Registrar Operations Manual

Version 3.2.14

February 2022

Copyright 2022 CentralNic Ltd. All Rights Reserved

Table of Contents

1. Version History

Version	Release Date	Changes
3.2.14	2022-02-02	● 50 check objects
3.2.13	2022-01-24	● 100 epp connections ● Small pending changes
3.2.11	2020-09-21	● add some more information around dropcatching ● add more information around special TLD's running in OTE
3.2.9	2020-08-24	● Add example command for adding registry lock with fee extension
3.2.8	2020-07-22	● Update logo ● Move master to google sheets ● EPP Registrar onboarding
3.2.7	2020-04-14	● Minor update to remove serverRenewProhibited when submitting a command for registry lock.
3.2.6	2018-05-14	● Fixed broken cross-reference in EPP session limits section. ● Corrected minimum RTT for domain <create> commands which result in a 2302 response. ● Add note about testing of the Fee extension in OT&E. ● Documented Lifetime Discounts system. ● Updated IDN documentation relating to contextual rules. ● Documented Account Settings JSON API. ● Updated to cover changes due to GDPR: required contacts, deletion of unused contacts, contact disclosure policies, presence of contact information in whois records. ● Documented use of EPP to add registry lock to a domain.
3.2.5	2016-03-21	● Added information about domain purge EPP messages.
3.2.4	2015-10-29	● Added documentation about EPP messages sent when. domains on pendingCreate are approved or rejected.

		- Updated information on EPP connection limits. - Added guidance on "drop catching" activity.
3.2.3	2015-09-08	- Added note about ensuring registrars have sufficient credit for GA launches. - Added documentation for Account Balance/Credit Limit JSON interface. - Added information about restrictions on contact \<info\> commands. - Added information about restrictions on contact whois queries. - Added documentation of release of names using authInfo codes. - Documented the use of lftp to automate downloads from the FTP system.
3.2.2	2014-09-30	- Added information about promotional pricing. - Updated information about purging of unused contact objects.
3.2.1	2014-08-29	Updated Section 11.11.5 to cover the new version of the fee extension.
3.2.0	2014-06-17	- Added Version History section. - Documented premium domains and variable pricing. - Documented Implementation details of the Fee extension in relation to premium domains. - Documented EPP session limits and timeouts. - Documented contact field sizes

2. About CentralNic

One of the world's pioneering registry service providers, CentralNic provides registry services, distribution, and strategic consultancy for gTLDs, ccTLDs and SLDs.

CentralNic uses our in-house developed IT platform to distribute domain names including to a global network of over 1,500 registrars and 100,000 resellers, who sell them on to end users.

Our powerful registry engine has a track record of uninterrupted service since 1994 and is supported by innovative marketing and personal customer service. Additionally, our flexible and scalable platform is above ICANN specification for new TLDs.

CentralNic operates from our global headquarters in London plus offices in Bonn, New York, Dubai, Melbourne and Napier, New Zealand. We also have on-the-ground account executives in Shanghai, Abu Dhabi and Los Angeles.

3. Introduction

This document has been written with the intention of providing new and existing registrars with all the information they will need to successfully integrate with the CentralNic Registry. This document covers technical, financial and policy considerations involved with integration. All registrars are strongly advised to read this document thoroughly.

If you have any questions regarding this document, please get in touch and we will be very happy to assist you.

- Telephone: +44 (0)20 33 88 0600 (24x7)

- E-mail: registrars@centralnic.com

4. Accreditation

To be granted access to the CentralNic registry system, all registrars must sign the CentralNic Master Access Agreement. You are not required to deposit funds with CentralNic in advance or complete an OT&E evaluation.

Some domain extensions require a separate accreditation agreement. You can sign the agreement and complete the accreditation for these extensions on the Registrar Console.

5. Interfaces

Registrars can interact with the CentralNic registry system using the following interfaces:

- Registrar Console

- EPP Server
- FTP Server

5.1. Registrar Console

The Registrar Console provides a simple way to manage objects in the registry using a web interface. Access to the Registrar Console is via secure web application at

https://registrar-console.centralnic.com/

This console allows registrars to remit payment, manage domain names and transfers, and change account settings.

5.1.1. Access Control

Each registrar may have any number of user accounts associated with it. Registrars should create a separate user account for each employee who accesses the Registrar Console. User accounts may be assigned to one of three roles:

1. **Administrator:** Each registrar account has only one administrator, has full access to all features, and may create, update and delete other user accounts (including Managers)

2. **Manager:** the user has full access to all features, and may create, update and delete other user accounts (excluding Managers and Administrators)

3. **Domain Administrator:** the user has limited access and can only access domain, contact and host management features

4. **Finance Administrator:** the user has limited access and can only access billing, payment and account management features

Two-Factor Authentication (2FA) is supported using any authenticator app which implements Time-Based One Time Password (TOTP), such as Google Authenticator.

Registrars are encouraged to use Two-Factor Authentication on all user accounts. Administrator and Manager accounts MAY enforce use of 2FA by all user accounts from the Account Settings page.

5.2. EPP Server

The EPP server is documented fully in Section EPP system.

5.2.1. Access Control

While access to the Registrar Console is not restricted, connections to the EPP server are restricted by IP address. Registrars must provide a list of IP addresses (or networks) from which they will make connections. This list can be managed online via the Registrar Console.

