Q23_Figure2 - List of Codes throughout the Application

IN OUR RESPONSE TO QUESTIONS 25, 29, 32, 36, 38, 42, 43, AND 44 WE ARE DEMONSTRATING EXAMPLES OF CODE TO ILLUSTRATE HOW OUR REGISTRY SERVICES WORK. OUR CODE CONTAINS THE BRACKETS "<" AND ">". WHEN GENERATING HTML PREVIEW VERSIONS OF AN APPLICATION VIA THE TAS, WE HAVE DETECTED THAT BROWSERS, OR PROGRAMS SUCH AS MICROSOFT WORD AND ADOBE READER, DO NOT SHOW THESE CODES (THEY ARE, HOWEVER, SHOWN IF YOU OPEN THE FILE WITH NOTEPAD OR SIMILAR TEXT EDITORS). HENCE, WE HAVE DECIDED TO ATTACH CODE FOR QUESTIONS 25, 29, 32, 36, 38, 42, 43, AND 44 IN ORDER FOR EVALUATORS TO HAVE AN EASY ACCESS TO ALL THE CODE EXAMPLES.

In the following we list the codes used in our responses.

We added the complete relevant chapters which contain the information stored in brackets and marked the codes with yellow.

Question 25:

B. Extensions used by KSR and Related Internet Drafts

B.1 Draft-tan-epp-launchphase-01 (Launch Phase Mapping for the EPP)

KSR intends to use this EPP internet draft to facilitate Sunrise phases during the initiation of this TLD registry. This internet draft proposes an extension mechanism that supports the organization of Sunrise related domain applications. The extension considers the following elements:

<lp:phase>
This element allows a Sunrise application submission to be marked by the EPP client as a particular Sunrise application type, in respect to running different types of Sunrise applications during a concurrent submission period. KSR will use this to identify Sunrise A and Sunrise B application types.

<lp:status>
This element allows the EPP server to assign one of a number of statuses indicating what stage the Sunrise application is in. These statuses can be expressed through the domaininfo command response and, optionally, through the RDDS service if applicable. The statuses listed below can be assigned uniquely or in combinations where appropriate:

<pvrc>
The Pre-Validation Result Code, an opaque string issued by a third-party validation agent

<claimIssuer>
contains the ID of a contact object (as described in RFC 5733 [RFC5733]) identifying the contact information of the authority which issued the right (for example, a trade mark office or company registration bureau)

<claimName>
identifying the text string in which the applicant is claiming a prior right

<claimNumber>
the registration number of the right (ie trademark number or company registration number)

<claimType>
indicates the type of claim being made (eg trademark, symbol, combined mark, company name)

<claimEntitlement>
indicates the applicant's entitlement to the claim (ie, owner or licensee)

<claimRegDate>
the date of registration of the claim

<claimExDate>
the date of expiration of the claim

<claimCountry>
indicates the country in which the claim is valid

<claimRegion>
indicates the name of a city, state, province or other geographic region in which the claim is valid. This may be a two-character code from [WIPO.ST3]

The complete draft is described in attachment Q25_Figure3.pdf.

B.2 Draft-obispo-epp-idn-00 (Internationalized Domain Name Mapping Extension for the EPP)

KSR intends to use this EPP internet draft to facilitate the usage of provisioning Internationalized Domain Names (IDNs). This internet draft extends the EPP domain name mapping to provide additional features that are required to implement registrations of domain names in character sets other than ASCII.

The extension considers the following element in both the domain create request and the domain-info response:

<idn:language>
This element allows for association of a domain name to a language tag, as defined in the code division of the Unicode code chart.

This element allows the registrar submitting the registration to identify a language tag associated with a punycode registration. The language tag refers the server to consider a declared set of table- and/or algorithmic-driven policies regarding a set and/or combination of defined unicode points, including variations of the punycode registration.
Insert Attachment with entire RFC including full xml schema examples

The complete draft is described in attachment Q25_Figure4.pdf.

Question 29:

D.1 Field Submission and Validation Safeguards

To support the Sunrise process the registry system will apply requirements to data submissions, in the registry application. KSregistry will use the data fields/extensions from the Dot Asia and Dot XXX launches. The field types listed below are mandatory during Sunrise submissions and are not able to be updated once an application has been submitted in order to avoid manipulation of the validation process. Please see the related EPP extension <Draft-tan-epp-launchphase-01> attached to the Question 25 response.

<claimName>

The word mark of the trademark, as noted in the trademark or service mark registration record.

<claimNumber>

The registration number (not the application number) of the trademark.

<claimRegDate>

The date the trademark was registered. This is the date that the trademark was granted.

<claimCountry>, [<claimRegion>]

The two-letter code for the nation or national jurisdiction the trademark was registered in. Allowable values will be limited to identify national trademark registries and regional trademark registries (such as the Benelux and EU trademark registry registrations).

