpG
 Z1FVUjRmanhUdjJqZGs5WnRhazhDQmlrWfJiV2p3d0NRWURWUjBUQkFJd0FEQVpCZ2txaGtpRzlu
 MEhRUUFRFRERS0d3U1d0eTR4CkF3SURxREFOQmdrcWhraUc5dzBCQVFRkFBT0JnUUJvdmZkeGNC
 SW84dn4R3Z0RjIyV2tpV0Zza2NXamxVaEtiR015amNwcGZxcVMKblo1aUR5NVQ5aU1HWS9odVQv
 N1Nnbkpiam5wTjZFRWE2VXFYVWJVemR0aDVPUddxQ1FrdktuT2tJeXVHamZ0Y3FWOHqYTRhd1pv
 cwo2RjZ0enFBWENSZUMya3d1dU40NHpyanozZmo3ZEEyN1FHcGltQ0pXenY1e1J5VzNOZz09PC9k
 czpYNTA5Q2VydG1maWnhdGU+PGRz01g1MD1DZXJ0aWZpY2F0ZT5NSU1FWFRDQ0E4YwdBd01CQWdJ
 RVFEWVgVREFQmdrcWhraUc5dzBCQVFRkFEQS9NuxN3Q1FZRFZRUUdFd0pFU3pFTU1Bb0dBMVVF
 CkNoTURWRVJETVnJd01BWURWUvFERXhsVvJFTWdUME5GVX1CVGVYtjBaVzEwW1hOMELFTkJRwXk
 TUI0WERUQTBNRE15TURFek5URTAkt1ZvWERUTNNRF15TURFME1qRTBPVm93UHpfTE1Ba0dBMVVF
 QmhNq1JfC3hEREFLQmdOVkjbB1RBMVJFUXpFaU1DQudBMVVFQXhNWgpWRVJESUU5RFJWtWdVM2x6
 ZEdWdGRHvnpkQ0JEUvNCS1NUQ0JuekFOQmdrcWhraUc5dzBCQVFRkFBT0JqUUF3Z11rQ2dZRUfY
 YXdBck5JNTZzbGpEc25vc0RVK01wNHirUktGeXm5YzVxeThqV1p5QSS3UF1GczQrSVpjRnhuYk51
 SGk4YUFjY1NGT1VKRjBQR3BOZ1BFde4KYytYQUS3cDE2aWF3T1RzCe1rSG0yVm9Jbk5md1dFai93
 R210YjRyS0RUMme3YXVhazc2cStYZHFubm80UFJPOGU3QUtFSHC3cE4zawppSG1aQ0k0OFBUUNBS
 eDhDQXdFQkFhT0NBbVf3Z2dKZ01BOEdBMVvkRXdfQi93UUZNQU1CQWY4d0RnWURWUjBQVVFIL0JB
 UURBZ0VHCK1JSUJbd11EV1IwZ0JSDdNSUG0TU1IMUJna3BBUUVQVFFQkFRRXdnZWN3THdZSUT3
 WUJCUVVIQWdF0kyADbkSEe2THk5M2QzY3UKWTJweWRHbG1hV3RoZEM1a2F5OX1aWEJ2YzJsMGIz
 SjVNSUd6QmdnckJnRUZCUWNDQWpDQnBqQUtGZ05VUkVNd0F3SUJBUncbDFSRQpReUJVVW1hOMELF
 TmxjblJwWmlsc11YUmXjaUJtY21FZ1pHVnVibVvNUTBFZ2RXUnpkR1ZrW1hNZ2RXNwtaWElnVDBs
 RU1ERXVNUzR4CkxqRXVNUzR4TGPfdu1TNHhMakV1SUZSRVF5Q1VaWE4wSUVObGNuUnBabWxqWVhS
 bGN5Qmljbt10SUhSb2FYTWdRMEVnWVhKbElHbHoKyZnWbFpDQjFibVJsY21CUFNvUwdNUzR4TGPf
 du1TNHhMakV1TVM0eExqRXVNUzR3RVFZS11Jwk1BWWI0UWdFQkJBUURBZ0FITU1HVvpCZ05WSFI4
 RWdZNHdnWXN3VnFCVW9GS2tVREJPTVFzd0NRWURWUvFHRXdKRVN6RU1NQW9HQTFVRUNoTURWRVJE
 TVNjD01BWURWUvFEckV4bFVSRU1nVDBOR1V5Q1R1WE4wW1cxMFpYtjBJRU5CSUVsSk1RMHdDd11E
 V1FRREV3UkrVa3d4TURHZ0w2QXRoaXRvZehSd09pOHYKZEdWemRDNWpjbx1YjJObGN5NwpaWEow
 YVdacGEYrjBmBVjYTDI5a1pYTXVZM0pzTUNzR0ExVWRfQVFRtUNLQUR6SXdnRFF3TWPJdwpNVE0x
 TVRRNVdvRVBNakF6TnpBMk1qQXhORE14TkrsYU1COEdBMVvkSXdrWU1CYUFGQnlZQ1VjYVREaTVF
 TVVFS1Z2ed1FNkFhc3grCk1CMEdBMVvkRGdRV0JCUWNTQWxIR2t3NHVSREZCQ2xiOGZST2dhck1m
 akFkQmdrcWhraUc5bjBIUVFBruVEQU9Hd2hXtmk0d09QUXUKTUFNQ0JKQXdeUV1KS29aSWH2Y05B
 UUVGQ1FBRGdZRUfweW9Bam1LcTZXszVYUUtXVXBWc2t1dHpvaHYxVmNda2UvM0p1VvZ0bUIrYgp5
 ZXhKTUMxNzFzNFJib3FjYnVmY0kyQVNWV3d1ODRpNDVNYUtnL254b3Fvak15WTE5L1cyd2JRRkVv
 c3hVQ25MYT11OXRSV2oweFMvCkFhS2VvaGsyTUJpCXYraE1kYzcxak9xYzVKTjdUMkjhN1pSSVkl
 dVhrTzNJR1ozWfVzd08L2Rz01g1MD1DZXJ0aWZpY2F0ZT48L2Rz01g1MD1EYXRhPjwvZHM6S2V5
 SW5mbz48L2Rz01NpZ25hdHVyZT48L3NtZDpzaWduZWRRNYXJrPg==

```

</smd:encodedSignedMark>
  </infData>
  </resData>
  <trID>
    <svTRID>info-smd-1390988225</svTRID>
  </trID>
</response>
</tmch>
    
```




```

YXdBCk5JNTZzbGpEc25vc0RVK01wNHlRUKtGeXM5YzVxeThqV1p5QSS3UF1GczQrSVpjRnhuYk51
SGk4YUfjY1NGT1VKRjBQR3BOZ1BFdE4KYytYQUs3cDE2aWF3T1RZcE1rSG0yVm9Jbk5md1dFai93
R210YjRyS0RUMMe3YXVHacz2cStYzHFubm80UFJPOGU3QUtFShc3cE4zawppSG1aQ0k00FBUUNBS
eDhDQXdFQUFhT0NBbVf3Z2dKZ01BOEdBMVvkRXdfQi93UUZNQU1CQWY4d0RnWURWUjBQVVFIL0JB
UURBZ0VHck1JSUJbd1lEV1IwZ0JJSdDnSUG0TU1IMUJna3BBUUVQVFFQkFRRXdnZWN3THdZSUT3
WUJCUVVIQWdFV0kyadbKSE2THk5M2QzY3UKWTJWeWRHbG1hV3RoZEM1a2F5OX1aWEJ2YzJSMGIz
SjVNSUd6QmdnckJnRUZCUWNDQWpDQnBqQUtGZ05VUkVNd0F3SUJBUnFCbDFSRQpReUJjVWlhOMElF
TmxjblJwWmlsc1lYUmxjaUJtY21FZlphVnVibVVnUTBFZ2RXUnpkR1ZrWlhNZ2RXNWtaWElnVDBs
RU1ERXVNUzR4CkxqRXVNUzR4TGpFdU1TNHhMakV1SUZSRVF5Q1VaWE4wSUVObGNuUnBabWxqWVhS
bGN5QmljbTl0SUhSb2FYTWdRMEVnWVhKbElHbHoKYzNWbFpDQjFibVJsY21CUFNvUWdNUzR4TGpF
dU1TNHhMakV1TVM0eExqRXVNUzR3RVFZS1lJWk1BWWI0UWdFQkBUURBZ0FITU1HVWpCZ05WSFI4
RwdZNHdnWXN3VnFCVW9GS2tVREJPTVFzd0NRWURWUjVFRXdkRvN6RU1NQW9HQTFVRUNoTURWRVJE
TVNJd01BWURWUjVFEckV4bFVSRU1nVDBORlV5Q1RlWE4wWlcmMFpYTjBjRU5CSUVsSk1RMHdDd1lE
V1FRREV3UkRva3d4TURHZw2QXRoaXRvZehSd09pOHYKZEdWemRDNWpjbXdlYjJObGN5NWpaWEow
YVdacGEyRjBmVjYTDI5alpYTXVZM0pzTUNzR0ExVWRFQVFRtTUNLQUR6SXdnRFF3TWpJdWpNVE0x
TVRRNVdvRVBNakF6TnpBMk1qQXhOREl4TkrSYU1COEdBMVvkSXdrWU1CYUFGQnlZQ1VjYVREaTVF
TVVFS1Z2eDlFNkFhc3grCk1CMEdBMVvkRGdRV0JCUWNTQWxIR2t3NHVSREZCQ2xiOGZST2dHck1m
akFkQmdrcWhraUc5bjBIUVFBRUVEQU9Hd2hXTmk0d09qUXUKTUFNQ0JKQXdeUUV1KS29aSWH2Y05B
UUVGQ1FBRGdZRUfweW9BamlLcTZXSzVYUUtXVXBWc2t1dHpvaHYxVmNda2UvM0plVVZ0bUIrYg5
ZXhKTUMxNzFzNFJib3FjYnVmY0kyQVNWV3d1ODRpNDVNYUtnL254b3Fvak15WTE5L1cyd2JRRkVk
c3hVQ25MYTl1OXRsV2oweFMvCkFhS2VvaGsyTUJPCyYraElkYzcxak9xYzVKTjdUMk1pSSVkl
dVhrTzNJR1ozWVfVzd08L2RzOlglMD1DZXJ0aWZpY2F0ZT48L2RzOlglMD1EYXRhPjwvZHM6S2V5
SW5mbz48L2RzOlNpZ25hdHVyZT48L3NtZDpzaWduZWRRNYXJrPg==
-----END ENCODED SMD-----
</>
</infData>
</resData>
<trID>
<svTRID>info-smd-1390989415</svTRID>
</trID>
</response>
</tmch>