5.3. FTP Server

An FTP server is provided to allow registrars to download reports and data. A list of the available files is provided in Section 5.3.2.

5.3.1. Access Control

Access to the FTP server must be enabled on your account. Access will also be restricted to hosts that are registered on your IP access list, which may be updated via the Registrar Console.

To ensure the confidentiality and integrity of the data provided by the FTP server, all connections must be secured by TLS (SSL). Registrars must therefore use an FTP client that supports FTPS, such as FileZilla, cURL, wget or lftp.

To connect to the FTP server, use your registrar ID and FTP password (which can be obtained from the "Passwords" page of the Registrar Console) and connect to **registrar-ftp.centralnic.com** on TCP port 21.

Registrars can also access the files on the FTP server through the "FTP File Browser" on the Registrar Console.

5.3.2. Available Files

5.3.2.1. Domain Drop List

Every 24 hours we generate a separate file for each domain extension listing all the domains that have the pendingDelete status. The file is a CSV file with the following fields:

Column	Description

Domain_Name	Domain name.
Date_Entered	Date and time the domain entered the RGP.
Date_Leaves	Date and time the domain will leave the RGP.
Date_Purged	The approximate date and time that the domain will be purged from the database.

These files are stored in the "droplist" directory of the FTP server.

5.3.2.2. List of Unavailable Domain Names

We provide a complete list of all "unavailable" domain names for each domain extension. This list includes:

- Registered names
- Banned names
- Blocked names
- Reserved names

A file is generated for each domain extension every 2 hours. The file is a Gzip-compressed plain-text file with Unix line endings. These files are stored in the "lordn" directory of the FTP server.

5.3.2.3. Pricing Information

Some domain extensions have variable pricing, whereby particular domains may have a non-standard price. We provide a CSV file for each domain extension containing the following fields:

Column	Description
domain	Domain name
type	Transaction type, one of Registration, Renewal, Restore
amount_eur	Price in EUR
amount_gbp	Price in GBP
amount_usd	Price in USD

Note that there may be multiple rows for the same domain representing different transactions, for example:

```
domain,type,amount_eur,amount_gbp,amount_usd
example.tld,Registration,100,85,110
example.tld,Renewal,50,42.5,55
example.tld,Registration,250,150,275
```

These files are stored in the "prices" directory on the FTP server.

5.3.2.4. Billable Transactions

Billable transactions are transactions that have passed the grace period and will as such make it onto an invoice. Registrars can download CSV files detailing all billable transactions against their account for the following periods:

- Daily
- Weekly
- Monthly

Files are in the CSV format with the following columns:

Column	Description
Timestamp	Date and time of the transaction
Domain	Domain Name
Type	Transaction type. One of: Registration, Renewal, Auto-Renewal
Grace Period	Grace period for the transaction in days
Currency	Currency for the transaction
Amount	Amount of the transaction in the above currency

Files are stored in the "transactions" directory of the FTP server. There are three sub-directories:

- "transactions/daily/" contains daily transaction reports. Filenames have the form "YYYY-MM-DD.csv"

- "transactions/weekly/" contains weekly transaction reports. Filename have the form "YYYY-NN.csv", where "NN" is the week number

- "transactions/monthly/" contains monthly transaction reports. Filenames have the form "YYYY-MM.csv"

Files will not be present if no billable transactions occurred during the period in question.

5.3.2.5. Zone Files

If zone file access is enabled for your account, then zone files will be made available in the "zones" directory. Files are Gzip-compressed.

5.3.2.6. Automated FTP Access

The lftp program[1] can be used to automate the downloading of files from the FTP server. The following lftp script provides an example of how the premium pricing for the .DESIGN TLD can be downloaded:

```
set xfer:clobber true
set ftp:ssl-auth TLS
set ftp:ssl-force true
set ftp:ssl-protect-list yes
set ftp:ssl-protect-data yes
connect registrar-ftp.centralnic.com
login REGISTRAR-ID FTP-PASSWORD (replace as applicable)
get prices/design.csv
quit
```

---

[1] Available from http://lftp.yar.ru/

6. Technical Support

A trouble tickets system is integrated into the Registrar Console, and registrars must use this system for all support requests. This allows us to monitor and track support requests and provides an escalation path for urgent issues with the Registry system. A Live Chat system is also available.

Telephone- and email-based technical support is available 24 hours a day, 7 days a week.

In the unlikely event that the Registrar Console is not available, registrars can contact the Operations team at **ops@centralnic.com**.

6.1. Maintenance Downtime

CentralNic may, on occasion, need to take down portions of its registry system for maintenance. This will never affect the resolution of domains already registered. Registrars will receive notification via e-mail well in advance of any planned outage (registrars must specify an "operations" email address for their account in order to receive these notifications).

Additionally, we will notify registrars at the beginning and end of any planned maintenance window, so that registrars are aware when planned maintenance has been completed.

Registrars will also be notified of any unscheduled outages.

7. Billing

CentralNic's billing system is designed to be highly flexible, allowing for multiple billing models in parallel. By default, your account will be set to the Debit Model.

7.1. Available Billing Models

7.2. Debit Model (Prepayment)

This model is the same as that operated by most gTLD and ccTLD registries. Registrars maintain a balance of funds with the registry, from which registration and renewal fees are deducted. Payment may be remitted to the registry by wire transfer or credit card.