<claimEntitlement>

Whether the applicant (corresponding to the Registrant Contact) holds the trademark as the original "owner," "co-owner," or "assignee".

Question 32:

I. TCP/IP Scheme
The communication to the KSregistry will be based on connections to <service>.ksregistry.net which will be available via:

- 109.234.109.0/24 for all services in St. Ingbert
- 95.130.21.0/24 for all services in Munich

Question 36:

C. Public IPv6 Transport for Registration Data Publication Services (RA, Specification 4)

Until ICANN requires a different protocol, the registry operator will operate a WHOIS service available via port 43 in accordance with RFC 3912 and a web-based directory service at <whois.nic.TLD> providing free public query-based access to at least the following elements in the format later described.

- WHOIS (RFC 3912)
- Web based Whois

Question 38:

A. Data Escrow

A.1 Format

Registry objects (domains, contacts, hosts, registrars) will be compiled into a UTF-8 encoded file constructed as described in draft-arias-noguchi-registry-data-escrow (http://tools.ietf.org/html/draft-arias-noguchi-registry-data-escrow-02).

Example:

A domain object with the name “domain.tld” will be represented by:

```
<rdeDomain:domain>
	<rdeDomain:name>domain.tld</rdeDomain:name>
	<rdeDomain:roid>3903990019997_DOMAIN-KEYSYS</rdeDomain:roid>
	<rdeDomain:registrant>P-MHM21</rdeDomain:registrant>
	<rdeDomain:contact type="admin">P-MHM21</rdeDomain:contact>
	<rdeDomain:contact type="tech">P-MHM21</rdeDomain:contact>
	<rdeDomain:contact type="billing">P-MHM21</rdeDomain:contact>
	<rdeDomain:ns/>
	<rdeDomain:clID>cebit2011</rdeDomain:clID>
	<rdeDomain:crID>cebit2011</rdeDomain:crID>
	<rdeDomain:crDate>2011-03-03T09:57:29Z</rdeDomain:crDate>
	<rdeDomain:upID>cebit2011</rdeDomain:upID>
	<rdeDomain:upDate>2011-03-03T09:57:29Z</rdeDomain:upDate>
	<rdeDomain:exDate>2012-03-03T09:57:29Z</rdeDomain:exDate>
	<rdeDomain:authInfo>
		<domain:pw>xjnag2CqO4</domain:pw>
	</rdeDomain:authInfo>
	<rdeDomain:dnssec>
		<secDNS:maxSigLife>604800</secDNS:maxSigLife>
	</rdeDomain:dnssec>
</rdeDomain:domain>
```

A contact object with the id “P-DEM21” will be represented by:

```
<rdeContact:contact>
	<rdeContact:id>P-DEM21</rdeContact:id>
```

```
        <rdeContact:postalInfo>
                <contact:name>Domain Manager</contact:name>
                <contact:org>TEST AG</contact:org>
                <contact:addr>
                        <contact:street>Musterstrasse 1</contact:street>
                        <contact:city>Musterstadt</contact:city>
                        <contact:sp/>
                        <contact:pc>12345</contact:pc>
                        <contact:cc>DE</contact:cc>
                </contact:addr>
        </rdeContact:postalInfo>
        <rdeContact:voice>+49.987654321</rdeContact:voice>
        <rdeContact:fax/>
        <rdeContact:email>test@domain.tld</rdeContact:email>
        <rdeContact:clID>test_doma76</rdeContact:clID>
        <rdeContact:crID>test_doma76</rdeContact:crID>
        <rdeContact:crDate>2011-08-17 07:33:53</rdeContact:crDate>
        <rdeContact:upID>test_doma76</rdeContact:upID>
        <rdeContact:upDate>2011-08-17 07:33:53</rdeContact:upDate>
</rdeContact:contact>
```

A host object with the name “ns1.domain.tld” will be represented by:

```
<rdeHost:host>
        <rdeHost:name>ns1.domain.tld</rdeHost:name>
        <rdeHost:roid>sdgsdgsdg</rdeHost:roid>
        <rdeHost:status s="pendingDelete"/>
        <rdeHost:addr ip="v4">1.2.3.4</rdeHost:addr>
</rdeHost:host>
```