```

6.5. Info court validated mark

The result is a set of info linked to a court validated mark with the given handle

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <resData>
      <infData>
        <id>00071234-992</id>
        <status s="new" />
        <pouStatus s="notSet" />
        <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
          <court>
            <id>00071234-992</id>
            <markName>My name</markName>
            <holder entitlement="assignee">
              <name>Mr. Rep</name>
              <org>Owner INC</org>
              <addr>
                <street>Jacobs Street</street>
                <city>Bruges</city>
                <sp>VA</sp>
                <pc>8100</pc>
                <cc>BE</cc>
              </addr>
            </holder>
          </court>
        </mark>
      </infData>
    </resData>
  </response>
</tmch>

```



```

    <voice x="1234">+1.32235523</voice>
    <fax>+1.32235523</fax>
    <email>j@bla</email>
  </holder>
  <goodsAndServices>BX-1421312-IRUNF-134</goodsAndServices>
  <refNum>12345678</refNum>
  <proDate>2009-08-16T00:00:00Z</proDate>
  <cc>BE</cc>
  <courtName>Court of Brussels</courtName>
</court>
</mark>
<label>
  <aLabel>my-name</aLabel>
  <uLabel>my-name</uLabel>
  <smdInclusion enable="0" />
  <claimsNotify enable="0" />
</label>
<label>
  <aLabel>myname</aLabel>
  <uLabel>myname</uLabel>
  <smdInclusion enable="0" />
  <claimsNotify enable="0" />
</label>
<crDate>2013-03-22T10:42:06.0Z</crDate>
<upDate>2013-03-19T00:32:45.9Z</upDate>
<exDate>2013-03-19T00:32:45.9Z</exDate>
</infData>
</resData>
<trID>
  <svTRID>poll-1390988544</svTRID>
</trID>
</response>
</tmch>

```

6.6. Info statute or treaty

The result is a set of info linked to a statute or treaty mark with the given handle

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <resData>
      <infData>
        <id>00000113678687551367868755-1</id>
        <status s="new" />
        <pouStatus s="na" />
        <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
          <treatyOrStatute>
            <id>00000113678687551367868755-1</id>
            <markName>treaty</markName>
            <holder entitlement="owner">
              <name>jan jansen</name>
              <org>CHIP</org>
              <addr>
                <street>dvv 37</street>
                <city>Leuven</city>
                <sp></sp>
                <pc>3000</pc>
                <cc>BE</cc>
              </addr>
            </holder>
          </treatyOrStatute>
        </mark>
      </infData>
    </resData>
  </response>
</tmch>