Under this model, our system will automatically send an email to your specified billing address if your balance is running low. If your balance reaches zero, then you will have access to an emergency credit balance (the default is £2,000 or the equivalent in your billing currency).

7.3. Credit Model

Under the Credit billing model, each registrar is assigned a credit limit, and may perform billable transactions until this limit is exhausted. Registrars are then invoiced for these billable transactions on a monthly basis. Registrars are required under the terms of their agreement to promptly remit payment for these invoices.

7.4. Renewals

CentralNic operates two models for domain name renewals.

Auto-renew. All domain names are automatically renewed. The "Auto-Renew Grace Period" applies to these renewals (see Section Renewal).

Auto-delete. All domains that are not explicitly renewed by the registrar are automatically placed on "Pending Deletion" when they expire. These domains may be restored using the RGP restore command (see Section Redemption Grace Period).

By default, all new registrar accounts are set to the auto-renew model.

7.5. Invoicing

Regardless of which billing model you choose, our system will generate a regular invoice for registrations and renewals. We will generate a new invoice on the last day of each month for all registrations and renewals that have taken place since the previous invoice.

Registrars may alternatively choose to be invoiced on a daily basis using the Account Settings page of the Registrar Console: this is often desirable for low-volume registrar accounts.

7.6. Submitting Payment

If using the Debit Model, you are requested to notify us of payment using the 'Top Up' facility on the Registrar Console. This will ensure your payment is allocated correctly when it reaches us.

If using the Credit Model, payment can be submitted for outstanding invoices on the Registrar Console. You can then select your preferred payment method, and details of where to submit payment are displayed. Once the remittance details are confirmed. CentralNic is notified and your payment can easily be allocated when it arrives.

Payment can be made by credit card or direct bank transfer. We have Sterling, US Dollar and Euro accounts to reduce costs to our registrars due to exchange commission. Details of our Sterling, US Dollar and Euro accounts are supplied when creating a payment batch on the Registrar Console.

7.7. Premium Domain Name Pricing

Some TLDs operated by CentralNic include domain names that are charged at a premium rate that differs from the base fee for the TLD. These domains are provisioned and managed using the same SRS interfaces, and billed to registrars in the same way, as regular domains.

If a domain name has a higher fee associated with it, then registrars must obtain this fee and include it in any EPP commands to provision and manage this name: this ensures that these domains cannot be accidentally registered or renewed without the premium cost being known to the registrar.

CentralNic provides two interfaces by which premium pricing can be obtained:

- FTP interface (see Section [FTP Server](#))

- EPP fee extension (see Section [Fee Extension](#))

Please consult the policy documentation of each TLD in order to obtain that TLD's policy on premium pricing.

7.8. Promotional Pricing

Registrars may be offered promotional pricing for certain domain extensions. If a billable transaction is subject to a promotional discount, then this discount will be applied at the moment of the transaction, rather than afterwards.

In most circumstances, this is not an issue, but if a domain extension offers variable or premium pricing (as described above), it is important that registrars be able to correctly calculate the discounted fee when submitting fee information via EPP (see Section 11.11.5).

7.9. Preparing your account for General Availability Launches

Many registrars participate in the launch of new domain extensions, by collecting pre-registrations from customers and submitting a large batch of registrations within the first few minutes of General Availability ("GA").

To avoid problems, registrars must ensure that their account has sufficient credit available to cover the fees for all pre-registrations, in addition to their normal transaction volumes, plus any pending applications (e.g. Landrush applications), which may not be processed until shortly before GA.

Note: it can take up to 2 hours for deposits made using a credit or debit card to become available for use via EPP or the Registrar Console. Please ensure that deposits made using cards are submitted at least 4 hours before a launch, and that wire transfers are submitted at least 2 business days before.

7.10. Account Balance/Credit Limit Automated Interface

A registrar account's account balance and/or credit limit may be programmatically retrieved by submitting an HTTP GET request to the following URI:

https://registrar-console.centralnic.com/json/balance

This URI is protected by HTTP authentication: use your registrar ID and EPP password to access it.

The response is a JSON object of the following form:

```
{
    "availableCredit" : "960398.50",
    "balance"         : "-39601.50",
    "creditLimit"     : "1000000.00",
    "creditThreshold" : "200000.00"
}
```

The properties of the object have the following meanings:

- **availableCredit**: the amount of credit, including any credit (emergency or otherwise), still available to fund registrations, renewals, and transfers

- **balance**: the balance available on your account. If this value is negative, any prepayment balance has been exhausted (for prepayment registrars) and billable transactions are being funded by your credit limit.

- **creditLimit**: what your credit limit is.

- **creditThreshold**: the threshold at which we will notify you (by email) that you may wish to deposit additional funds.

All values are in the account's preferred billing currency.

7.11. Account Settings

Registrars can query for their current account settings by submitting an HTTP GET request to the following URI:

https://registrar-console.centralnic.com/json/settings

This URI is protected by HTTP authentication: use your registrar ID and EPP password to access it.

