                     TMCH functional specifications
                     draft-lozano-tmch-func-spec-01

Abstract

   This document describes the requirements, the architecture and the
   interfaces between the Trademark Clearing House (TMCH) and Domain
   Name Registries as well as between the TMCH and Domain Name
   Registrars for the provisioning and management of domain names during
   Sunrise and Trademark Claims Period of a domain name Registry.

Status of this Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

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

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on September 24, 2013.

described in the Simplified BSD License.

Table of Contents

1. Introduction

Domain Name Registries may operate in special modes within certain
periods of time to facilitate registration of domain names.

Along with the upcoming introduction of new generic Top Level Domains
(gTLD), two special modes will come into effect:

o Sunrise Period

o Trademark Claims Period

The Sunrise and Trademark Claims Periods are defined in the gTLD
Applicant Guidebook [ICANN-GTLD-AGB-20120604].

This document describes the requirements, the architecture and the
interfaces between the Trademark Clearing House (TMCH) and Domain
Name Registries as well as between the TMCH and Domain Name
Registrars for the provisioning and management of domain names during
the Sunrise and Trademark Claims Periods.

After the Terminology (Section 2) the Glossary (Section 3) is listed,
followed by the architecture (Section 4) and the different process
descriptions (Section 5). Afterwards, in Section 6, the necessary
Data Formats are defined, followed by their formal syntax
(Section 7).

For any date and/or time indications, Coordinated Universal Time
(UTC) applies.

1.1. Discussion

Any interested parties are invited to contribute to the discussion of
this draft and provide feedback. The discussion currently takes
place on the tmch-tech discussion list <tmch-tech@icann.org>.
Interested parties can subscribe to this mailing list via
<https://mm.icann.org/mailman/listinfo/tmch-tech>.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
"SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
document are to be interpreted as described in [RFC2119].

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

implementation.

"tmNotice-1.0" is used as an abbreviation for
"urn:ietf:params:xml:ns:tm-notice-1.0". The XML namespace prefix
"tmNotice" is used, but implementations MUST NOT depend on it and
instead employ a proper namespace-aware XML parser and serializer to
interpret and output the XML documents.

3. **Glossary**

In the following the most common terms are briefly Explained:

o CA: Certificate Authority, see [RFC5280] and [RFC6818]

o CSV: Comma-Separated Values, see [RFC4180]

o CNIS, Claims Notice Information Service: This service provides
   trademark claims notices to Registrars.

o CRC32, algorithm used in the ISO 3309 standard and in section
   8.1.1.6.2 of ITU-T recommendation V.42.

o CRL: Certificate Revocation List, see [RFC5280] and [RFC6818].

o Date and time, dateandtime: Date and time are specified following
   the standard "Date and Time on the Internet specification", see
   [RFC3339].

o DN: Domain Name, see [RFC1034]

o DNS: Domain Name System, see [RFC1034]

o DNL, Domain Name Label: An ASCII string used in the DNS (an FQDN
   is composed by such labels separated by dots) as defined in
   [RFC1034]. For IDNs the A-Label is used [RFC5890].

o EPP: Extensible Provisioning Protocol, see [RFC5730].

o FQDN: Fully Qualified Domain Name (e.g. myname.example.com)

o HTTP: Hypertext Transfer Protocol, see [RFC2616]

o HTTPS: HTTP over TLS (Transport Layer Security), [RFC2818]. Only
   TLS 1.0, 1.1 and 1.2 are supported.

o ICANN-CA: ICANN's Certificate Authority (CA); Trust Anchor for the
   SMD PKI model.

o IDN: Internationalized Domain Names, see [RFC5890]

o LORDN, List of Registered Domain Names: This is the list of
   registered domain names matching a DNL of a PRM that Registries
   daily upload to the TMDB (during the NORDN process).

o Lookup Key: A random string of up to 64 chars from the set [a-zA-
   Z0-9/] to be used as the lookup Key by Registrars to obtain the
   claims notice using the CNIS. Lookup Keys are unique and are
   related to one DNL only.

o NORDN, Notification of Registered Domain Names: The process by
   which Registries daily upload their recent LORDN to the TMDB.

o PGP: Pretty Good Privacy, see [RFC4880]

o PKI: Public Key Infrastructure, see [RFC5280] and [RFC6818]

o PRM, Pre-registered mark: Mark that has been pre-registered with
   the TMCH.

o Registrant: Person or Organization registering a Domain Name via a
   Registrar.

o Registrar, Domain Name Registrar: Entity that registers Domain
   Names with the Registry on behalf of the Registrant.

o Registry, Domain Name Registry: Entity that accepts Domain Name
   registrations from Registrars, maintains the central Database of
   Registered Domains.

o SMD, Signed Mark Data: A cryptographically signed token issued by
   the TMV to the TMH to be used in the Sunrise Period to apply for a
   Domain Name that matches a DNL of a PRM; see also
   [I-D.lozano-tmch-smd]

o SMD File: A file containing the SMD (see above) and some human
   readable data. The latter is usually ignored in the processing of
   the SMD File. See also Section 6.4.

o SMDM, SMD Manager: A entity managing the SMDs, mainly maintaining
   lists of revoked SMDs; see also [I-D.lozano-tmch-smd]

o SMD revocation list: The SMD revocation list is used by Registries
   (and optionally by Registrars) during the Sunrise Period to ensure
   that an SMD is still valid (i.e. not revoked). The SMD revocation
   list has a similar function as CRLs used in PKI.

o SRS: Shared Registration System, see also
   [ICANN-GTLD-AGB-20120604].

o Sunrise Period: During this period Domain Names matching a DNL of
   a PRM can be exclusively obtained by the respective mark holders.
   For domain names matching a PRM, a special process applies to
   ensure a TMH gets informed on the registration of a domain name
   matching his/her PRM. Every launch of a new gTLD Registry starts
   with a Sunrise Period, followed by a Trademark Claims Period (see
   below).

o TLD: Top Level Domain Name, see [RFC1591]

o Trademark, mark: Trademarks are used to claim exclusive properties
   of products or services. A trademark is typically a name, word,
   phrase, logo, symbol, design, image, or a combination of these
   elements. For the scope of this document only textual trademarks
   are relevant.

o Trademark Claims, Claims: Provides information to enhance the
   understanding of the Trademark rights being claimed by the
   Trademark Holder.

o Trademark Claims Notice, Claims Notice, Trademark Notice, TCN: A
   Trademark Claims Notice consist of one or more Trademark Claims
   and are provided to prospective Registrants of Domain Names.

o Trademark Claims Notice Identifier, Claims Notice Identifier,
   Claims Notice ID, TCNID: An element of the Trademark Claims Notice
   (see above), identifying said Claims Notice. The Trademark Claims
   Notice identifier is specified in the element <tmNotice:id>.

o Trademark Claims Period: During this period, a special process
   applies to DNs matching DNL derived from PRMs, to ensure a TMH
   gets informed on the registration of a domain name matching his
   PRM. For DNs matching DNL derived from PRMs, Registrars show a
   Trademark Claims Notice to prospective Registrants that has to be
   acknowledged before registration.

o TMCH, Trademark Clearinghouse: The Trademark Clearinghouse is an
   ICANN central repository for information to be authenticated,
   stored, and disseminated, pertaining to the rights of trademark
   holders. The Clearinghouse is split into two functions TMV and
   TMDB (see below). There could be several entities performing the
   TMV function, but only one entity performing the TMDB function.

o TMDB, Trademark Clearinghouse Database: Serves as a database of
   the TMCH to provide information to the new gTLD Registries and

Registrars to support Sunrise or Trademark Claims Services. There
is only one TMDB in the TMCH that concentrates the information
about the "verified" Trademark records from the TMVs.

o TMH, Trademark Holder: The person or organization owning rights on
a Trademark.

o TMV, Trademark Validator, Trademark validation organization: An
entity authorized by ICANN to authenticate and validate
registrations in the TMDB ensuring the marks qualify as registered
or are court-validated marks or marks that are protected by
statute or treaty. This entity would also be asked to ensure that
proof of use of marks is provided, which can be demonstrated by
furnishing a signed declaration and one specimen of current use.

o UTC: Coordinated Universal Time, as maintained by the Bureau
International des Poids et Mesures (BIPM); see also [RFC3339].