```




```
<voice>+1.123</voice>
<fax>+2.457</fax>
<email>jan@j.abc</email>
</holder>
<protection>
  <cc>US</cc>
  <ruling>AF</ruling>
  <ruling>AX</ruling>
  <ruling>AL</ruling>
</protection>
<goodsAndServices>n.a.</goodsAndServices>
<refNum>n.a.</refNum>
<proDate>2000-01-01T00:00:00Z</proDate>
<title>n.a.</title>
<execDate>2000-01-01T00:00:00Z</execDate>
</treatyOrStatute>
</mark>
<label>
  <aLabel>treaty</aLabel>
  <uLabel>treaty</uLabel>
  <smdInclusion enable="1" />
  <claimsNotify enable="1" />
</label>
<crDate>2013-05-06T21:32:35.6Z</crDate>
<upDate>2013-05-06T21:32:35.6Z</upDate>
<exDate>2014-05-06T19:32:35Z</exDate>
</infData>
</resData>
<trID>
  <svTRID>poll-1390988640</svTRID>
</trID>
</response>
</tmch>
```



7. UPDATE MARK

To update a mark the update command can be used. The update command can contain three different sections

1. add : this allows for adding labels, documents to the mark
2. rem : this allows to remove labels from the mark
3. chg : this block allows you to change the contents of the mark block

7.1. Update mark settings

The following command adds a document and a label, removes a label and changes the type of the mark from trademark to a treatyOrStatute type.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <update>
      <id>000712423-2</id>
      <add>
        <document>
          <docType>tmOther</docType>
          <fileName>C:\\ddafs\\file2.png</fileName>
          <fileType>jpg</fileType>
          <fileContent>YnJvbAo=</fileContent>
        </document>
        <label>
          <aLabel>my-name</aLabel>
          <smdInclusion enable="0"/>
          <claimsNotify enable="1"/>
        </label>
      </add>
      <rem>
        <label>
          <aLabel>my-name</aLabel>
          <smdInclusion enable="1"/>
          <claimsNotify enable="1"/>
        </label>
      </rem>
      <chg>
        <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
          <treatyOrStatute>
            <id>000712423-2</id>
            <markName>Example One</markName>
            <holder entitlement="owner">
              <org>Example Inc.</org>
              <addr>
                <street>123 Example Dr.</street>
                <street>Suite 100</street>
                <city>Reston</city>
                <sp>VA</sp>
                <pc>20190</pc>
                <cc>US</cc>
              </addr>
            </holder>
            <protection>
              <cc>US</cc>
              <region>Puerto Rico</region>
              <ruling>US</ruling>
              <ruling>PR</ruling>
            </protection>
          </treatyOrStatute>
        </mark>
      </chg>
    </update>
  </command>
</tmch>
```



```

        </protection>
        <goodsAndServices>Dirigendas et eiusmodi
            featuring infringo in airfare et cartam servicia.
        </goodsAndServices>
        <refNum>234235</refNum>
        <proDate>2009-08-16T09:00:00.0Z</proDate>
        <title>My Mark Title</title>
        <execDate>2010-01-05T09:00:00.0Z</execDate>
    </treatyOrStatute>
</mark>
</chg>
</update>
</command>
</tmch>

```

7.2. Adding a label

Alternative is a very small update adding just one label. The following provides you with an example.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <update>
      <id>0000011363778801363778080-1</id>
      <add>
        <label>
          <aLabel>exampleone</aLabel>
          <smdInclusion enable="0"/>
          <claimsNotify enable="1"/>
        </label>
      </add>
    </update>
  </command>
</tmch>

```

Response to an update command if update succeeded

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <trID>
      <svTRID>update-1391086383-5308</svTRID>
    </trID>
  </response>
</tmch>

```

7.3. Changing label flags

Similar to adding labels, the flags on the labels can be toggled using the <chg> block.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <update>
      <id>0000011363778801363778080-1</id>
      <chg>
        <label>
          <aLabel>exampleone</aLabel>

```



```
<smdInclusion enable="0"/>  
<claimsNotify enable="1"/>  
</label>  
</chg>  
</update>  
</command>  
</tmch>
```

In a similar fashion extra documents can be added.

Please note that documents can never be removed.



8. UDRP OR COURT CASE

To add a court case or an udrp case to an existing mark, the mark has to be in status VERIFIED. If not, you will not be able to add a case block to the mark.

The case handle starts with the keyword 'case' followed by a dash ('-') followed by six digits containing the uid of the agent (left padded with '0') followed by a set of digits (maximum length 63 characters).

8.1. Adding a udrp case

The update command can be used to add an udrp case to an existing mark using the add block.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <update>
      <id>000001132-1</id>
      <add>
        <case>
          <id>case-0000012346698999999</id>
          <udrp>
            <caseNo>987654321</caseNo>
            <udrpProvider>National Arbitration Forum</udrpProvider>
            <caseLang>Spanish</caseLang>
          </udrp>
          <document>
            <docType>courtCaseDocument</docType>
            <fileName>02-2013-TMCHdefect1.jpg</fileName>
            <fileType>jpg</fileType>
            <fileContent>YnJvbAo=</fileContent>
          </document>
          <label>
            <aLabel>a</aLabel>
          </label>
          <label>
            <aLabel>b</aLabel>
          </label>
        </case>
      </add>
    </update>
  </command>
</tmch>
```

8.2. Adding a court case

The update command can be used to add a court case to an existing mark using the add block.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <update>
      <id>000001132-1</id>
      <add>
        <case>
          <id>case-00000123466989979999</id>
          <court>
            <refNum>987654321</refNum>
          </court>
        </case>
      </add>
    </update>
  </command>
</tmch>
```



```

        <cc>BE</cc>
        <courtName>Bla</courtName>
        <caseLang>Spanish</caseLang>
    </court>
</document>
    <docType>courtCaseDocument</docType>
    <fileName>02-2013-TMCHdefect2.jpg</fileName>
    <fileType>jpg</fileType>
    <fileContent>YnJvbAo=</fileContent>
</document>
<label>
    <aLabel>a</aLabel>
</label>
<label>
    <aLabel>b</aLabel>
</label>
</case>
</add>
</update>
</command>
</tmch>