The response is a JSON object of the following form:

```json
{
  "org": {
    "registrar-id": "H12345",
    "name": "Example, Inc",
    "trading-name": null,
    "iana-id": 9999,
    "website": {
      "rel": "website",
      "href": "http://www.example.com/"
    },
    "port43": null,
    "voice": {
      "rel": "voice",
      "href": "tel://+44.123456789"
    },
    "fax": null
  },
  "addr": {
    "int": {
      "street": [
        "123 Example Street",
        "Suite 4"
      ],
      "city": "Exampletown",
      "sp": "Exampleshire",
      "pc": "EX12 3AM",
      "cc": "GB"
    }
  },
  "notificiations": {
    "primary": "info@example.com",
    "billing": "domain.billing@example.com",
    "transfers": "domain.transfers@example.com",
    "operator": "domain.operations@example.com",
    "abuse": "domain.security@example.com",
    "legal": "domain.legal@example.com"
  },
  "billing": {
    "auto-renew-enabled": true,
    "billing-model": "pre-payment",
    "billing-interval": "monthly",
    "billing-currency": "USD",
    "invoice-per-currency": false,
    "balance": {
      "rel": "balance",
      "href":
"https://registrar-console.centralnic.com/json/balance"
    }
```

```
    },
    "epp": {
      "message-queue-enabled": true,
      "password-last-updated": "2017-03-01T14:06:53.0Z",
      "check-options": {
        "premiums-are-unavailable": false,
        "claims-are-unavailable": false,
        "idns-are-unavailable": false
      }
    },
    "transfers": {
      "email-notifications": false
    },
    "data-protection": {
      "contact-purge-enabled": true
    }
  }
```

7.12. Lifetime Discounts

CentralNic's registry system implements point-of-sale promotions, whereby the standard transaction fees are subject to a discount. The specific promotion which will be applied to a billable transaction is determined by a number of rules, including the extension of the domain; the registrar, the type of transaction; the currency in which the transaction takes place; and the date and time of the transaction.

If, when a domain is registered, the transaction fee is subject to a promotional discount, then the discount will be applied to the "Registration" transaction. However, subsequent "Renewal" transactions will not normally be discounted, unless the specific promotion applies to both registrations and renewals, and remains "open" at the time of the transaction.

The Lifetime Promotions system changes the above system so that promotion can, if configured by CentralNic, also apply to future billable transactions for a domain.

Each promotion in the system has a human-readable name, which is included in the "description" element of <fee:fee> and <fee:credit> elements of EPP Fee <check> responses. CentralNic will configure all "lifetime" promotions so that the name includes the string "lifetime". Registrars can use the presence of this string in the "description" attribute of <fee:fee> and <fee:credit> elements to determine if the discount will also apply to future renewals of the domain.

An EPP Fee <check> will **not** apply a lifetime discount to renewals of domain names that are not already registered, so a registrar wishing to query for the renewal fee of a domain prior to registering it, should perform a Fee <check> for the "create"

command first, then determine if the "create" command is subject to a lifetime promotion. If it is, then this discount should be applied to the "renew" Fee <check> response to calculate the correct renewal fee. As an example:

"create" fee <check> response for example.uk.com:

```
<fee:cd>
    <fee:name>example.uk.com</fee:name>
    <fee:currency>GBP</fee:currency>
    <fee:command>create</fee:command>
    <fee:period unit="y">1</fee:period>
    <fee:fee description="Registration Fee" refundable="1"
      grace-period="P5D">16.25</fee:fee>
    <fee:fee description="UK.COM Lifetime Promotion"
      refundable="1"
      grace-period="P5D">-4.06</fee:fee>
    <fee:class>standard</fee:class>
</fee:cd>
```

"renew" fee <check> response for example.uk.com (before registration):

```
<fee:cd>
    <fee:name>example.uk.com</fee:name>
    <fee:currency>GBP</fee:currency>
    <fee:command>renew</fee:command>
    <fee:period unit="y">1</fee:period>
    <fee:fee
      description="Renewal Fee"
      refundable="1"
      grace-period="P5D">16.25</fee:fee>
    <fee:class>standard</fee:class>
</fee:cd>
```

To calculate the renewal fee before the domain is registered, deduct the value of the "UK.COM Lifetime Promotion" fee in the "create" response (ie 4.06) from the "Renewal Fee" in the "renew" response. The renewal fee will, therefore, be 12.19.

Once the domain has been registered, a "renew" Fee <check> will return the correct renewal fee for the domain. As an example:

"create" fee <renew> response for example.uk.com (after registration):

```
<fee:cd>
    <fee:name>example.uk.com</fee:name>
    <fee:currency>GBP</fee:currency>
    <fee:command>renew</fee:command>
    <fee:period unit="y">1</fee:period>
    <fee:fee
```

```
    description="Renewal Fee"
    refundable="1"
     grace-period="P5D">16.25</fee:fee>
   <fee:fee
     description="UK.COM Lifetime Promotion"
     refundable="1"
     grace-period="P5D">-4.06</fee:fee>
    <fee:class>standard</fee:class>
 </fee:cd>
```

This works out to the same fee as was calculated above.

Lifetime discounts will override any other discounts which may apply at the time of a renewal transaction, instead of being accumulative.

The lifetime discount is applied to the domain while it remains under your sponsorship. If the domain is transferred to a different registrar account, the discount is lost.

The lifetime discount does not apply to subsequent registrations of the same name. If the domain is ever deleted, then the discount is removed.

The lifetime discount will still apply if the domain goes onto the pendingDelete status but is subsequently restored.