4. Architecture

4.1. Sunrise Period

Architecture Sunrise Period

```
                 SMD hand over (out of band;
                 trivial if Registrant == TMH)
          ...........................................
          :                                         :
          :             .'''''''''''''''''.         :
          :           .       TMCH         .        :
          :           .....................        :
         v            .                  .         :
     .------------.   . .--------------.  . hv   .------.
     | Registrant |   . |      TMV     |<-------->| TMH |
     '------------'   . '--------------'  . vh  '-----'
          |           .      |      ^    \ .
          |           .      |      |     \.
       tr |           . vs   |      |      \
          |           .      |    | dv   .\
          v           .      v      |    . \
     .-----------.  sr .   .---.    |    .  \
   .->| Registrar |<.........| S | vd |    .   \
   : '-----------'     .   | M |    |    .    \
   :       |         sy  . | D |    v    .     \
   :    ry |      .-----------| M |   .---.   . vc \
   :       |    |     .   '---'   | T |  .      \
   :       v    v      .          | M |  .       v
   :  .-----------.     .          | D |  .  .----------.
   :  | Registry  |----------------->| B |  .  | ICANN-CA |
   :  '-----------'     .      yd  '---'   .  '----------'
   :        ^           .                   .     |   :
   :        |           '''''''''''''''''''           |   :
   :        | cy                                    |   :
   : cr     '-------------------------------------'   :
   :...........................................:
```

Figure 1

4.2. Trademark Claims Period

Architecture Trademark Claims Period

```
                        .'''''''''''.
                        .    TMCH       .
                        ..............
                        .             .
   .------------.       .  .-------.  . hv    .-----.
   | Registrant |       .  |  TMV  |<------->| TMH |
   '------------'       .  '-------'  .  vh  '-----'
         |              .       ^     .
      tr |              .       | dv  .
         |              .    vd |     .
         v              .       v     .
   .-----------.  dr    .  .-------.   .
   | Registrar |<-------|    T   |   .
   '-----------'        .  |       |   .
         |              .  |   M   |   .
      ry |              .  |       |   .
         v              .  |   D   |   .
   .-----------.  dy    .  |       |   .
   | Registry  |<------>|    B   |   .
   '-----------'   yd . '-------'   .
                        .             .
                        '''''''''''''
```

Figure 2

4.3. Interfaces

In the sub-sections below follows a short description of each
interface to provide an overview of the architecture. More detailed
descriptions of the relevant interfaces follow further below
(Section 5).

4.3.1. hv

The TMH registers a mark with a TMV via the hv interface.

After the successful registration of the mark, the TMV makes
available a SMD (Signed Mark Data) file (see also Section 6.4) to the
TMH to be used duringn the Sunrise Period.

The specifics of the hv interface are beyond the scope of this
document.

4.3.2. vd

After successful mark registration, the TMV ensures the TMDB inserts
the corresponding DNLs and mark information into the database via the
vd interface.

The specifics of the vd interface are beyond the scope of this
document.

4.3.3. dy

Not used during the Sunrise Period.

During the Trademark Claims Period the Registry fetches the latest
DNL list from the TMDB via the dy interface in regular intervals.
The protocol used on the dy interface is HTTPS.

4.3.4. tr

The Registrant communicates with the Registrar via the tr interface.

The specifics of the tr interface are beyond the scope of this
document.

4.3.5. ry

The Registrar communicate with the Registry via the ry interface.
The ry interfaces is typically implemented in EPP [RFC5730]. A TMCH
specific EPP standard extension is yet to be developed.

4.3.6. dr

Not used during the Sunrise Period.

During the Trademark Claims Period the Registrar fetches the
Trademark Claims Notice from the TMDB (to be displayed to the
Registrant via the tr interface) via the dr interface. The protocol
used for fetching the Trademark Claims Notice is HTTPS [RFC2818].

4.3.7. yd

During the Sunrise period the Registry notifies the TMDB daily via
the yd interface of all domain names registered.

During the Trademark Claims period, the Registry notifies daily the
TMDB via the yd interface of all domains registered that matched an
entry in the Registry's DNL during the creation of the domain name.

The protocol used on the yd interface is HTTPS.

4.3.8. dv

The TMDB notifies the TMV of all domains registered that match a mark
registered in the TMDB by that TMV as informed by Registries via the
dv interface.

The specifics of the dv interface are beyond the scope of this
document.

4.3.9. vh

The TMV notifies the TMH via the vh interface after a domains has
been registered that matches a PRM of this TMH.

The specifics of the vh interface are beyond the scope of this
document.

4.3.10. vs

The TMV requests to add a revoked SMD to the list of revoked SMDs at
the SMDM.

The specifics of the vs interface are beyond the scope of this
document.

Not relevant during the Trademark Claims Period.

4.3.11. sy

During the Sunrise Period the Registry fetches the most recent list of revoked SMDs from the SMDM via the sy interface in regular intervals. The protocol used on the sy interface is HTTPS.

Not relevant during the Trademark Claims Period.

4.3.12. sr

During the Sunrise Period the Registrar may fetch the most recent list of revoked SMDs from the SMDM via the sr interface. The protocol used on the sr interface is the same as on the sy interface (s. above), i.e. HTTPS.

Not relevant during the Trademark Claims Period.

4.3.13. vc

The TMV requests to add a revoked TMV certificate to the CRL at the ICANN-CA via the vc interface.

The specifics of the vc interface are beyond the scope of this document.

Not relevant during the Trademark Claims Period.

4.3.14. cy

During the Sunrise Period the Registry fetches the most recent CRL from the ICANN-CA via the cy interface in regular intervals. The CRL is mainly used for validation of TMV certificates. The protocol used on the cy interface is HTTPS [RFC2818].

Not relevant during the Trademark Claims Period.

4.3.15. cr

During the Sunrise Period the Registrar may fetch the most recent CRL from the ICANN-CA via the cr interface. The protocol used on the cr interface is the same as on the cy interface (s. above).

Not relevant during the Trademark Claims Period.

5. **Process Decriptions**

5.1. **Bootstrap**

5.1.1. **Registries**

5.1.1.1. **Credentials**

   Each Registry will receive authentication credentials from the TMDB/
   SMDM to be used:

   o During the Sunrise Period to fetch the lists of revoked SMDs from
      the SMDM via the sy interface ([Section 4.3.11](#)).

   o During Trademark Claims Period to fetch the lists of DNLs from the
      TMDB via the dy interface ([Section 4.3.3](#)).

   o During the NORDN process to notify the LORDN to the TMDB via the
      yd interface ([Section 4.3.7](#)).

   o Credentials are created per TLD and provided to the registry
      operator which is the contracting party for the TLD.

5.1.1.2. **TMCH Trust Anchor**

   Each Registry MUST fetch the X.509 certificate ([[RFC5280](#)] /
   [[RFC6818](#)]) of the ICANN-CA (Trust Anchor) from
   < https://ca.icann.org/tmch.crt > to be used:

   o during the Sunrise Period to validate the TMV certificates and the
      CRL of TMV certificates.

5.1.1.3. **TMDB/SMDM PGP Key**

   The TMDB MUST provide each Registry with the public portion of the
   PGP Key used by TMDB and SMDM, to be used:

   o During the Sunrise Period to perform integrity checking of the
      list of revoked SMDs fetched from the SMDM via the sy interface
      ([Section 4.3.11](#)).

   o During Trademark Claims Period to perform integrity checking of
      the list of DNL fetched from the TMDB via the dy interface
      ([Section 4.3.3](#)).

5.1.2. Registrars

5.1.2.1. Credentials

Each ICANN-accredited Registrar will receive authentication
credentials from the TMDB to be used:

o During the Sunrise Period to (optionally) fetch the lists of
   revoked SMDs from the SMDM via the sr interface (Section 4.3.12).

o During Trademark Claims Period to fetch Claims Notices from the
   TMDB via the dr interface (Section 4.3.6).

5.1.2.2. TMCH Trust Anchor

Registrars MAY fetch the X.509 certificate ([RFC5280] / [RFC6818]) of
the ICANN-CA (Trust Anchor) from < https://ca.icann.org/tmch.crt > to
be used:

o during the Sunrise Period to (optionally) validate the TMV's
   certificates and the CRL of TMV certificates.

5.1.2.3. TMDB PGP Key

Registrars MUST receive the public portion of the PGP Key used by
TMDB and SMDM from the TMDB administrator to be used:

o during the Sunrise Period to (optionally) perform integrity
   checking of the list of revoked SMDs fetched from the SMDM via the
   sr interface (Section 4.3.12).

5.1.2.4. IP Addresses for Access Control

The Registrars MUST provide to the TMDB administrator all IP
addresses, which will be used by their hosts to fetch Trademark
Claims Notices via the dr interface (Section 4.3.6). The TMDB
restricts access to known IP Addresses only. This access restriction
is applied in addition to HTTP Basic access authentication (for
credentials to be used, see Section 5.1.2.1). The TMDB MAY limit the
number of IP addresses to be accepted per Registrar.

5.2. Sunrise Period

5.2.1. Domain Registration

Registration during Sunrise Period

```
 .------------.  .-----------.                   .----------.
 | Registrant |  | Registrar |                   | Registry |
 '------------'  '-----------'                   '----------'
        |             |                                |
        |   Request DN |                               |
        | Registration |                               |
        |  (with SMD)  |                                |
        |------------->|    Check DN availability      |
        |             |---------------------------->|
        |             |                                |
        |             |.--------. DN unavail. .-------------.
        |             || ABORT  |<-----------( DN available? )
        |             |'-------'           no '-------------'
        |             |                                | yes
        |             |                                |
        |             |   Request DN registration      |
        |             |        (with SMD)              |
        |             |---------------------------->|
        |             |                                |
        |             |            .--------------------------------.
        |             |            |      Validate SMD and          |
        |             |            | corresponding TMV certificate |
        |             |            '--------------------------------'
        |             |                                |
        |             |.-------. Error  .----------------------.
        |             ||  TRY  |<------( Validation successful? )
        |             || AGAIN |    no '----------------------'
        |             ||  OR   |                        | yes
        |             || ABORT |                        |
        |             |'-------'                        |
        |             |                                |
        |             |   DN registered                |
        |   DN regist. |<----------------------------|
        |<------------|                                |
        |             |                                |
```

Figure 3

5.2.2. **DN Registration by Registries**

   Registries MUST perform a minimum set of checks for verifying DN
   registrations during Sunrise Period upon reception of a registration
   request over the ry interface (Section 4.3.5). If any of these
   checks fails the Registry MUST abort the DN registration. Each of
   these checks MUST be performed before the DN is registered (see
   Section 3).