```

8.3. Managing labels linked to a case

Add or remove labels linked to a case using the update add or update remove case label section.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
    <command>
        <update>
            <id>000001132-1</id>
            <add>
                <case>
                    <id>case-00000123466989979998</id>
                    <label>
                        <aLabel>x</aLabel>
                    </label>
                </case>
            </add>
        </update>
    </command>
</tmch>

```

8.4. Managing documents linked to a case

Add documents linked to a case using the update add case label section.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
    <command>
        <update>
            <id>0000011363778801363778080-1</id>
            <add>
                <case>
                    <id>case-00000123456</id>
                    <document>
                        <docType>tmOther</docType>
                        <fileName>C:\\ddafs\\file2.png</fileName>
                        <fileType>jpg</fileType>
                        <fileContent>YnJvbAo=</fileContent>
                    </document>
                    <label>
                        <aLabel>my-name-three</aLabel>
                    </label>
                </case>
            </add>
        </update>
    </command>
</tmch>

```



```

        </case>
      </add>
    </update>
  </command>
</tmch>

```

8.5. Changing a case

To change case data use the update chg case block

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <update>
      <id>0000011363778801363778080-1</id>
      <chg>
        <case>
          <id>case-00000123456</id>
          <udrp>
            <caseNo>987654321</caseNo>
            <udrpProvider>NAF</udrpProvider>
            <caseLang>Spanish</caseLang>
          </udrp>
        </case>
      </chg>
    </update>
  </command>
</tmch>

```

8.6. Retrieve case data

The case information can be retrieved using the info:mark command.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="75" id="19" />
    <resData>
      <infData>
        <id>000001136757513215-1</id>
        <status s="verified" />
        <pouStatus s="valid" />
        <mark xmlns="urn:ietf:params:xml:ns:mark-1.0">
          <trademark>
            <id>000001136757513215-1</id>
            <markName>Example 3</markName>
            <holder entitlement="owner">
              <name>Example name</name>
              <org>Example Inc.</org>
              <addr>
                <street>123 Example Dr.</street>
                <street>Suite 100</street>
                <city>Reston</city>
                <sp>VA</sp>
                <pc>20190</pc>
                <cc>LY</cc>
              </addr>
              <email>test@test.test</email>
            </holder>
            <jurisdiction>LY</jurisdiction>
            <class>35</class>
          </trademark>
        </mark>
      </infData>
    </resData>
  </response>
</tmch>

```



```
<class>36</class>
<goodsAndServices>Dirigendas et eiusmodi featuring infringo in airfare et cartam
servicia.
</goodsAndServices>
  <regNum>234235</regNum>
  <regDate>2009-08-16T00:00:00Z</regDate>
  <exDate>2015-08-16T00:00:00Z</exDate>
</trademark>
</mark>
<label>
  <aLabel>example-one</aLabel>
  <uLabel>example-one</uLabel>
  <smInclusion enable="0" />
  <claimsNotify enable="0" />
</label>
<case>
  <id>case-165955219104862426891240536623</id>
  <udrp>
    <caseNo>123</caseNo>
    <udrpProvider>Asian Domain Name Dispute Resolution Centre</udrpProvider>
    <caseLang>Afrikaans</caseLang>
  </udrp>
  <status s="new" />
  <label>
    <aLabel>label5</aLabel>
    <status s="new" />
  </label>
  <label>
    <aLabel>label3</aLabel>
    <status s="new" />
  </label>
  <label>
    <aLabel>label2</aLabel>
    <status s="new" />
  </label>
  <label>
    <aLabel>label1</aLabel>
    <status s="new" />
  </label>
  <label>
    <aLabel>label4</aLabel>
    <status s="new" />
  </label>
  <upDate>2013-10-09T10:20:45.2Z</upDate>
</case>
<case>
  <id>case-176169232111416328571942201148</id>
  <udrp>
    <caseNo>456</caseNo>
    <udrpProvider>Asian Domain Name Dispute Resolution Centre</udrpProvider>
    <caseLang>Afrikaans</caseLang>
  </udrp>
  <status s="new" />
  <label>
    <aLabel>second1</aLabel>
    <status s="new" />
  </label>
  <upDate>2013-10-09T10:21:35.0Z</upDate>
</case>
<crDate>2013-05-03T11:58:53.3Z</crDate>
<upDate>2013-12-18T10:45:53.3Z</upDate>
<exDate>2014-05-03T00:00:00Z</exDate>
</infData>
</resData>
```




```
<trID>  
  <svTRID>poll-1391093637</svTRID>  
</trID>  
</response>  
</tmch>
```



9. RENEW

Command layout

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <renew>
      <id>00000126-1</id>
      <curExpDate>2014-01-02</curExpDate>
      <period unit="y">3</period>
    </renew>
  </command>
</tmch>
```

Please note that

1. Currently only extensions for 1 year and 3 years are allowed.
2. A renew forces a creation on a new SMD file

Response to the renew command contains a confirmation or a reason why the renew was not possible. If the command executes successfully you will also get a balance block that includes the new expiration date and a msgQ block that contains the number of pending messages.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <resData>
      <renData>
        <id>00000126-1</id>
        <exDate>2017-08-08T00:00:00Z</exDate>
        <balance>
          <amount currency="USD">56588.2500</amount>
          <statusPoints>1414</statusPoints>
        </balance>
      </renData>
    </resData>
    <trID>
      <svTRID>renew-1391087231-5310</svTRID>
    </trID>
  </response>
</tmch>
```



10. CHECK MARK

To verify if a handle is already in use (system wide) the check command can be used. This is to facilitate the choice of mark handles.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <check>
      <id>000712423-2</id>
      <id>000001436269876872643629798257-1</id>
      <id>00000113637780801363778080-1</id>
    </check>
  </command>
</tmch>
```

The results is a list of handles that can or should not be used

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="76" id="17" />
    <resData>
      <chkData>
        <cd>
          <id avail="0">000712423-2</id>
          <reason>Format: 000001XXXXXXXXXX-1</reason>
        </cd>
        <cd>
          <id avail="1">000001436269876872643629798257-1</id>
        </cd>
        <cd>
          <id avail="0">00000113637780801363778080-1</id>
          <reason>Already exists</reason>
        </cd>
      </chkData>
    </resData>
    <trID>
      <svTRID>check-1391087493</svTRID>
    </trID>
  </response>
</tmch>
```

If there are marks that change status or require attention due to external changes such as validator approval or disapproval a message referencing the mark will be pushed on a queue which can be polled in a much similar way to the EPP poll mechanism.



11. POLL

Poll messages are used to inform you of changes to the database that were triggered by a third party process. This way you are informed of any external change that is relevant to you mark record or your account status. These messages vary from warnings on the expiry of certain documents, mark records, expiration dates, over account balance messages to the notices on registered domain names.

For a list of poll codes and the messages linked to them see appendix 2.