<u>Lifetime Discounts are not applied to the second and subsequent years of multi-year registration</u>. If you receive an order for a multi-year registration for a domain that is subject to a lifetime discount, please submit a <create> command for 1 year, and then a subsequent <renew> command for the remaining period.

8. Registry Operations

8.1. DNS

CentralNic operates a network of DNS servers at many locations around the world. We are proud to be able to claim 100% availability of DNS services since we began operating in 1995.

8.1.1. Wildcard Records

CentralNic MAY place wildcard records into certain DNS zones. Registrars will be notified of the value of this record so that they can configure any DNS-based availability tests to ignore records that have this value.

If a wild-card record is placed into the zone, catch-all SPF records will also be placed in the zone to prevent domain-spoof spamming. The format of these SPF records is:

```
*.uk.com. 3600 IN TXT "v=spf1 -all"
*.uk.com. 3600 IN TXT "v=spf2.0/pra -all"
```

The SPF specification (see RFC 4408) explicitly states that SPF records do not propagate down to lower nodes in the DNS, so the use of the SPF record above will have no effect on domain names registered under .UK.COM.

Note: wildcard records will not be used with any generic TLDs (except during Controlled Interruption periods, see https://icann.org/namecollision).

8.1.2. DNSSEC

All zones are signed using DNSSEC. For further information please see

> https://www.centralnic.com/support/dnssec

As a general rule, all zones are signed using 2048-bit KSKs, 1024-bit ZSKs, and NSEC3 with opt-out, one iteration and no salt. Specific details, including our DNSSEC Practice Statement, may be found at the above URL.

8.2. WHOIS Service

Whois is one of the oldest Internet protocols still in use. It allows interested persons to retrieve information relating to Internet resources (domain names and IP addresses). Whois services are operated by the registries of these resources, namely TLD registries and RIRs.

Whois is described by RFC 3912, which serves as a description of existing systems rather than requiring specific behaviours from clients and servers. The protocol is a query-response protocol, in which both the query and the response are opaque to the protocol, and their meanings are known only the server and to the human user who submits a query. Whois has a number of limitations but remains ubiquitous as a means for obtaining information about name and number resources.

CentralNic operates a standard WHOIS server that you can use to check the details of any domain name registered with us. You can use any standard WHOIS client (including the UNIX terminal client) to query our WHOIS server at **whois.centralnic.com**.

Note: *use of the WHOIS service is subject to strict usage limits, to prevent unauthorised access to personal information about registrants. For more information, please see our WHOIS Access Policy at:*

> *https://registrar-console.centralnic.com/pub/whois_guidance*

8.2.1. Supported Query Types

8.2.1.1. Domain Query

By default, any query is assumed to be a domain name unless a keyword is prepended to the query. If the domain exists, then a record is returned which includes the following fields:

- Domain Name
- Internationalized Domain Name (if any)
- Registry Domain ID
- Registrar WHOIS Server
- Registrar URL
- Updated Date
- Creation Date
- Registry Expiry Date
- Registrar
- Registrar IANA ID
- Domain Status (multiple)
- Registrant Contact Information*
- Administrative Contact Information*
- Technical Contact Information*
- Name Servers (if any, multiple)
- DNSSEC status
- Billing Contact Information*
- Registrar Abuse Contact Email
- Registrar Abuse Contact Phone

The Domain ROID is the Repository Object Identifier as described in RFC 5730, §2.8. The ROID field corresponds to the <domain:roid> element of EPP <info> responses.

A domain may be associated with one or more status codes. These are represented using standard EPP mnemonic codes. A domain may have any of the following status codes:

- **pendingCreate** – a <domain:create> command has been received through the SRS, but the registration has not yet been finalised as an out-of-band review process has not yet been completed

- **addPeriod** – the domain is in the Add Grace Period

- **clientHold** – the registrar has temporarily removed the domain from the DNS

- **clientDeleteProhibited** – deletion of the domain is not permitted unless the sponsoring registrar removes this status code

- **serverDeleteProhibited** – deletion of this domain has been forbidden by the registry

- **inactive** – the domain has no DNS servers

- **pendingDelete** – the domain has left the Redemption Grace Period and is scheduled for deletion

- **redemptionPeriod** – the domain is in the Redemption Grace Period

- **pendingRestore** – a restore request has been received, but the Restore Report has not been received

- **pendingTransfer** – there is an active inter-registrar transfer for the domain

- **renewPeriod** – the domain is either in the Renew Grace Period or the Auto-Renew Grace Period

- **clientRenewProhibited** - renewal of the domain is not permitted unless the sponsoring registrar removes this status code

- **serverRenewProhibited** – renewal of the domain has been forbidden by the registry

- **serverHold** – the registry has temporarily removed the domain from the DNS

- **transferPeriod** – the domain is in the Transfer Grace Period

- **clientTransferProhibited** – transfer requests for the domain will be rejected unless the sponsoring registrar removes this status code

- **serverTransferProhibited** – transfer requests for the domain will be rejected by the registry

- **clientUpdateProhibited** – updates to the domain are not permitted unless the sponsoring registrar removes this status code)

- **serverUpdateProhibited** – updates to the domain have been forbidden by the registry

ⓒ CentralNic
Registry

- **ok** - present if none of the above applies.

The Registrant, Administrative, Technical and Billing Contact sections of the Whois record display the contact information for the contact objects that are associated with the domain.