   In case of asynchronous registrations (e.g. auctions), the minimum
   set of checks MAY be performed when creating the intermediate object
   (e.g. a domain name application) used for DN registration. However,
   the validation of the SMD compared to the moment of effective
   registration of the DN MUST NOT be older than a period of time (i.e.,
   the SMD-validation validity period) that is defined in the PRM
   requeriments document referenced in Specification 7 of the gTLD
   Applicant Guidebook [ICANN-GTLD-AGB-20120604].

   Performing the minimum set of checks Registries MUST verify that:

   1. A SMD has been received from the Registrar along with the DN
       registration.

   2. The certificate of the TMV has been correctly signed by the
       ICANN-CA. (The certificate of the TMV is contained within the
       SMD.)

   3. The validity period of the TMV certificate is within the allowed
       range.

   4. The certificate of the TMV is not be listed in the CRL file
       specified in the CRL distribution point of the TMV certificate.

   5. The signature of the SMD (signed with the TMV certificate) is
       valid.

   6. The validity period of the SMD is within the allowed range (<smd:
       notBefore> and <smd:notAfter> elements).

   7. The SMD has not been revoked, i.e., is not contained in the SMD
       revocation list.

   8. The leftmost DNS label (A-label in case of IDNs) of the DN being
       registered matches one of the labels (<mark:label>) elements in
       the SMD.

   These procedure apply to all DN registrations at the second level as
   well as to all other levels subordinate to the TLD that the Registry

accepts registrations for.

5.2.3. TMCH Sunrise Services for Registries

5.2.3.1. SMD Revocation List

   A new SMD revocation list MUST be published by the SMDM twice a day,
   by 00:00 and 12:00 UTC.

   Registries MUST download the latest version of the SMD revocation
   list at least once every 24 hours.

   Update SMD revocation list

```
       .----------.                                        .------.
       | Registry |                                        | SMDM |
       '----------'                                        '------'
            |                                                  |
    .----------------.                                         |
    |  Periodically, |                                         |
    |    at least    |                                         |
    | every 24 hours |                                         |
    '----------------'                                         |
            |                                                  |
            |------------------------------------------------->|
            | Download latest revocation list for SMD certificates |
            |<-------------------------------------------------|
            |                                                  |
            |                                                  |
```

                              Figure 4

5.2.3.2. Certificate Revocation List

   Registries MUST update their local copy of the CRL at least every 4
   hours using the CRL distribution point specified in the TMV
   certificate.

   Operationally, the CRL file and CRL distribution point is the same
   for all TMVs and (at publication of this document) located at
   < https://crl.icann.org/tmch.crl >.

   Update CRL for TMV certificates

```
     .----------.                              .----------.
     | Registry |                              | ICANN-CA |
     '----------'                              '----------'
          |                                          |
   .----------------.                                |
   | Periodically,  |                                |
   |    at least    |                                |
   | every 24 hours |                                |
   '----------------'                                |
          |                                          |
          |----------------------------------------->|
          |    Download latest CRL for TMV certificates  |
          |<-----------------------------------------|
          |                                          |
          |                                          |
```

                         Figure 5

5.2.3.3. **Notice of Registered Domain Names**

   The Registry MUST send the LORDN file containing all DNs registered
   on the previous calendar day (UTC), to the TMDB (over the yd
   interface, Section 4.3.7) between 00:00 and 03:00 UTC each day.

   The TMDB MUST process each uploaded LORDN file within 30 minutes of
   the finalization of the upload and have the response indicating the
   errors and warnings to the Registry.

   A LORDN file with headers and no data elements is used to indicate
   that there were no registrations the previous calendar day (UTC).

   If a new LORDN file (containing the registered DNs for the same date)
   is uploaded before 03:00 UTC, the previous LORDN file will be
   discarded and the new LORDN file will be processed instead.

   At 03:00 UTC, the last uploaded LORDN file is considered final, given
   no ERRORs (i.e. only OKs and warnings) occur during the processing of
   the file at the TMDB. If a LORDN file is considered final, it is
   stored in the TMDB, and information is sent to the TMVs for informing
   the TMHs. Any LORDN file (for the same date) that is uploaded later
   than 03:00 will be ignored. The TMDB SHOULD not allow upload of
   LORDN files outside the 00:00 to 03:00 window.

   In case that a LORDN file was not received for the previous calendar
   day or errors are encountered, a manual process between the TMDB and
   the Registry will be performed to transmit the file.

NORDN

```
      .----------.              .------.           .-----.          .-----.
      | Registry |              | TMDB |           | TMV |          | TMH |
      '----------'              '------'           '-----'          '-----'
           |                       |                 |                |
   .------------------.      .-----------.           |                |
   | Periodically;    |      | Wait for  |<--------.  |                |
   | every 24 hours   |      | LORDN     |         | |                |
   | (between 00:00   |      '-----------'         | |                |
   | and 03:00 UTC)   |           |                | |                |
   '------------------'           |                | |                |
           |                      |                | |                |
   .--------->|     Upload LORDN  |                | |                |
   |          |------------------->|               | |                |
   |          |                    |               | |                |
   |          |        .-------------------------. | |                |
   |          |        | Verify each domain name | | |                |
   |          |        |   in the uploaded file  | | |                |
   |          |        |  and write results to a | | |                |
   |          |        |  .err-file (within 30') | | |                |
   |          |        '-------------------------' | |                |
   |          |                    |               | |                |
   |          |   Download log file |              | |                |
   |          |<-------------------|               | |                |
   |          |                    |               | |                |
   | .-----------------.   .---------------.       | |                |
   | | Check whether   |   / everything fine \ no  | |                |
   | |    log file     |  (  (i.e. no errors  )----' |                |
   | | contains errors |   \ in log file )?  /       |                |
   | '-----------------'    '---------------'        |                |
   |         |                     | yes             |                |
   | .---------------.             |                 |                |
   | / everything fine \ yes       |                 |                |
   |(  (i.e. no errors  )-----.    |  Notify TMVs    |                |
   | \ in log fiel )?  /      |    |  on the LORDN   |                |
   |  '---------------'       |    | pre-registered  |                |
   |         | no             v    |  with said TMV  |                |
   | .---------------.   .------.  |--------------->|                 |
   '-| Correct Errors |   | DONE |  |                |                |
     '---------------'   '------'  |                | Notify each    |
           |                     |                  | affected TMH   |
           |                     |                  |-------------->|  |
           |                     |                  |                |
```
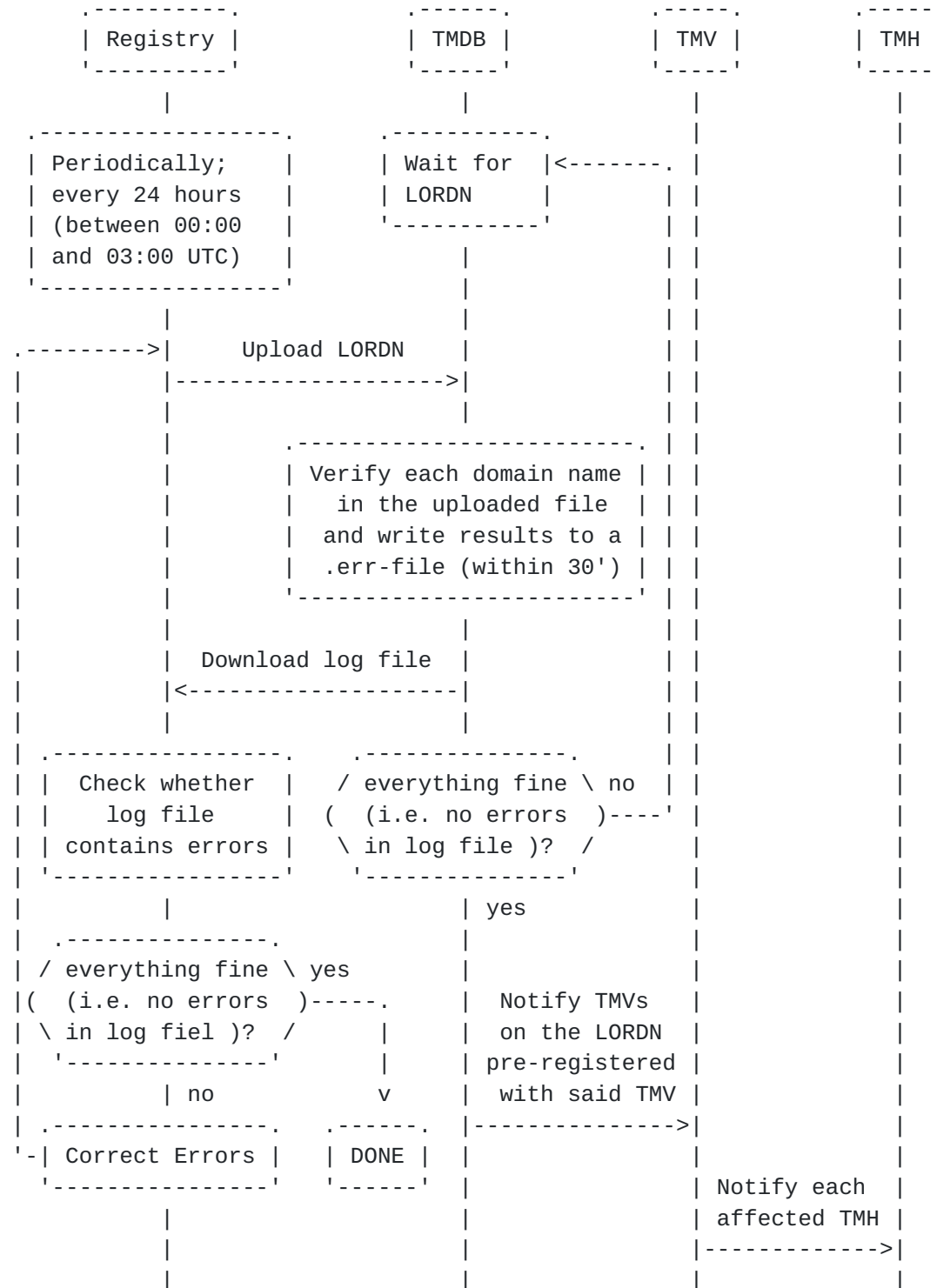
Figure 6