11.1. Poll request

Check if messages are available

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <poll op="req" />
  </command>
</tmch>
```

11.1.1. Poll response (no msg)

No messages for polling are available.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1300">
      <msg>Command completed successfully; no messages</msg>
    </result>
    <trID>
      <svTRID>TMCH-000000001</svTRID>
    </trID>
  </response>
</tmch>
```

11.1.2. Poll response (msg pending)

There are messages available for polling: the first message is returned and gives the handle to the mark that has undergone changes. NOTE: do not forget to dequeue the message.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1301">
      <msg>Command completed successfully; ack to dequeue</msg>
    </result>
    <msgQ count="76" id="17">
      <qDate>2013-07-25T13:36:11.3Z</qDate>
      <msg>123 Mark has been verified and approved</msg>
    </msgQ>
    <resData>
      <infData>
        <id>00000113675751323-1</id>
        <status s="verified" />
      </infData>
    </resData>
```



```

    <trID>
      <clTRID>poll-123</clTRID>
      <svTRID>poll-1391087629</svTRID>
    </trID>
  </response>
</tmch>

```

11.1.3. Poll response (not a mark)

In case there are messages that are not related to marks you will get a different formatted message.

NOTE : do not forget to dequeue the message.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1301">
      <msg>Command completed successfully; ack to dequeue</msg>
    </result>
    <msgQ count="1" id="123">
      <qDate>2013-01-21T12:01:01.999Z</qDate>
      <msg>Watermark of USD 2500 passed</msg>
    </msgQ>
    <trID>
      <svTRID>TMCH-000000001</svTRID>
    </trID>
  </response>
</tmch>

```

11.2. Poll ack (dequeue)

Dequeue a message row be able to access the next message in the queue.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <command>
    <poll op="ack" msgID="18"/>
  </command>
</tmch>

```

11.2.1. Poll ack (response last message)

Response if there is no more message after the last dequeue.

```

<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <trID>
      <svTRID>poll-1391089968</svTRID>
    </trID>
  </response>
</tmch>

```



11.2.2. *Poll ack (response more message)*

Response if there are more messages after the last dequeue.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="2" id="124"/>
    <trID>
      <svTRID>TMCH-000000001</svTRID>
    </trID>
  </response>
</tmch>
```



12. TRANSFER

12.1. Transfer – request to execute

Check if messages are available

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
<command>
  <transfer op="execute">
    <id>000001123456789876543211113333-1</id>
    <authCode>qwertyasdfgh</authCode>
  </transfer>
</command>
</tmch>
```

12.1.1. Transfer response

If the transfer is executed successfully a trnData block is returned including the new mark id and an updated balance block.

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <msgQ count="75" id="19" />
    <resData>
      <trnData>
        <newId>000001123456789876543211113333-1</newId>
        <trnDate>2014-01-30T15:27:14.3Z</trnDate>
        <balance>
          <amount currency="USD">56565.2500</amount>
          <statusPoints>1414</statusPoints>
        </balance>
      </trnData>
    </resData>
    <trID>
      <svTRID>create-1391092034-5317</svTRID>
    </trID>
  </response>
</tmch>
```



13. ERROR RESPONSE

Error responses take the following form

```
<?xml version="1.0" encoding="utf-8"?>
<tmch xmlns="urn:ietf:params:xml:ns:tmch-1.1">
  <response>
    <result code="2303">
      <msg>Object does not exist: domain 100</msg>
    </result>
  </response>
  <trID>
    <svTRID>TMCH-000000001</svTRID>
  </trID>
</tmch>
```

You can find an overview of the codes in Appendix 2. The 'msg' label is a human readable string that can be used to further diagnose the problem. This text is not formatted or structured and should not be used to automate processes.



14. APPENDIX 1

14.1. Country codes

Andorra	AD
United Arab Emirates	AE
Afghanistan	AF
Antigua and Barbuda	AG
Anguilla	AI
Albania	AL
Armenia	AM
Angola	AO
Argentina	AR
American Samoa	AS
Austria	AT
Australia	AU
Aruba	AW
Alands islands	AX
Azerbaijan	AZ
Bosnia and Herzegovina	BA
Barbados	BB
Bangladesh	BD
Belgium	BE
Burkina Faso	BF
Bulgaria	BG
Bahrain	BH
Burundi	BI
Benin	BJ
Saint Barthélemy	BL
Bermuda	BM
Brunei Darussalam	BN
Bolivia	BO
Bonaire, Sint Eustatius and Saba	BQ
Brazil	BR
Bahamas	BS
Bhutan	BT
Bouvet Island	BV
Botswana	BW
Belarus	BY
Belize	BZ
Canada	CA
Cocos (Keeling) Islands	CC
Congo, the Democratic Republic of the	CD
Central African Republic	CF
Congo	CG
Switzerland	CH



Cote D'Ivoire	CI
Cook Islands	CK
Chile	CL
Cameroon	CM
China	CN
Colombia	CO
Costa Rica	CR
Cuba	CU
Cape Verde	CV
Curaçao	CW
Christmas Island	CX
Cyprus	CY
Czech Republic	CZ
Germany	DE
Djibouti	DJ
Denmark	DK
Dominica	DM
Dominican Republic	DO
Algeria	DZ
Ecuador	EC
Estonia	EE
Egypt	EG
Western Sahara	EH
Eritrea	ER
Spain	ES
Ethiopia	ET
Finland	FI
Fiji	FJ
Falkland Islands (Malvinas)	FK
Micronesia, Federated States of	FM
Faroe Islands	FO
France	FR
Gabon	GA
United Kingdom	GB
Grenada	GD
Georgia	GE
French Guiana	GF
Guernsey	GG
Ghana	GH
Gibraltar	GI
Greenland	GL
Gambia	GM
Guinea	GN
Guadeloupe	GP
Equatorial Guinea	GQ
Greece	GR
South Georgia and the South Sandwich Islands	GS
Guatemala	GT



Guam	GU
Guinea-Bissau	GW
Guyana	GY
Hong Kong	HK
Heard Island and Mcdonald Islands	HM
Honduras	HN
Croatia	HR
Haiti	HT
Hungary	HU
Indonesia	ID
Ireland	IE
Israel	IL
Isle Of Man	IM
India	IN
British Indian Ocean Territory	IO
Iraq	IQ
Iran, Islamic Republic of	IR
Iceland	IS
Italy	IT
Jersey	JE
Jamaica	JM
Jordan	JO
Japan	JP
Kenya	KE
Kyrgyzstan	KG
Cambodia	KH
Kiribati	KI
Comoros	KM
Saint Kitts and Nevis	KN
Korea, Democratic People's Republic of	KP
Korea, Republic of	KR
Kuwait	KW
Cayman Islands	KY
Kazakhstan	KZ
Lao People's Democratic Republic	LA
Lebanon	LB
Saint Lucia	LC
Liechtenstein	LI
Sri Lanka	LK
Liberia	LR
Lesotho	LS
Lithuania	LT
Luxembourg	LU
Latvia	LV
Libyan Arab Jamahiriya	LY
Morocco	MA
Monaco	MC
Moldova, Republic of	MD