Only those fields that have been enabled for display will appear in whois records. From July 22, 2018, the default setting is for contact fields to **not** be displayed.

Domains may have 0-13 DNS servers. If a domain name has no DNS servers, then the "inactive" status code appears in the Status section. If the registrant provided DS records for their DNSSEC-signed domain, then the "DNSSEC" field will contain the string "signedDelegation", otherwise it will contain the string "unsignedDelegation".

8.2.1.2. Host Query

Users can query for information about a host object by submitting a query of the form "nameserver [HOST]". The following information is included in host records:

- Server Name

- IPv4 address(es) (if any)

- IPv6 address(es) (if any)

- EPP status codes

- Sponsoring Registrar

- Creation Date

- Referral URL (if any)

A host object may have one or more IPv4 or IPv6 addresses if the host is "in-bailiwick", ie subordinate to a domain name within a zone operated by the registry. IP address information is not shown for "out-of-bailiwick" hosts.

Host objects may only have two status codes:

- **INACTIVE** - the host is not associated with any domain names

- **LINKED** - the host is associated with one or more domain names

The Referral URL is the website of the Sponsoring Registrar for this host. If the host is subordinate to a domain name in the TLD, this will be the sponsoring registrar of the parent name. If the host is out-of-bailiwick, then the sponsoring registrar is the registrar who issued the original <create> request.

8.2.1.2.1 Query by IP address

If an IPv4 or IPv6 address is provided instead of a fully qualified domain name, then the server will look for host objects that have this address. If there are multiple host objects that match the address, the server will return a list of all the matching host objects.

8.2.1.3. Registrar Query

Users can query for information about a registrar by submitting a query of the form "registrar [REGISTRAR]", where "[REGISTRAR]" is either the registrar's unique ID (eg H12345) or the full Organisation Name of the registrar.

The following information is included in Registrar records:

- Registrar ID

- Registrar Name

- Postal Address (as a single field)

- Telephone Number

- Email address

- Website URL

- Administrative Contact

- IANA ID (if ICANN accredited)

- Last Updated Date

8.2.2. Character Encoding

Responses are encoded as UTF-8. The server assumes that all queries are encoded in UTF-8.

8.2.3. IDN Support

The Whois service supports Internationalised Domain Names. Users may submit queries for IDN domains using either the (UTF-8 encoded) U-label or the A-label.

8.2.4. Rate Limiting and Access Control

The Whois protocol provides information in a human-readable format. It was not designed for consumption by automated systems but serves to provide information regarding domain name registrations.

Use of the Whois for performing domain name availability checks is not recommended. CentralNic does not guarantee, and registrars should not assume, the accuracy or timeliness of the data produced by the Whois system. Furthermore, CentralNic may change the layout or content of WHOIS records at any time, which may break automated parsers of WHOIS responses.

Registrars wishing to do high volume domain name availability checks are advised to use EPP, which offers significant benefits over the Whois service.

The EPP system also provides additional methods for obtaining information about domain name registrations, in a format that is strictly defined and easy to parse.

9. The Shared Registry System

CentralNic provides an object-oriented registry system, in which registrars manage domain, contact and host objects. Each object is associated with a single registrar, who is said to be the "sponsor" of that object. Objects may be created, updated, deleted and transferred between registrars.

9.1. Domain Objects

9.1.1. Registration

In order to be successfully registered in the system, a domain name MUST:

- have a prefix (the label to the left of the TLD) of between one and sixty-three characters

- only contain characters from the set A-Z, 0-9 and "." and "-" (case insensitive)

Additional restrictions on the validity of the prefix may also apply depending on the domain extension in question.

Domain names MAY also:

- have a Registrant Contact

- have an Administrative Contact

- have a Technical Contact

- have a Billing Contact

- have 0-13 DNS servers

- have zero or more DS records

Depending on the domain extension's policy, the Registrant, Administrative, Technical and/or Billing contacts may be mandatory.

DNS delegation information is not required for a successful registration. However, please note that domains that are registered without any DNS delegation MAY

resolve to an advertising page due to the presence of a wildcard in the zone (see Section [Wildcard records](#)).

Domain names may be registered for a period between 1 and 10 years.

> *Note: CentralNic may withhold access to domain registration functions if a registrar's account balance exceeds the credit limit specified for their account.*

9.1.2. Renewal

Registrars may renew domain names for any whole period of years, no more than 10 years into the future.

> *Note: CentralNic may withhold access to domain renewal functions if a registrar's account balance exceeds the credit limit specified for their account.*

9.1.3. Transfer

Domain names may be transferred between registrars. The procedure for domain name transfers is as follows:

1. The registrant acquires the authInfo code for the domain name from the losing registrar, and, depending on the status of the domain, requests that the "clientTransferProhibited" status be removed from it.

2. The registrant supplies the authInfo code to the gaining registrar.

3. Depending on TLD policy, the gaining registrar MAY be required to obtain authorisation from the Administrative Contact before submitting the transfer request to the registry.

4. The gaining registrar submits a transfer request to the registry, using the authInfo code received from the registrant.

5. The registry notifies the losing registrar of the transfer request. The losing registrar has five calendar days to explicitly approve or reject the transfer, after which the transfer is automatically approved.

6. Depending on TLD policy, the losing registrar MAY be required to obtain authorisation from the Administrative Contact before approving the transfer request.