The format used for the LORDN is described in Section 6.3

5.2.4. DN registration by Registrars

Registrars MAY choose to perform the checks for verifying DN
registrations as performed by the Registries (see Section 5.2.2)
before sending the command to register a DN.

5.2.5. TMCH Sunrise Services for Registrars

The processes described in Section 5.2.3.1 and Section 5.2.3.2 are
also available for Registrars to optionally validate the SMDs
received.

5.3. Trademark Claims Period

5.3.1. Domain Registration
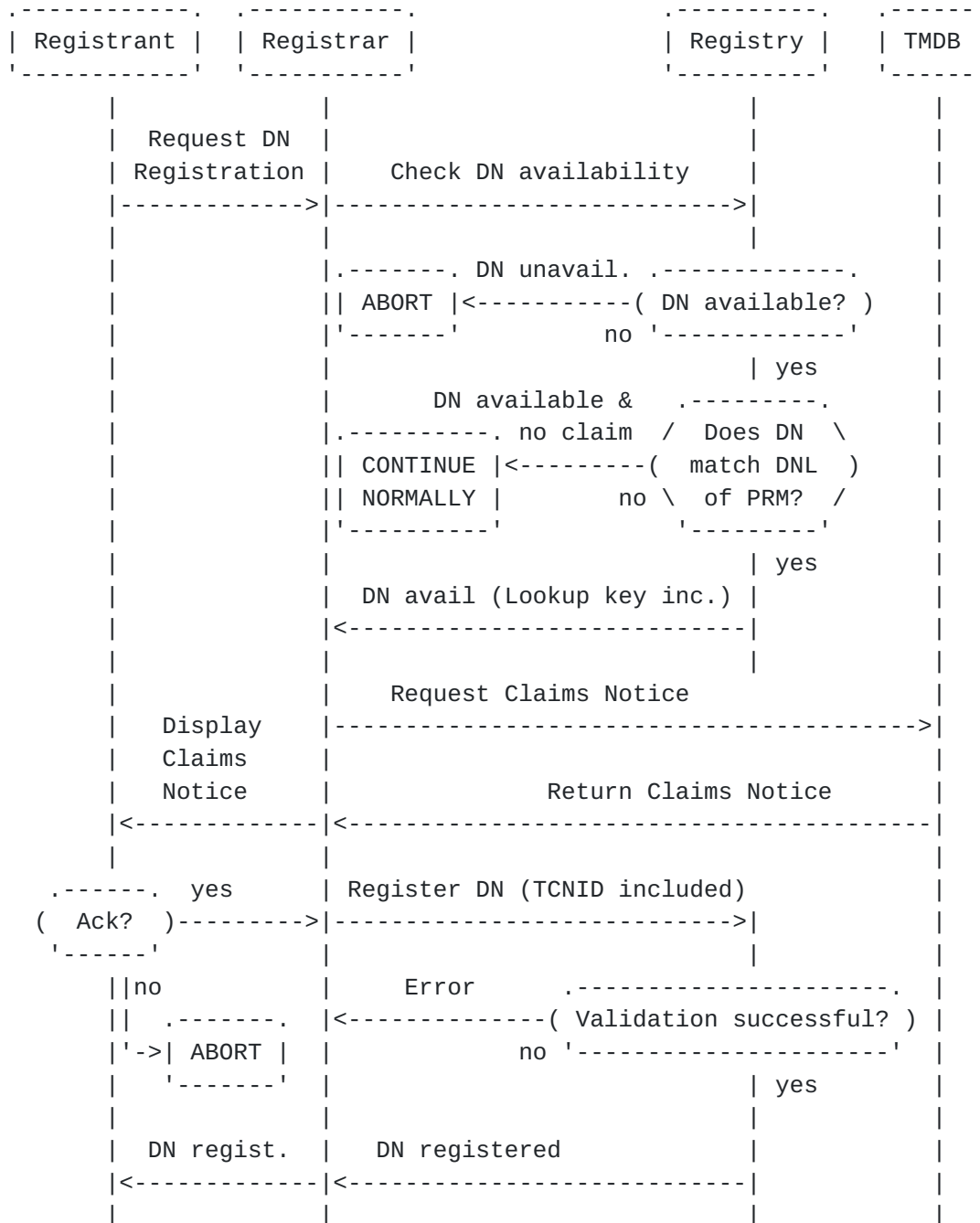
   Registration during Trademark Claims Period

```
   .------------.  .-----------.                    .-----------.  .------.
   | Registrant |  | Registrar |                    | Registry  |  | TMDB |
   '------------'  '-----------'                    '-----------'  '------'
         |               |                                |             |
         |   Request DN  |                                |             |
         | Registration  |    Check DN availability       |             |
         |-------------->|------------------------------->|             |
         |               |                                |             |
         |               |.--------. DN unavail. .--------------.        |
         |               || ABORT  |<-----------( DN available? )       |
         |               |'--------'         no '--------------'        |
         |               |                                | yes         |
         |               |       DN available &    .----------.         |
         |               |.----------. no claim   /  Does DN   \        |
         |               || CONTINUE |<---------(   match DNL   )       |
         |               || NORMALLY |      no  \  of PRM?     /        |
         |               |'----------'           '----------'           |
         |               |                                | yes         |
         |               |  DN avail (Lookup key inc.)    |             |
         |               |<-------------------------------|             |
         |               |                                |             |
         |               |       Request Claims Notice                  |
         |   Display     |--------------------------------------------->|
         |   Claims      |                                |             |
         |   Notice      |             Return Claims Notice             |
         |<--------------|<---------------------------------------------|
         |               |                                |             |
     .-------.  yes      | Register DN (TCNID included)                  |
    (  Ack?   )--------->|------------------------------->|             |
     '------'            |                                |             |
       ||no             |        Error     .----------------------.     |
       ||   .--------.  |<--------------( Validation successful? ) |     |
       |'->| ABORT  |  |              no '----------------------'  |     |
       |    '-------'   |                                | yes      |     |
       |               |                                |             |
       |   DN regist.   |   DN registered                |             |
       |<--------------|<-------------------------------|             |
       |               |                                |             |
```

                         Figure 7

5.3.2. DN registration by Registries

   During Trademark Claim Periods, Registries perform two main
   functions:

   o Whenever a Registrar queries the domain availability (e.g., an EPP
      domain Check command) that matches a DNL of a PRM (over the ry
      interface, Section 4.3.5) for availability, the Registry MUST
      return the corresponding Lookup Key.

   o For any DN matching a DNL of a PRM, Registries MUST perform a
      minimum set of checks for verifying DN registrations during
      Trademark Claims Period upon reception of a registration request
      over the ry interface (Section 4.3.5). If any of these checks
      fails the Registry MUST abort the DN registration. Each of these
      checks MUST be performed before the DN is registered (see
      Section 3).

   o In case of asynchronous registrations (e.g. auctions), the minimum
      set of checks MAY be performed when creating the intermediate
      object (e.g. a domain name application) used for DN registration.
      However, the acknowledgement of the Trademark Notice compared to
      the moment of effective registration of the DN MUST NOT be older
      than a period of time (i.e., the TCN-Ack validity period) that is
      defined in the PRM requeriments document referenced in
      Specification 7 of the gTLD Applicant Guidebook
      [ICANN-GTLD-AGB-20120604].

   o Performing the minimum set of checks Registries MUST verify that:

      1. The Claims Notice Identifier, expiration datetime and
          acceptance datetime of the Trademark Notice, have been
          received from the Registrar along with the DN registration.