Montenegro	ME
Saint Martin (French part)	MF
Madagascar	MG
Marshall Islands	MH
Macedonia, the Former Yugoslav Republic of	MK
Mali	ML
Myanmar	MM
Mongolia	MN
Macao	MO
Northern Mariana Islands	MP
Martinique	MQ
Mauritania	MR
Montserrat	MS
Malta	MT
Mauritius	MU
Malawi	MW
Mexico	MX
Malaysia	MY
Mozambique	MZ
Namibia	NA
New Caledonia	NC
Niger	NE
Norfolk Island	NF
Nigeria	NG
Nicaragua	NI
Netherlands	NL
Norway	NO
Nepal	NP
Nauru	NR
Niue	NU
New Zealand	NZ
Oman	OM
Panama	PA
Peru	PE
French Polynesia	PF
Papua New Guinea	PG
Philippines	PH
Pakistan	PK
Poland	PL
Saint Pierre and Miquelon	PM
Pitcairn	PN
Puerto Rico	PR
Palestinian Territory, Occupied	PS
Portugal	PT
Palau	PW
Paraguay	PY
Qatar	QA
Reunion	RE



Romania	RO
Serbia	RS
Russian Federation	RU
Rwanda	RW
Saudi Arabia	SA
Solomon Islands	SB
Seychelles	SC
Sudan	SD
Sweden	SE
Singapore	SG
Saint Helena	SH
Slovenia	SI
Svalbard and Jan Mayen	SJ
Slovakia	SK
Sierra Leone	SL
San Marino	SM
Senegal	SN
Somalia	SO
Suriname	SR
South Sudan	SS
Sao Tome and Principe	ST
El Salvador	SV
Sint Maarten (Dutch part)	SX
Syrian Arab Republic	SY
Swaziland	SZ
Turks and Caicos Islands	TC
Chad	TD
French Southern Territories	TF
Togo	TG
Thailand	TH
Tajikistan	TJ
Tokelau	TK
Timor-Leste	TL
Turkmenistan	TM
Tunisia	TN
Tonga	TO
Turkey	TR
Trinidad and Tobago	TT
Tuvalu	TV
Taiwan, Province of China	TW
Tanzania, United Republic of	TZ
Ukraine	UA
Uganda	UG
United States Minor Outlying Islands	UM
United States	US
Uruguay	UY
Uzbekistan	UZ
Holy See (Vatican City State)	VA



Saint Vincent and the Grenadines	VC
Venezuela	VE
Virgin Islands, British	VG
Virgin Islands, U.s.	VI
Viet Nam	VN
Vanuatu	VU
Wallis and Futuna	WF
Samoa	WS
Yemen	YE
Mayotte	YT
South Africa	ZA
Zambia	ZM
Zimbabwe	ZW

14.2. Jurisdictions

Andorra	AD
United Arab Emirates	AE
Afghanistan	AF
Antigua and Barbuda	AG
Anguilla	AI
Albania	AL
Armenia	AM
Angola	AO
African Regional Intellectual Property Organization (ARIPO)	AP
Argentina	AR
Austria	AT
Australia	AU
Aruba	AW
Azerbaijan	AZ
Bosnia and Herzegovina	BA
Barbados	BB
Bangladesh	BD
Bulgaria	BG
Bahrain	BH
Burundi	BI
Bermuda	BM
Brunei Darussalam	BN
Bolivia, Plurinational State of	BO
Bonaire, Sint Eustatius and Saba	BQ
Brazil	BR
Bahamas	BS
Bhutan	BT
Botswana	BW
Benelux Office for Intellectual Property (BOIP)	BX
Belarus	BY
Belize	BZ
Canada	CA



Democratic Republic of the Congo	CD
Switzerland	CH
Chile	CL
China	CN
Colombia	CO
Costa Rica	CR
Cuba	CU
Cape Verde	CV
Cyprus	CY
Czech Republic	CZ
Germany	DE
Djibouti	DJ
Denmark	DK
Dominica	DM
Dominican Republic	DO
Algeria	DZ
Ecuador	EC
Estonia	EE
Egypt	EG
Western Sahara	EH
Office for Harmonization in the Internal Market (Trademarks and Designs) (OHIM)	EM
Eritrea	ER
Spain	ES
Ethiopia	ET
Finland	FI
Fiji	FJ
Micronesia, Federated States of	FM
France	FR
United Kingdom	GB
Georgia	GE
Ghana	GH
Gambia	GM
Greece	GR
Guatemala	GT
Guyana	GY
The Hong Kong Special Administrative Region of the People's Republic of China	HK
Honduras	HN
Croatia	HR
Haiti	HT
Hungary	HU
International Bureau of the World Intellectual Property Organization (WIPO)	IB
Indonesia	ID
Ireland	IE
Israel	IL
India	IN
Iraq	IQ
Iran (Islamic Republic of)	IR
Iceland	IS



Italy	IT
Jamaica	JM
Jordan	JO
Japan	JP
Kenya	KE
Kyrgyzstan	KG
Cambodia	KH
Comoros	KM
Saint Kitts and Nevis	KN
Democratic People's Republic of Korea	KP
Republic of Korea	KR
Kuwait	KW
Cayman Islands	KY
Kazakhstan	KZ
Lao People's Democratic Republic	LA
Lebanon	LB
Saint Lucia	LC
Liechtenstein	LI
Sri Lanka	LK
Liberia	LR
Lesotho	LS
Lithuania	LT
Latvia	LV
Libya	LY
Morocco	MA
Monaco	MC
Republic of Moldova	MD
Montenegro	ME
Madagascar	MG
The former Yugoslav Republic of Macedonia	MK
Myanmar	MM
Mongolia	MN
Macao	MO
Malta	MT
Mauritius	MU
Malawi	MW
Mexico	MX
Malaysia	MY
Mozambique	MZ
Namibia	NA
Nigeria	NG
Nicaragua	NI
Norway	NO
Nepal	NP
Nauru	NR
New Zealand	NZ
African Intellectual Property Organization (OAPI)	OA
Oman	OM