7. Once the transfer is approved, the registry notifies the gaining registrar, and the domain name is transferred within 60 seconds.

Depending on TLD policy, the gaining registrar may choose not to add renewal years when requesting the transfer. For all gTLDs, there is a minimum renewal period of 1 year for all inter-registrar transfers.

Notifications are sent to both the gaining and losing registrar via email and the EPP message queue (see Section Message Queue).

Registrars are required to provide the authInfo code for domain names to registrants upon request.

When a domain transfer is approved and processed by our system, any subordinate host objects will also be transferred to the gaining registrar.

Note: CentralNic may withhold access to domain transfer functions if a registrar's account balance exceeds the credit limit specified for their account.

9.1.4. Deletion

Registrars may delete domain names at any time.

- If the domain name is still within the Add Grace Period, then it is immediately deleted upon receipt of a deletion request, and may be immediately re-registered. Note that the registrar may still subsequently be charged for the registration of the domain if determined by the Add Grace Limits Policy.

- If the domain name is not within the Add Grace Period, it is marked as "pendingDelete" and, if not restored during the Redemption Grace Period, will be deleted after 35 days.

9.1.4.1. Restoring a Domain on "Pending Deletion" Status

If a domain name has been placed on "Pending Deletion", either because (a) it has expired and your account is set to "auto-delete", or (b) you have requested its deletion, you may restore it at any time within the "Pending Delete Restorable" period (30 days) by using the 'restore' function on the Registrar Console, or the EPP RGP extension (see Section Registry Grace Period Mapping). Restoration of an expired domain requires a renewal for at least one year.

Domains are immediately restored upon receipt of a restore request: it is not necessary to submit a restore report.

9.1.5. Grace Periods

The following Grace Periods are supported:

Name	Description	Period
Add Grace Period*	Time after a domain is registered during which it may be deleted in exchange for a full credit	5 days
Renew Grace Period	Time after a domain is renewed during which it may be deleted in exchange for a full credit	5 days
Auto-renew Grace Period	Time after a domain is auto-renewed during which it may be deleted in exchange for a full credit	45 days
Transfer Grace Period**	Time after a domain is transferred during which it may be deleted in exchange for a full credit	5 days
Redemption Grace Period	The time following the deletion of a domain during which it may be restored. There is a five-day "Pending Delete Period" which follows the Redemption Grace Period.	30 days

*Note: depending on TLD policy, the Add Grace Period Limits Policy may apply, and you may still be charged if the number of AGP deletes exceeds the permitted volume.

**Note: the Transfer Grace Period only applies to domain transfers that include a renewal as part of the transfer: this may or may not be mandatory, depending on the specific TLD policy.

9.1.6. Hold Periods

The following Hold Periods are supported:

Name	Description	Period

Registration Hold Period	The period following the initial registration of a domain, during which it cannot be transferred	60 days
Transfer Hold Period	The period following a successful transfer, during which another transfer cannot be requested	60 days

9.1.7. pendingCreate Status

The Registry system supports the "pendingCreate" status for domain names, as described in RFC 5731, §3.3. Domains in this state are fully registered in the database (subsequent <create> commands would fail with an Object Exists error) but are not present in the DNS.

This status is used when a particular TLD implements a policy whereby registration requests are verified by a third party such as a Sponsoring Organisation or Validation Agent. Following the out-of-band review of the request, the registration may be approved or denied. Domain Launch Applications submitted using the Launch EPP extension (see Section Launch Phases) will also have the pendingCreate status, however, multiple applications for the same domain may be present in the database (and are distinguished by the "applicationID" element).