              If the three elements mentioned above are not provided by
              the Registrar for a DN matching a DNL of a PRM, but the DNL
              was inserted (or re-inserted) for the first time into DNL
              list less than 24 hours ago, the registration MAY continue
              without this data and the tests listed below are not
              required to be performed. (Note however, that such DN
              registrations MUST be included in the LORDN.)

      2. The claim notice has not expired (according to the expiration
          datetime sent by the Registrar).

      3. The acceptance datetime is no more than 48 hours in the past.

4. Using the leftmost DNS label (A-label in the case of IDNs) of
    the DN being registered, the expiration datetime provided by
    the registrar and the TMDB identifier extracted from the
    Claims Notice identifier provided by the registrar compute the
    checksum. Verify using that the checksum match the checksum
    present in the Claims Notice identifier.

These procedures apply to all DN registrations at the second level as
well as to all other levels subordinate to the TLD that the Registry
accepts registrations for.

5.3.3. Trademark Claims Services for Registries

5.3.3.1. Domain Name Label (DNL) List

A new DNL list MUST be published by the TMDB twice a day, by 00:00
and 12:00 UTC.

Registries MUST download the latest version of the DNL list at least
once every 24 hours.

Update DNL list

```
    .----------.                           .------.
    | Registry |                           | TMDB |
    '----------'                           '------'
         |                                     |
  .----------------.                           |
  |  Periodically, |                           |
  |    at least    |                           |
  | every 24 hours |                           |
  '----------------'                           |
         |                                     |
         |------------------------------------>|
         |   Download latest list of DNLs      |
         |<------------------------------------|
         |                                     |
         |                                     |
```

Figure 8

5.3.3.2. Notice of Registered Domain Names

The NORDN process during the Trademark Claims Period is almost the
same as during Sunrise Period as defined in Section 5.2.3.3 with the
difference that only registrations subject to a Trademark Claim
(i.e., at registration time the name appeared in the current DNL list
downloaded by the Registry) are included in the LORDN.

5.3.4. DN Registration by Registrars

For any DN matching a DNL of a PRM, Registrars MUST perform the
following steps:

1. Use the Lookup Key received from the Registry (while verifying DN
    availability) to obtain the Claims Notice from the TMDB using the
    dr interface (Section 4.3.6)

2. Present the Claims Notice to the Registrant as described in
    [ICANN-GTLD-AGB-20120604].

3. Ask Registrant for acknowledgement, i.e. the Registrant MUST
    consent with the Claims Notice, before any further processing.
    (The transmission of a Trademark Claims Identifier to the
    Registry over the ry interface, Section 4.3.5 implies that the
    Registrant has expressed his consent with the Claims Notice.)

4. Perform the minimum set of checks for verifying DN registrations.
    If any of these checks fails the Registrar MUST abort the DN
    registration. Each of these checks MUST be performed before the
    registration is sent to the Registry. Performing the minimum set
    of checks Registrars MUST verify that:

    1. The claim notice has not expired (using expiration date
        (<tmNotice:notAfter>).

    2. The claim notice validity based on the (<tmNotice:
        notBefore>).

    3. The leftmost DNS label (A-label in case of IDNs) of the DN
        being registered matches the label (<tmNotice:label>) element
        in the Trademark Claim Notice.

    4. The Registrant has acknowledged (expressed his consent with)
        the Claims Notice.

5. Record the date and time when the registrant acknowledged the
    trademark claims notice.

6. Send the registration to the Registry (ry interface,
    Section 4.3.5) and include the following information:

    *   Trademark Claims Identifier (<tmNotice:id>)

    *   Expiration date of the claims notice (<tmNotice:notAfter>)

    *   Acceptance datetime of the trademark claims notice.

Claims notices are generated twice a day. The expiration date of the
claims notice is set to 48 hours in the future, allowing the
implementation of a cache by Registrars and enough time for the
registration workflow process to finalize. Registrars SHOULD
implement a cache of claims notices to minimize the number of queries
sent to the TMDB. The cached TM Notice must be removed from the
cache after the expiration date of the TM notice as defined by
<tmNotice:notBefore>. The TMDB MAY implement rate-limiting as one of

the protection mechanisms to mitigate the risk of performance
degradation.

5.3.5. Tradermark Claims Services for Registrars

5.3.5.1. Claims Notice Information Service

The Trademark Claims Notices are provided by the TMDB online and are
fetched by the Registrar via the dr interface (Section 4.3.6). To
get access to the Trademark Claims Notices, the Registrar needs the
credentials provided by the TMDB (Section 5.1.2.1) and the Lookup Key
received from the Registry via the ry interface (Section 4.3.5). The
dr interface (Section 4.3.6) uses HTTPS with Basic access
authentication. Furthermore the Registrar has to ensure that all IP
addresses used by his own systems on dr interface have been informed
to the TMDB to be entered into the list known IP Addresses
(Section 5.1.2.4).

The URL of the dr interface (Section 4.3.6) is:
<https://<tmdb-domain-name>/cnis/<lookupkey>.xml>

Note that the "lookupkey" may contain SLASH characters ("/").

The TLS certificate (HTTPS) used on the dr interface (Section 4.3.6)
MUST be signed by a well-know public CA. Registrars MUST perform
standard validation of the TLS (HTTPS) certificate. Registrars are
authenticated in the dr interface using HTTP Basic access
authentication. The dr (Section 4.3.6) interface MUST support HTTPS
keep-alive and MUST maintain the connection for up to 30 minutes.

6. Data Format Descriptions

6.1. DNL List file

   This section defines format of the list containing every Domain Name
   Label (DNL) that matches a Pre-Registered Mark (PRM). The list is
   maintained by the TMDB and downloaded by Registries in regular
   intervals (see Section 5.3.3.1). The Registries use the DNL list
   during the Trademark Claims period to check whether a requested DN
   matched a DNL of a PRM.

   The DNL list is contained in a CSV-like formatted file that has the
   following structure:

   o first line: <version>,<DNL list creation datetime>

         Where:

         + <version>, version of the file. Version 1 MUST always be
            present.

         + <DNL list creation datetime>, date and time that the DNL was
            created.

   o second line: a header line as specified in [RFC4180]

         With the header names as follows:

         DNL,lookup-key,insertion-datetime

   o One or more lines with: <DNL>,<lookup key>,<DNL insertion
      datetime>

         Where:

         + <DNL>, a domain label covered by a trademark.

         + <lookup key>, lookup key that the registry MUST provide to
            the registrar.

         + <DNL insertion datetime>, datetime when the DNL was first
            inserted into the DNL. The possible two values of time for
            inserting a DNL to the DNL list are 00:00 and 12:00.

Example for DNL list


```
1,2012-08-16T00:00:00.0Z
DNL,lookup-key,insertion-datetime
example,j8f/KR6/Dex/dac5KfYddfEr,2010-07-14T00:00:00.0Z
another-example,lj3/l4F/33F/f4HGgg42l5Ts,2012-08-16T00:00:00.0Z
anotherexample,h7h/nm9/FEJ/iHH82mkj3d2c,2011-08-16T12:00:00.0Z
```


                              Figure 9

To provide authentication and integrity protection, the DNL list will
be PGP [RFC4880] signed by the TMDB with the private key of the TMDB
(see also Section 5.1.1.3). The PGP signature of the DNL list can be
found in the similar URI but with extension .sig as shown below.

The URL of the dy interface (Section 4.3.3) is:

o <https://<tmdb-domain-name>/dnl/dnl-latest.csv>

o <https://<tmdb-domain-name>/dnl/dnl-latest.sig>

6.2. SMD revocation list

   This section defines format of the list of SMDs that have been
   revoked. The list is maintained by the SMDM and downloaded by
   Registries (and optionally by Registrars) in regular intervals (see
   Section 5.2.3.1). The SMD revocation list is used during the Sunrise
   Period to validate SMDs received. The SMD revocation list has a
   similar function as CRLs used in PKI [RFC5280] / [RFC6818].

   The SMD revocation list is contained in a CSV-like formatted file
   that has the following structure:

   o first line: <version>,<SMD revokation list creation datetime>

         Where:

         + <version>, version of the file. Version 1 MUST always be
            present.

         + <SMD revokation list creation datetime>, datetime that the
            SMD revokation list was created.

   o second line: a header line as specified in [RFC4180]

         With the header names as follows:

         smd-id

   o One or more lines with: <smd-id>

         Where:

         + <smd-id>, identifier of the SMD that was revoked.

   To provide integrity protection, the SMD revocation list is PGP
   [RFC4880] signed by the TMDB with the private key of the TMDB (see
   also Section 5.1.1.3). The SMD revocation list is provided by the
   TMDB with extension .csv. The PGP signature of the SMD revocation
   list can be found in the similar URI but with extension .sig as shown
   below.

   The URL of the sr interface (Section 4.3.12) and sy interface
   (Section 4.3.11) is:

   o <https://<tmdb-domain-name>/smdrl/smdrl-latest.csv>

   o <https://<tmdb-domain-name>/smdrl/smdrl-latest.sig>

Example for SMD Revocation list


```
1,2012-08-16T00:00:00.0Z
smd-id
2-2
3-2
1-2
```


Figure 10

6.3. **LORDN File**

   This section defines the format of the List of Registered Domain
   Names (LORDN), which is maintained by each Registry and uploaded
   daily to the TMDB. Every time a DN matching a DNL of a PRL said DN
   is added to the LORDN along with further information related to its
   registration.