Panama	PA
Peru	PE
Papua New Guinea	PG
Philippines	PH
Pakistan	PK
Poland	PL
Puerto Rico	PR
Palestinian Territory, Occupied	PS
Portugal	PT
Palau	PW
Paraguay	PY
Qatar	QA
Romania	RO
Serbia	RS
Russian Federation	RU
Rwanda	RW
Saudi Arabia	SA
Seychelles	SC
Sudan	SD
Sweden	SE
Singapore	SG
Slovenia	SI
Slovakia	SK
San Marino	SM
Somalia	SO
Suriname	SR
South Sudan	SS
Sao Tome and Principe	ST
El Salvador	SV
Sint Maarten (Dutch part)	SX
Syrian Arab Republic	SY
Swaziland	SZ
Turks and Caicos Islands	TC
Thailand	TH
Tajikistan	TJ
Timor–Leste	TL
Turkmenistan	TM
Tunisia	TN
Tonga	TO
Turkey	TR
Trinidad and Tobago	TT
Tuvalu	TV
Taiwan, Province of China	TW
United Republic of Tanzania	TZ
Ukraine	UA
Uganda	UG
United States of America	US
Uruguay	UY



Uzbekistan	UZ
Saint Vincent and the Grenadines	VC
Venezuela, Bolivarian Republic of	VE
Viet Nam	VN
Vanuatu	VU
World Intellectual Property Organization (WIPO) (International Bureau of)	WO
Samoa	WS
Yemen	YE
South Africa	ZA
Zambia	ZM
Zimbabwe	ZW

14.3. Languages

Afrikaans
Albanian
Amharic
Arabic
Armenian
Aymara
Azerbaijani
Bengali
Bislama
Bosnian
Bulgarian
Burmese
Byelorussian
Cantonese
Catalan
Chamorro
Chichewa
Chinese
Croatian
Czech
Danish
Dari
Dutch
Dzongkha
English
Estonian
Fijian
Filipino
Finnish
French
Georgian
German
Greek



Greenlandic
Guarani
Haitian Creole
Hebrew
Hindi
Hiri Motu
Hungarian
Icelandic
Indonesian
Irish
Italian
Japanese
Kazakh
Khmer
Kinyarwanda
Kirghiz
Kirundi
Korean
Lao
Latvian
Lithuanian
Luxembourgish
Macedonian
Malagasy
Malay
Maltese
Mandarin Chinese
Maori
Marshallese
Mongolian
Nauruan
Ndebele
Nepali
Niue
Norwegian
Palauan
Papiamento
Pashto
Pedi
Persian
Polish
Portuguese
Quechua
Romanian
Russian
Samoan
Serbian
Sesotho



Seychellois Creole
Shikomor
Sinhala
Slovak
Slovene
Somali
Spanish
Swahili
Swati
Swedish
Tajik
Tamil
Tetum
Thai
Tigrinya
Tok Pisin
Tongan
Tsonga
Tswana
Turkish
Turkmen
Ukrainian
Urdu
Uzbek
Venda
Vietnamese
Xhosa
Zulu



15. APPENDIX 2

15.1. Poll codes

102	Proof of use (POU) is expire
103	Proof of use (POU) has been verified and approved
104	Proof of use (POU) has not been approved and invalidated
105	Proof of use (POU) is not approved. Further attention needed
123	Mark has been verified and approved
124	Mark is not approved. Further attention needed
126	Mark has not been approved and invalidate
127	Mark has not been corrected within 20 days and invalidated.
134	Exception : validator still needs more info to approve mark
137	Mark expired
146	Correction period expired
154	Exception : user request to re-edit after changes
164	Exception : mark can be re-edited
190	POU will expire soon
191	Mark registration will expire soon
196	Warning : Mark about to be invalidated due to INCORRECT status time-out
197	Mark will expire soon
198	Mark has been deactivated
199	Mark transferred out/SMD has been revoked
210	The domain name [\$dn] was registered during sunrise
220	The domain name [\$dn] was registered during claims period
230	The domain name [\$dn] has been registered
240	The domain name [\$dn] has been registered during the Qualified Launch Program
310	SMD has been created
320	SMD has been regenerated
330	SMD has been revoked
400	Account status has gone below watermark level (\$level)
500	UDRP/Court case status change

15.2. API error codes

And human readable error messages

2001	Not a valid XML document tree
2201	Not logged in
2202	Login failed
2303	Attributes not allowed when removing a label
2303	Case <casehandle> does not exist



2303	Case language not found
2303	Case language not found in list
2303	Case status does not allow update to case block
2303	Case type does not correspond with current type (<casetype>)
2303	Could not find case object
2303	Could not find court case country code \$cc
2303	file format is not conform jpg of PDF
2303	file is too big (max <maxfilesize>)
2303	fileType <fileType> is not allowed
2303	Found virus: <virus>
2303	Invalid block in case change request
2303	Invalid court case country code <casetype>
2303	Invalid mark id format (Format: <ooid>XXXXXXXXXX-1)
2303	Invalid period length
2303	Invalid period unit
2303	Invalid transfer code
2303	Invalid udrpProvider
2303	Label <aLabel> does not exists
2303	Mark id <handle> does not exist
2303	Mark id <handle> has different curExpDate
2303	Mark id <handle> does not exist
2303	Not enough credits left to execute transfer
2303	Object does not exist
2303	Status of label <aLabel> does not allow removal
2303	Too many labels connected to case (max=<casemaxlabels>)
2303	Too many labels linked to mark (max=<maxlabels>)
2306	Cannot map <field>
2306	Case block contains sections that can not be removed
2306	Case <casehandle> does not exist
2306	caseNo is a required field
2306	Command not implemented
2306	courtName is a required field
2306	Document block in mark block not supported
2306	Error updating prepaid for mark case
2306	Error updating prepaid for mark: '. \$mark_i
2306	handle <handle> already exists
2306	Handle <handle> has invalid format - should be <ooid>XXXXXXXXXX-1
2306	Invalid case country code \$cc
2306	Invalid case id format (format should be 'case-<ooid>XXX', XXX a series of digits)
2306	Invalid class <class>
2306	Invalid country code (<ccref>) <cc>
2306	Invalid document type for court case
2306	Invalid document type for udrp case
2306	Invalid duration <duration>