A registrar object with the id “testregistrar” will be represented by:

```
<rdeRegistrar:registrar>
        <rdeRegistrar:id>testregistrar</rdeRegistrar:id>
        <rdeRegistrar:roid>sdgsdgsdg</rdeRegistrar:roid>
        <rdeRegistrar:postalInfo>
                <contact:name>Roadshow User</contact:name>
                <contact:org>Test Registrar</contact:org>
                <contact:addr>
                        <contact:street>Im Oberen Werk 1</contact:street>
                        <contact:city>St. Ingbert</contact:city>
                        <contact:sp>Saarland</contact:sp>
                        <contact:pc>66386</contact:pc>
                        <contact:cc>DE</contact:cc>
                </contact:addr>
        </rdeRegistrar:postalInfo>
        <rdeRegistrar:voice>+49.68949396930</rdeRegistrar:voice>
        <rdeRegistrar:fax/>
        <rdeRegistrar:email>roadshow@key-systems.net</rdeRegistrar:email>
        <rdeRegistrar:contact type="admin">P-123</rdeRegistrar:contact>
        <rdeRegistrar:crDate>2011-08-22T07:10:54Z</rdeRegistrar:crDate>
        <rdeRegistrar:upDate>2011-08-22T08:14:12Z</rdeRegistrar:upDate>
        <rdeRegistrar:authInfo>
                <contact:pw>sdgdggffhd</contact:pw>
        </rdeRegistrar:authInfo>
</rdeRegistrar:registrar>
```

There will be no additional data or extensions.

Question 42:

In Chapter A.2 Service Monitoring:

Case of DNS

To check the behavior of the Domain Name Service basic dns checks are run from all of the monitoring systems. These checks are very similar to a basic dig.

```
dig example.STRING

; <<>> DiG 9.7.3 <<>> example.STRING
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38770
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 4
```

Question 43:

B.3 Registrant Key Material Provisioning

Registrant key material is provisioned via the SRS Extensible Provisioning Protocol (EPP) interface by authenticated users. The SRS implements the EPP DNSSEC Extensions (RFC 5910) with the DS data interface and the key data interface.

Submitted delegation signer (DS) data is stored in the SRS database and included in the zone by the zone file generator. For submitted key data, the DS data is generated on the server with digest type SHA-256 (RFC 4509).

DS data example:

```
<secDNS:dsData>
        <secDNS:keyTag>12345</secDNS:keyTag>
        <secDNS:alg>3</secDNS:alg>
        <secDNS:digestType>1</secDNS:digestType>
        <secDNS:digest>49FD46E6C4B45C55D4AC</secDNS:digest>
</secDNS:dsData>
```

Key data example:

```
<secDNS:keyData>
        <secDNS:flags>257</secDNS:flags>
   <secDNS:protocol>3</secDNS:protocol>
   <secDNS:alg>1</secDNS:alg>
        <secDNS:pubKey>AQPJ////4Q==</secDNS:pubKey>
</secDNS:keyData>
...
```

and in chapter C.2

…

C.2 Key and Signature Policy (KASP)

The parameters for key management are manifested in the “Key and Signature Policy” (KASP) inside OpenDNSSEC.

```
<KASP>
  <Policy name="default">
    <Signatures>
      <Resign>PT2H</Resign>
      <Refresh>P3D</Refresh>
      <Validity>
        <Default>P7D</Default>
        <Denial>P7D</Denial>
      </Validity>
      <Jitter>PT12H</Jitter>
      <InceptionOffset>PT3600S</InceptionOffset>
    </Signatures>
...
    <Keys>
      <!-- Parameters for both KSK and ZSK -->
      <TTL>PT3600S</TTL>

      <!-- Parameters for KSK only -->
      <KSK>
        <Algorithm length="2048">8</Algorithm>
        <Lifetime>P1Y</Lifetime>
        <Repository>SoftHSM</Repository>
      </KSK>

      <!-- Parameters for ZSK only -->
      <ZSK>
        <Algorithm length="1024">8</Algorithm>
        <Lifetime>P30D</Lifetime>
        <Repository>SoftHSM</Repository>
      </ZSK>
    </Keys>
...
  </Policy>
</KASP>
```

Question 44:

G.2 EPP:

The EPP service accepts only A-labels in the domain name section. As described in RFC 5731, EPP supports the following basic commands:

- Check: The check command expects no language tag. It therefore checks only, if the given domain name is available for registration. All language tag specific checks (like the IDN-table check) are not done at this stage, but later by the create domain command. This keeps the check command simple and lightweight which enables an optimal technical performance.
- Info: The info command returns the domain specific language tag. Here is an example of the EPP extension (in accordance with http://tools.ietf.org/html/draft-obispo-epp-idn-00):

```
...
<extension>
       <idn:language xmlns:idn="urn:ietf:params:xml:ns:idn-1.0">en
       </idn:language>
</extension>
...
```

- Transfer: Calling the domain transfer command requires no IDN extension.
- Delete: A delete domain command expects no IDN extension.
- Renew: There is no IDN extension in the renew domain command.
- Update: It is not planned, to modify the language tag of an existing domain. Therefore there is no IDN extension in the update domain command.
- Create: For IDN registrations, a language tag has to be submitted. This is done by an EPP extension. Here is an example of the EPP extension (in accordance with http://tools.ietf.org/html/draft-obispo-epp-idn-00):

```
...
<extension>
  <idn:language xmlns:idn="urn:ietf:params:xml:ns:idn-1.0">en
  </idn:language>
</extension>
...
```