   The URI of the dy interface (Section 4.3.3) used to upload the LORDN
   file is:

   o https://<tmdb-domain-name>/LORDN/<TLD>/<date>

   o For example, to upload the LORN file for 2012-08-15 and TLD
      "example", the URL would be
      https://<tmdb-domain-name>/LORDN/example/2012-08-15

   The LORDN is contained in a CSV-like formatted file that has the
   following structure:

   o For Sunrise Period:

      * first line: <version>,<LORDN creation datetime>,<date of
         registration>,<type of LORDN>,<Number of lines>

            Where:

            - <version>, version of the file. Version 1 MUST always be
               present.

            - <LORDN creation datetime>, date and time that the LORDN
               was created.

            - <date of registration>, day when the domains were
               registered.

            - <type of LORDN>: Sunrise

            - <Number of lines>, number of domain names registered

      * second line: a header line as specified in [RFC4180]

            With the header names as follows:

            roid,domain-name,SMD-id,registrar-id,application-
            datetime,registration-datetime

* One or more lines with: <roid>,<DN registered>,<smd-id>, <IANA
  registrar id>,<datetime of application creation>,<datetime of
  registration>

    Where:

    - <roid>, Repository Object IDentifier (ROID) in the SRS.

    - <DN registered>, domain name that was registered.

    - <SMD-id>, SMD ID used when the domain was registered.

    - <IANA registrar ID>, IANA registrar ID.

    - OPTIONAL <datetime of application creation>, date and
      time that the application was created, in case of a
      domain registration based on an asynchronous registration
      (e.g., when using auctions).

    - <datetime of registration>, date and time that the domain
      was registered.

  Example for LORDN during Sunrise

```
1,2012-08-16T00:00:00.0Z,2012-08-15,Sunrise,3
roid,domain-name,SMD-id,registrar-id,application-datetime\
   registration-datetime,
SH8013-REP,example1.gtld,1-2,9999,2012-07-15T00:50:00.0Z,\
   2012-08-15T13:20:00.0Z
EK77-REP,example2.gtld,2-2,9999,,2012-08-15T14:00:03.0Z
HB800-REP,example3.gtld,3-2,9999,,2012-08-15T15:40:00.0Z
```

                           Figure 11

o For Trademark Claims Period:

   * first line: <version>,<LORDN creation datetime>,<date of
      registrations>,<type of LORDN>,<Number of lines>

        Where:

        - <version>, version of the file. Version 1 MUST always be
          present.

        - <LORDN creation datetime>, date and time that the LORDN
          was created.

- <date of registrations>, day when the domains were
  registered.

- <type of LORDN>: Claims

- <Number of lines>, number of domain names registered

* second line: a header line as specified in [RFC4180]

   With the header names as follows:

   roid,domain-name,notice-id,registrar-id,application-
   datetime,registration-datetime,ack-datetime

* One or more lines with: <roid>,<DN registered>,<TM notice ID>,<
  IANA registrar id>,<datetime of application creation>,<datetime
  of registration>, <datetime of acceptance of the TM notice>

   Where:

   - <roid>, Repository Object IDentifier (ROID) in the SRS.

   - <DN registered>, domain name that was registered.

   - <TM notice ID>, trademark notice identifier as specified
     in <tmNotice:id>.

   - <IANA registrar ID>, IANA registrar ID.

   - OPTIONAL <datetime of application creation>, date and
     time that the application was created, in case of a
     domain registration based on an asynchronous registration
     (e.g., when using auctions).

   - <datetime of registration>, date and time that the domain
     was registered.

   - <datetime of acceptance of the TM notice>, date and time
     that the TM notice was acknowledged.

   For a DN matching a DNL of a PRM at the moment of
   registration, created without the Claims Notice Identifier,
   expiration datetime and acceptance datetime, becauste DNL
   was inserted (or re-inserted) for the first time into DNL
   list less than 24 hours ago, the string "recent-dnl-
   insertion" MUST be specified in <TM notice ID> and <datetime
   of acceptance of the TM notice>.

Example for LORDN during Claims


```
1,2012-08-16T00:00:00.0Z,2012-08-15,Claims,3
roid,domain-name,notice-id,registrar-id,application-datetime,\
    registration-datetime,ack-datetime
SH8013-REP,example1.gtld,a76716ed9223352036854775808,\
    9999,,2012-08-15T14:20:00.0Z,2012-08-15T13:20:00.0Z
EK77-REP,example2.gtld,a7b786ed9223372036856775808,\
    9999,,2012-08-15T11:20:00.0Z,2012-08-15T11:19:00.0Z
HB800-REP,example3.gtld,recent-dnl-insertion,\
    9999,,2012-08-15T13:20:00.0Z,recent-dnl-insertion
```


Figure 12

6.3.1. LORDN Log File

After reception of the LORDN File, the TMDB verifies its content for
syntactical and semantical correctness. The ouput of the LORDN File
verification is retrieved using the dy interface (Section 4.3.3).

The URI of the dy interface (Section 4.3.3) used to retrieve the
LORDN Log File is:

o https://<tmdb-domain-name>/LORDN/<TLD>/<date>/result

o For example, to obtain the LORN Log File for 2012-08-15 and TLD
   "example" the URI would be
   https://<tmdb-domain-name>/LORDN/example/2012-08-15/result

The LORDN log file is contained in a CSV-like formatted file that has
the following structure:

o first line: <version>,<LORDN Log creation datetime><LORDN file
   creation datetime>, <date of registration>,<type of LORDN>,<Status
   flag>

     Where:

     + <version>, version of the file. Version 1 MUST always be
        present.

     + <LORDN Log creation datetime>, date and time that the LORDN
        Log was created.

     + <LORDN file creation datetime>, date and time of creationg
        for the LORDN file that this log file is referring to.

+ <date of registrations>, day when the domains were
registered as indicated in the related LORDN file.

+ <type of LORDN>, Sunrise or Claims.

+ <Status flag>, whether the LORDN file has been accepted for
processing by the TMDB. Possible values are "accepted" or
"rejected".

o   second line: a header line as specified in [RFC4180]

With the header names as follows:

roid,result-code

o   One or more lines with: <roid>,<result code>

Where:

+ <roid>, Repository Object IDentifier (ROID) in the SRS.

+ <result coded>, return code as described in Section 6.3.1.1.

Example for LORDN Logfile


```
1.0,2012-08-16T02:15:00.0Z,2012-08-16T00:00:00.0Z,\
    2012-08-15,Claims,accepted
roid,result-code
SH8013-REP,2000
```


Figure 13

6.3.1.1. **LORDN Return Codes**

   In Figure 14 the classes of return codes (rc) are listed. Those
   classes in square brackets are not used at this time, but may come
   into use at some later stage. The first two digits of a return code
   denote the return code class, which defines the outcome at the TMDB:

   o ok: Success, domain name line accepted

   o warn: a warning is issued, domain name line accepted

   o err: an error is issued, LORDN file rejected

   LORDN Return Code Classes

   code Class   outcome
   ---- -----   -------

   20xx Success   ok

   33xx [header line syntax warning]   warn
   34xx [header line semantic warning]   warn
   35xx [domain name line syntax warning]   warn
   36xx domain name line semantic warning   warn

   43xx header information syntax error   err
   44xx header information semantic error   err
   45xx domain name line syntax error   err
   46xx domain name line semantic error   err

                           Figure 14

   In the following the LORDN return codes used by the TMDB are
   described:

   LORDN Return Codes

rc   Short Decription
      Long Description
---- ---

2000 **OK**
      Domain Name line successfully processed

3601 **TCN Acknowledgement Date after Registration Date**

TCN Acknowledgement Date in Domain Name is newer than the
Registration Date

3602 **Duplicate DN Line**
This DN Line is an exact duplicate of another line in same
file, DN line ignored

3603 **ROID Notified Earlier**
Same ROID has been notified earlier, DN line ignored

3604 **Checksum Invalid**
Based on the DN registered, the TMC-ID and
the expiration date of the linked TM notice, the
checksum is invalid.

3605 **TMC-ID Expired**
Trademark Claim Notice Identifier was expired
when accepted, but still accepted.
In case of an asynchronous registration, this may refer
to the date of the application creation.

3606 **Wrong TMC-ID used**
The TMC-ID used for the registration was not valid
for this DN.

3607 **SMD-Validation too old**
The validation of the SMD was done outside of the SMD-validation
validity period, and still was used for registration.

3608 **TCN-Acknowledgement too old**
The Acknowledgement of the TCN was done outside of the TCN-Ack
validity period, and still was used for registration.

3609 **Invalid SMD used**
The SMD used for registration was not valid. In case of
an asynchronous registration, this may refer to the date of
the application creation.

4301 **Syntax Error in Header**
Syntac Error in Header information.