2306	Invalid duration unit <duration unit>
2306	Invalid jurisdiction <jurisdiction>
2306	Invalid label <aLabel>
2306	Label block in mark block not supported
2306	Label <aLabel> already exists
2306	Mark id <handle> can not be extended with this period
2306	Mark id <handle> is not in status VERIFIED
2306	Mark id <handle> will not expire this year - no renew possible
2306	Mark ids do not match!
2306	Mark status does not allow update of mark
2306	Missing "Copy of the court order"
2306	Missing "Declaration of Proof of use including one single sample"
2306	Not allowed to add udrp/court case without label(s)
2306	not enough credit
2306	Object status does not allow extra file upload
2306	Object status does not allow udrp/court case addition
2306	Only extra POU documents can be added
2306	Only one handle allowed
2306	Output processing not implemented
2306	Policy violation : holder name or organization required
2306	Policy violation : <msg>
2306	Policy violation: no valid POU - impossible to switch smdInclusion on
2306	Policy violation : only one <nodeName> expected
2306	Policy violation : region not supported
2306	Policy violation : <ref> required
2306	refNum is a required field
2306	Too many cases connected to this mark (max=<maxcases>)
2306	Too many labels connected to case (max=<maxcaselabels>)
2306	Too many labels connected to mark (max=<maxcaselabels>)
2306	Transfer operation not implemented
2306	Unique case identifier already in use
2400	file <fileName> could not be saved



16. APPENDIX 3

16.1. Transliteration of & and @ per country and per jurisdiction.

Country /Jurisdiction	&	@
AD	i	
AE	و	ىف
AG	and	at
AI	and	at
AL	dhe	
AO	e	em
AP	et,e,و,le,and,na	em,kwenye,ىف,at,ka
AR	y	
AS	and	at
AT	und	
AU	and	at
AW	en,y	
AX	och	
AZ	və	
BA	i	at
BB	and	at
BD	و	
BE	en,et,und	
BF	et	
BG	и	
BH	و	ىف
BI	et	
BJ	et	
BL	et	
BM	and	at
BN	dan,and	di,at
BO	y,wan	
BQ	en	
BR	e	em
BS	and	at
BV	og	
BW	le,and	at,ko
BX	en,et,y,und,an	
BY	и	
BZ	and	at
CA	et,and	at
CC	and	at



CD	et	
CF	et	
CG	et	
CH	et,e,und	
CI	et	
CK	and	at
CL	y	
CM	et,and	at
CN	和	
CO	y	
CR	y	
CU	y	
CV	e	em
CW	en,y	
CX	and	at
CY	και,ve	
CZ	a	
DE	und	
DJ	et,و	ىف
DK	og	
DM	and	at
DO	y	
DZ	و	ىف
EC	y	
EE	ja	
EG	و	ىف
EH	و	ىف
EM	och,a,og,και,y,□i,и,u,ja ,in,agus,i,et,e,und,en,ir, és,un,and,an,ve	eta,w,em,at,afna,ät,ag
ER	و,and	ىف,at
ES	y	
FI	et,och	ät
FJ	and	at
FK	and	at
FM	and	at
FO	og	
FR	et	
GA	et	
GB	and	at
GD	and	at
GE	da	
GF	et	
GG	et,and	at
GH	and	at



GI	and	at
GL	og	
GM	and	at
GN	et	
GP	et	
GQ	et,y	
GR	και	
GS	and	at
GT	y	
GU	and	at
GW	e	em
GY	and	at
HK	和,and	at
HM	and	at
HN	y	
HR	i	
HT	et,ak	
HU	és	
IB	dhe,a,和,da,dan,le,og,và,□i,y,in,agus,ва,1,м,ྐྐྐྐ,е,et,en,мен,ir,б олон,un,and,och,و,ア ンド,και,и,ja,i,und,그 리고,və,és,na,ve,an	w,di,да,kwenye,ät,ag,на,дээр,at,ف ى,ka,afna,кучукча,hjá,ko,em,eta
ID	dan	at
IE	agus,and	at,ag
IL	و,1	ىف
IM	and	at
IN	and,और	at,में
IO	and	at
IQ	و	ىف
IR	و	
IS	og	hjá
IT	e	
JE	et,and	at
JM	and	at
JO	و	ىف
JP	ア ンド	
KE	and,na	kwenye,at
KG	и	
KI	and	at
KM	et,و	ىف
KN	and	at
KP	그리고	
KR	그리고	



KW	و	ىف
KY	and	at
KZ	мен,и	да
LB	و	ىف
LC	and	at
LI	und	
LK	மற்றும்	
LR	and	at
LS	le,and	at,ka
LT	ir	eta
LU	et,und,an	
LV	un	at
LY	و	ىف
MA	و	ىف
MC	et	
MD	□i	
ME	i	
MF	et	
MG	et	
MH	and	at
MK	i	at
ML	et	
MN	болон	дээр
MO	和	
MP	and	at
MQ	et	
MR	و	ىف
MS	and	at
MT	u,and	at
MU	and	at
MW	and	at
MX	y	
MY	dan	di
MZ	e	em
NA	and	at
NC	et	
NE	et	
NF	and	at
NG	and	at
NI	y	
NL	en	
NO	og	
NR	and	at
NU	and	at



NZ	and	at
OA	et,e,y,g,and	em,ف,ى,at
OM	و	ىف
PA	y	
PE	y,wan	
PF	et	
PG	and	at
PH	and	at
PK	and,روا	bamutabiq,م,ى,at
PL	i	w
PM	et	
PN	and	at
PR	y,and	at
PS	و	ىف
PT	e	em
PW	ha,and	at
PY	y,ha	
QA	و	ىف
RE	et	
RO	□i	
RS	i	
RU	и	
RW	et,and	at
SA	و	ىف
SB	and	at
SC	et,and	at
SD	و	ىف
SE	och	
SG	மற்றும்,和,dan,and	di,at
SH	and	at
SI	in	afna
SJ	og	
SK	a	
SL	and	at
SM	e	
SN	et	
SO	و	ىف
SR	en	
SS	and	at
ST	e	em
SV	y	
SX	en,and	at
SY	و	ىف
SZ	and,na	ngama,at



TC	and	at
TD	et,و	ىف
TF	et	
TG	et	
TH	และ	ທີ່
TK	and	at
TL	e	em
TN	و	ىف
TO	and	at
TR	ve	
TT	and	at
TV	and	at
TW	和	
TZ	and,na	kwenye,at
UA	i	на
UG	and	at
UM	and	at
US	et,and	at
UY	y	
UZ	и,ва	кучукча
VA	e	
VC	and	at
VE	y	
VG	and	at
VI	and	at
VN	và	
VU	et,and	at
WF	et	
WO	dhe,a,和,da,dan,le,og, và,şì,y,in,agus,ва,г,мр дрм,е,et,en,мен,иг,бо лон,un,and,och,و,アン ド,ка,и,ja,i,und,그리 고,vø,és,na,ve,an	w,di,да,kwenye,ät,ag,на,дээр,at,ف ى,ka,afna,кучукча,hjá,ko,em,eta
WS	and	at
YE	و	ىف
YT	et	
ZA	en,kanye-no-,kanye- no,le,and,na,na-	hi,ngama-,ngama,nga,at,ka
ZM	and	at
ZW	and	at