4401 **Domain Name Count Mismatch**
The number of Domain Names in Header does not match
the count of DN lines

4402 **Creation Date in past or future**
Creation Date of LORDN is out of range

4403 **Registration Date in past or future**
      Registration date of LORDN is out of range

4404 **Sunrise/Claims mismatch**
      The LORDN is for Sunrise, but the TLD is marked in the TMDB
      as being in Claims or viceversa

4501 **Syntax Error in DN line**
      Syntax Error in Domain Name line

4601 **Invalid TLD used**
      The TLD in the DN line does not match what is expected for this LORDN

4602 **Registrar ID Invalid**
      Registrar ID in DN line is not valid

4603 **Registration Date out of range**
      Registration Date in the DN line is in the past or the future

                        Figure 15

6.4. **SMD File**

   This section defines the format of the SMD File. After a successful
   registration of a mark, the TMV returns an SMD File to the TMH. The
   SMD file can then be used for registration of one or more DNs covered
   by the PRM during the Sunrise Period of a TLD.

   Two encapsulation boundaries are defined for delimiting the
   encapsulated base64 encoded SMD, i.e. "-----BEGIN ENCODED SMD-----"
   and "-----END ENCODED SMD-----". Only data inside the encapsulation
   boundaries MUST be used by Registries and Registrars for validation
   purposes, i.e. any data outside these boundaries as well as the
   boundaries themselves MUST be ignored for validation purposes.

   The structure of the SMD File is as follows:

   o Marks: <marks>

   o smdID: <smd-ID>

   o U-labels: <comma separated list of U-labels>

   o notBefore: <begin validity>

   o notAfter: <end validity>

   o -----BEGIN ENCODED SMD-----

   o <encoded SMD (see [I-D.lozano-tmch-smd])>

   o -----END ENCODED SMD-----

Example for SMD File (shortened at [...]):

Marks: Example One
smdID: 1-2
U-labels: example-one, exampleone
notBefore: 2011-08-16 09:00
notAfter: 2012-08-16 09:00
-----BEGIN ENCODED SMD-----
PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4KPHNtZDpzaWdu
ZWRNYXJrIHhtbG5zOnNtZD0idXJuOmlldGY6cGFyYW1zOnhtbDpuczpzaWduZWRN
YXJrLTEuMCIgaWQ9InNpZ25lZE1hcmsiPgogIDxzbWQ6aWQ+MS0yPC9zbWQ6aWQ+
CiAgPHNtZDppc3N1ZXJJbmZvIGlzc3VlcklEPSIyIj4KICAgIDxzbWQ6b3JnPkV4
YW1wbGUgSW5jLjwvc21kOm9yZz4KICAgIDxzbWQ6ZW1haWw+c3VwcG9ydEBleGFt
cGxlLnRsZDwvc21kOmVtYWlsPgogICAgPHNtZDp1cmw+aHR0cDovL3d3dy5leGFt
cGxlLnRsZDwvc21kOnVybD4KICAgIDxzbWQ6dm9pY2UgeD0iMTIzNCI+KzEuNzAz
NTU1NTU1NTwvc21kOnZvaWNlPgogIDwvc21kOmlzc3VlckluZm8+CiAgPHNtZDpu
b3RCZWZvcmU+MjAwOS0wOC0xNlQwOTowMDowMC4wWjwvc21kOm5vdEJlZm9yZT4K
ICA8c21kOm5vdEFmdGVyPjIwMTAtMDgtMTZUMDk6MDA6MDAuMFo8L3NtZDpub3RB
ZnRlcj4KICA8bWFyazptYXJrIHhtbG5zOm1hcms9InVybjppppZXRmOnBhcmFtczp4
[...]
UFMxOWw3REJLcmJ3YnpBZWEvMGpLV1Z6cnZtVjdUUmZzqeEQzQVFvMVIKYlU1ZEJy
NklqYmRMRmxuTzV4MEEcwbXJHN3g1T1VQdXVyaWh5aVVScEZEcHdIIOEtBSDF3TWND
cFhhWEZSdEdLawp3eWRneVZZQXR5N290a2wvejNiWmtDVlQzNGdQdkkY3MHNSNitR
eFV5OHUwTHpGNUEvYmVZYVpwweFNZRzMxYW1CkFkWGl0VFdGaXBhSUdlYTlsRUdG
TTBMOStCZzdYek5uNG5WTFhva3lFQjNiZ1M0c2NHNlF6blgyM0ZHazwvWDUwOUNl
cnRpZmljYXRlPgo8L1g1MDlEYXRhPgogICAgPC9LZXlJbmZvPgogIDwvU2lnbmF0
dXJlPgo8L3NtZDpzaWduZWRNYXJrPgo=
-----END ENCODED SMD-----

                            Figure 16

6.5. Trademark Claims Notice

   The TMDB MUST provide the Trademark Claim Notice to Registrars in XML
   format as specified below.

   An enclosing element <tmNotice:notice> that describes the Trademark
   Notice to a given label.

   The child elements of the <tmNotice:notice> element include:

   o A <tmNotice:id> element that contains the unique identifier of the
      Trademark Notice.

          The Trademark Notice identifier is a concatenation of a
          checksum and the TMDB identifier. The first 8 characters of
          the Trademark Notice identifier is a checksum. The rest is the
          TMDB identifier, which is a zero-padded number in the range of
          1 to 9223372036854775808.

          Example of a Trademark Notice identifier
          a7b216ed9223372036854775808.

          Where:

          + Checksum=a7b216ed

          + TMDB identifier=9223372036854775808

          The checksum is a 8 characters long Base16 encoded output of
          computing the CRC32 of the string concatenation of: label +
          unix_timestamp(<tmNotice:notAfter>) + TMDB identifier

          The TMDB uses the <tmNotice:label> and <tmNotice:notAfter>
          elements from the Trademark Notice along with the TMDB
          identifier to compute the checksum.

          A registry MUST use the leftmost DNS label (A-label in case of
          IDNs) of the DN being registered, the expiration datetime of
          the Trademark Notice and the TMDB identifier extracted from the
          Trademark Notice identifier provided by the Registrar to
          compute the checksum. For example the DN "foo.bar.example"
          being registered, the left most label would be "foo".

          Example of computation of the checksum: CRC32(example-
          one12819492009223372036854775808)=a7b216ed

   o A <tmNotice:notBefore> element that contains the start of the
      validity date and time of the TM notice.

   o A <tmNotice:notAfter> element that contains the expiration date
      and time of the TM notice.

   o A <tmNotice:label> elements that contain the DNS label (A-label in
      case of IDNs) form of the label that correspond to the DN covered
      by a PRM.

   o One or more <tmNotice:claim> elements that contain the claim. The
      <tmNotice:claim> element contains the following child elements:

      * A <tmNotice:markName> element that contains the mark text
         string.

      * One or more <tmNotice:holder> elements that contains the
         information of the holder of the mark. An "entitlement"
         attribute is used to identify the entitlement of the holder,
         possible values are: owner, assignee and licensee.

      * Zero or more OPTIONAL <tmNotice:contact> elements that contains
         the information of the representative of the mark registration.
         A "type" attribute is used to identify the type of contact,
         possible values are: owner, agent or thirdparty.

      * A <tmNotice:jurDesc> element that contains the name (in
         English) of the jurisdiciton where the trademark is protected.
         A jurCC attribute contains the two-character code of the
         jurisdiction where the trademark was registered. This is a
         two-character code from [WIPO.ST3].

      * Zero or more OPTIONAL <tmNotice:classDesc> element that
         contains the description (in English) of the Nice
         Classification as defined in [WIPO-NICE-CLASSES]. A classNum
         attribute contains the class number.

      * A <tmNotice:goodsAndServices> element that contains the full
         description of the goods and services mentioned in the mark
         registration document.

Example of <tmNotice:notice> object:

```
<?xml version="1.0" encoding="UTF-8"?>
<tmNotice:notice xmlns:tmNotice="urn:ietf:params:xml:ns:tm-notice-1.0">
  <tmNotice:id>a7b216ed9223372036854775808</tmNotice:id>
  <tmNotice:notBefore>2010-08-14T09:00:00.0Z</tmNotice:notBefore>
  <tmNotice:notAfter>2010-08-16T09:00:00.0Z</tmNotice:notAfter>
  <tmNotice:label>example-one</tmNotice:label>
  <tmNotice:claim>
    <tmNotice:markName>Example One</tmNotice:markName>
    <tmNotice:holder entitlement="owner">
      <tmNotice:org>Example Inc.</tmNotice:org>
      <tmNotice:addr>
        <tmNotice:street>123 Example Dr.</tmNotice:street>
        <tmNotice:street>Suite 100</tmNotice:street>
        <tmNotice:city>Reston</tmNotice:city>
        <tmNotice:sp>VA</tmNotice:sp>
        <tmNotice:pc>20190</tmNotice:pc>
        <tmNotice:cc>US</tmNotice:cc>
      </tmNotice:addr>
    </tmNotice:holder>
    <tmNotice:contact type="owner">
      <tmNotice:name>Joe Doe</tmNotice:name>
      <tmNotice:org>Example Inc.</tmNotice:org>
      <tmNotice:addr>
        <tmNotice:street>123 Example Dr.</tmNotice:street>
        <tmNotice:street>Suite 100</tmNotice:street>
        <tmNotice:city>Reston</tmNotice:city>
        <tmNotice:sp>VA</tmNotice:sp>
        <tmNotice:pc>20190</tmNotice:pc>
        <tmNotice:cc>US</tmNotice:cc>
      </tmNotice:addr>
      <tmNotice:voice x="4321">+1.7035555555</tmNotice:voice>
      <tmNotice:email>jdoe@example.com</tmNotice:email>
    </tmNotice:contact>
    <tmNotice:jurDesc jurCC="US">UNITED STATES OF AMERICA</tmNotice:jurDes>
    <tmNotice:classDesc classNum="35">
        Advertising; business management; business administration.
    </tmNotice:classDesc>
    <tmNotice:classDesc classNum="36">
        Insurance; financial affairs; monetary affairs; real estate.
    </tmNotice:classDesc>
    <tmNotice:goodsAndServices>
      Bardus populorum circumdabit se cum captiosus populum.
      Smert populorum circumdabit se cum captiosus populum qui eis differimus.
    </tmNotice:goodsAndServices>
  </tmNotice:claim>
```

```
   <tmNotice:claim>
     <tmNotice:markName>Example-One</tmNotice:markName>
     <tmNotice:holder entitlement="owner">
       <tmNotice:org>Example S.A. de C.V.</tmNotice:org>
       <tmNotice:addr>
         <tmNotice:street>Calle conocida #343</tmNotice:street>
         <tmNotice:city>Conocida</tmNotice:city>
         <tmNotice:sp>SP</tmNotice:sp>
         <tmNotice:pc>82140</tmNotice:pc>
         <tmNotice:cc>BR</tmNotice:cc>
       </tmNotice:addr>
     </tmNotice:holder>
     <tmNotice:jurDesc jurCC="BR">BRAZIL</tmNotice:jurDesc>
     <tmNotice:goodsAndServices>
       Bardus populorum circumdabit se cum captiosus populum.
       Smert populorum circumdabit se cum captiosus populum qui eis differimus.
     </tmNotice:goodsAndServices>
   </tmNotice:claim>
</tmNotice:notice>
```

   This example generates a TRADEMARK NOTICE with two mark claims.

   For formal syntax of the Trademark Claims Notice please refer to
   [Section 7.1](#).

[7](#). Formal Syntax

[7.1](#). Trademark Claims Notice

   The schema presented here is for the Trademark Claims Notice.

```
BEGIN
<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="urn:ietf:params:xml:ns:tm-notice-1.0"
  xmlns:tmNotice="urn:ietf:params:xml:ns:tm-notice-1.0"
  xmlns:mark="urn:ietf:params:xml:ns:mark-1.0"
  xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">

  <annotation>
    <documentation>
```

```
      Schema for representing a Trademark Notice.
    </documentation>
  </annotation>
  <import namespace="urn:ietf:params:xml:ns:mark-1.0"
    schemaLocation="mark-1.0.xsd"/>
  <element name="notice" type="tmNotice:noticeType"/>
  <complexType name="holderType">
    <sequence>
      <element name="name" type="token" minOccurs="0"/>
      <element name="org" type="token" minOccurs="0"/>
      <element name="addr" type="tmNotice:addrType"/>
      <element name="voice" type="mark:e164Type" minOccurs="0"/>
      <element name="fax" type="mark:e164Type" minOccurs="0"/>
      <element name="email" type="mark:minTokenType" minOccurs="0"/>
    </sequence>
    <attribute name="entitlement" type="mark:entitlementType"/>
  </complexType>
  <complexType name="noticeType">
    <sequence>
      <element name="id" type="tmNotice:idType"/>
      <element name="notBefore" type="dateTime"/>
      <element name="notAfter" type="dateTime"/>
      <element name="label" type="mark:labelType"/>
      <element name="claim" type="tmNotice:claimType" minOccurs="0"
        maxOccurs="unbounded"/>
    </sequence>
  </complexType>
  <complexType name="claimType">
    <sequence>
      <element name="markName" type="token"/>
      <element name="holder" type="tmNotice:holderType"
        maxOccurs="unbounded"/>
      <element name="contact" type="tmNotice:contactType" minOccurs="0"
        maxOccurs="unbounded"/>
      <element name="jurDesc" type="tmNotice:jurDescType"/>
      <element name="classDesc" type="tmNotice:classDescType"
        minOccurs="0" maxOccurs="unbounded"/>
      <element name="goodsAndServices" type="token"/>
    </sequence>
  </complexType>
  <complexType name="jurDescType">
    <simpleContent>
      <extension base="token">
        <attribute name="jurCC" type="mark:ccType" use="required"/>
      </extension>
    </simpleContent>
  </complexType>
  <complexType name="classDescType">
```

```
      <simpleContent>
        <extension base="token">
          <attribute name="classNum" type="integer" use="required"/>
        </extension>
      </simpleContent>
    </complexType>
    <complexType name="addrType">
      <sequence>
        <element name="street" type="token" minOccurs="1" maxOccurs="3"/>
        <element name="city" type="token"/>
        <element name="sp" type="token" minOccurs="0"/>
        <element name="pc" type="mark:pcType" minOccurs="0"/>
        <element name="cc" type="mark:ccType"/>
      </sequence>
    </complexType>
    <complexType name="contactType">
      <sequence>
        <element name="name" type="token"/>
        <element name="org" type="token" minOccurs="0"/>
        <element name="addr" type="tmNotice:addrType"/>
        <element name="voice" type="mark:e164Type"/>
        <element name="fax" type="mark:e164Type" minOccurs="0"/>
        <element name="email" type="mark:minTokenType"/>
      </sequence>
      <attribute name="type" type="mark:contactTypeType"/>
    </complexType>
    <simpleType name="idType">
      <restriction base="token">
        <pattern value="[a-fA-F0-9]{8}\d{1,19}"/>
      </restriction>
    </simpleType>
</schema>
END
```

8. Status of this Draft

   This draft is the product of several discussions regarding the
   interaction between the TMCH and Registries/Registrars. It does by
   no means claim to be complete and minor updates are likely to be
   added in future versions of this document.

9. Acknowledgements

   TBD

10. IANA Considerations

TBD

11. Security Considerations

TBD

12. References

12.1. Normative References

[I-D.lozano-tmch-smd]
          Lozano, G., "Mark and Signed Mark Objects Mapping",
          draft-lozano-tmch-smd-01 (work in progress), March 2013.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
          Requirement Levels", BCP 14, RFC 2119, March 1997.

12.2. Informative References

[ICANN-GTLD-AGB-20120604]
          ICANN, "gTLD Applicant Guidebook Version 2012-06-04",
          June 2012, <http://newgtlds.icann.org/en/applicants/agb/
          guidebook-full-04jun12-en.pdf>.

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities",
          STD 13, RFC 1034, November 1987.

[RFC1591] Postel, J., "Domain Name System Structure and Delegation",
          RFC 1591, March 1994.

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and
          E. Lear, "Address Allocation for Private Internets",
          BCP 5, RFC 1918, February 1996.

[RFC1952] Deutsch, P., Gailly, J-L., Adler, M., Deutsch, L., and G.
          Randers-Pehrson, "GZIP file format specification version
          4.3", RFC 1952, May 1996.

[RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H.,
          Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext
          Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000.

   [RFC3339]   Klyne, G., Ed. and C. Newman, "Date and Time on the
               Internet: Timestamps", RFC 3339, July 2002.

   [RFC4180]   Shafranovich, Y., "Common Format and MIME Type for Comma-
               Separated Values (CSV) Files", RFC 4180, October 2005.

   [RFC4880]   Callas, J., Donnerhacke, L., Finney, H., Shaw, D., and R.
               Thayer, "OpenPGP Message Format", RFC 4880, November 2007.

   [RFC5280]   Cooper, D., Santesson, S., Farrell, S., Boeyen, S.,
               Housley, R., and W. Polk, "Internet X.509 Public Key
               Infrastructure Certificate and Certificate Revocation List
               (CRL) Profile", RFC 5280, May 2008.

   [RFC5730]   Hollenbeck, S., "Extensible Provisioning Protocol (EPP)",
               STD 69, RFC 5730, August 2009.

   [RFC5890]   Klensin, J., "Internationalized Domain Names for
               Applications (IDNA): Definitions and Document Framework",
               RFC 5890, August 2010.

   [RFC6818]   Yee, P., "Updates to the Internet X.509 Public Key
               Infrastructure Certificate and Certificate Revocation List
               (CRL) Profile", RFC 6818, January 2013.

   [WIPO-NICE-CLASSES]
               WIPO, "Nice List of Classes", <http://www.wipo.int/
               classifications/nivilo/nice/index.htm>.

   [WIPO.ST3]
               WIPO, "Recommended standard on two-letter codes for the
               representation of states, other entities and
               intergovernmental organizations", March 2007.

Appendix A. Document Changelog

   [RFC Editor: This section is to be removed before publication]

   draft-lozano-tmch-functional-spec-00:

   o Initial version

Authors' Addresses

   Gustavo Lozano (editor)
   ICANN
   12025 Waterfront Drive, Suite 300
   Los Angeles 90292
   US

   Phone: +1.3103015800
   Email: gustavo.lozano@icann.org

   Bernie Hoeneisen
   Ucom Standards Track Solutions GmbH
   CH-8000 Zuerich
   Switzerland

   Phone: +41 44 500 52 44
   Email: bernie@ietf.hoeneisen.ch (bernhard.hoeneisen AT ucom.ch)
   URI: http://www.ucom.ch/