When a domain on pendingCreate is either approved or rejected, the registrar notified by EPP message queue. An example message is shown below:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully; ack to dequeue</msg>
    </result>
    <msgQ count="5" id="12345">
      <qDate>2015-04-04T22:01:00.0Z</qDate>
      <msg>Pending action completed successfully.</msg>
    </msgQ>
    <resData>
      <domain:panData
       xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name paResult="1">example.tickets</domain:name>
        <domain:paTRID>
          <clTRID>ABC-12345</clTRID>
          <svTRID>54321-XYZ</svTRID>
        </domain:paTRID>
        <domain:paDate>2105-04-04T22:00:00.0Z</domain:paDate>
      </domain:panData>
    </resData>
    <trID>
      <clTRID>BCD-23456</clTRID>
      <svTRID>65432-WXY</svTRID>
    </trID>
  </response>
</epp>
```

See Section 3.3 of RFC 5731 for further details. The "paResult" attribute of the <domain:name> element indicates whether the <create> was successful (1) or unsuccessful (0).

9.1.8. Internationalised Domain Names (IDNs)

CentralNic's registry system allows the registration of Punycode-encoded internationalised domain names (IDNs). CentralNic's IDN system operates in a similar manner to that of other registries, taking an inclusion-based approach with defined tables of permitted code points. These tables are published at the following URL

> https://www.centralnic.com/names/domains/idn/tables

IDNs are subject to the following rules:

1. The "A-label" must be a valid domain name in its own right (ie length and composition rules for ASCII domain names must also successfully be passed).

2. The A-label must be valid according to the IDNA2008 rules. We will test this by decoding the A-label to a UTF-8 string, and then re-encoding. If the re-encoded string matches the original string, this test is passed.

3. The "U-label" must contain at least the minimum number of characters. The minimum number varies depending on the domain extension and the registrar but is normally 1.

4. The U-label must be wholly composed of characters from one of the IDN tables associated with the domain extension.

5. Certain IDN tables have additional validation and contextual rules. Registrars should consult our website for the rules for specific tables:

> https://www.centralnic.com/support/idn/tables

9.1.9. Domain Lifecycle
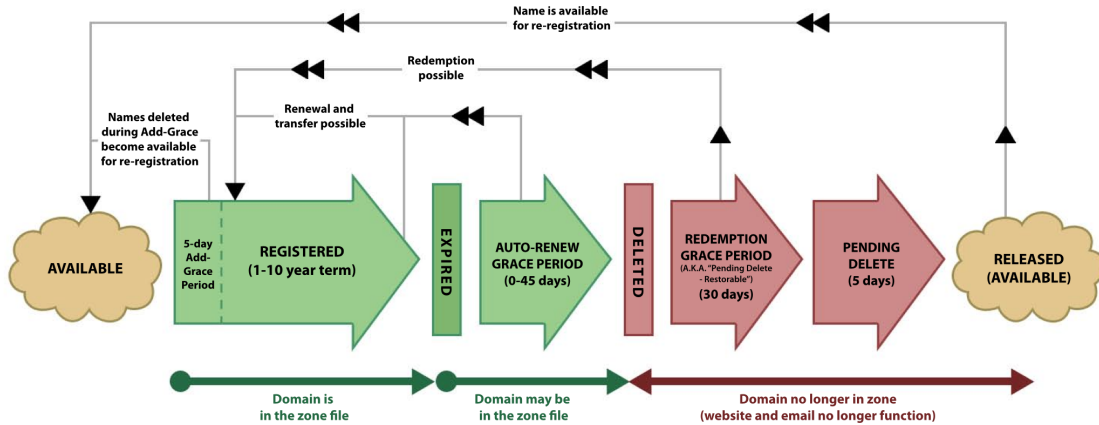
Figure 1: Domain Name Lifecycle

9.1.9.1. Available

The domain is not registered. No delegation records are present in the DNS, and the whois system will return a "NOT FOUND" response to queries. An EPP <check> command will return an "avail" status of 1.

9.1.9.2. Registered

A registrar submits an EPP <create> command or registers the domain name via the Registrar Console. The registration fee is deducted from the registrar's balance. The initial registration period may be any whole number of years between one (1) and ten (10).

For five (5) calendar days after the registration of the domain, the registrar can delete the domain and receive a credit for the registration fee (subject to the Add Grace Period Limits Policy).

While the domain is registered, it is delegated to the specified name servers and will resolve normally. During this time, the registrar may update the domain name's DNS settings, lock statuses and contact associations, and may extend the registration period (subject to a maximum of ten (10) years) by submitting a <renew> EPP command or using the Registrar Console.

The domain may also be transferred to a different sponsoring registrar. Upon such transfer, the domain name may also be renewed for at one or more years.

9.1.9.3. Expired

When the expiry date is reached, the domain name is automatically renewed for a period of one year, and the renewal fee is deducted from the registrar's account.

For forty-five (45) days after the auto-renewal (Auto-Renew Grace Period), the registrar can delete the domain and receive a credit for the renewal fee.

9.1.9.4. Redemption Grace Period

Should the registrar delete the domain, the domain enters the Redemption Grace Period. During this period, the domain name will no longer resolve as all delegation information is removed from the TLD zone.

For the first thirty (30) days after receipt of the delete request, the domain is in the "Pending Delete Restorable" state. During this time, the registrar may submit an RGP restore request via EPP or the Registrar Console. The domain is then placed into the "Pending Restore" state.

The registrar must then submit an RGP Restore Report detailing the reason why the restore request has been submitted. If the Restore Report is received within five (5) calendar days of the original restore request, then the domain is restored. However, if the Restore Report is not received within this period, then the domain falls back into the "Pending Delete Restorable" state.

9.1.9.5. Pending Delete

Forty (40) days after the receipt of the delete request, the domain leaves the "Pending Delete Restorable" and enters the "Pending Delete" status. The registrar cannot submit a Restore Request during this period.

9.1.9.6. Released

Five (5) days after the domain enters the "Pending Delete" status the domain name is purged from the database and is once again available for registration.

9.2. Contact Objects

Domain names may have a Registrant, Administrative, Technical or Billing Contacts.

A contact object may have the following attributes:

- Name: the full name of the person the contact refers to, or an organisational name (ie "Hostmaster") - Max 100 characters

- Company: the full name of the organisation, if appropriate - Max 80 characters

- Street Address: 1-3 text fields representing the postal address - Max 255 characters each

- City: the city or town - Max 255 characters

- State/Province: the state, province or geographic region - Max 255 characters

- Postcode: the postal or ZIP code of the contact - Max 15 characters

- Country: the two-letter ISO-3166 country code for the contact. Note that "UK" may be used interchangeably with "GB", however our server will translate "UK" into "GB" in <info> commands and whois records. - Max 2 characters

- Phone: the telephone number of the contact in e164a format.

- Fax: the fax number of the contact in e164a format.

- E-mail: the e-mail address of the contact. - Max 100 characters

Of the above, the fields that are mandatory for all contact objects are name, city, country and email. All other fields are optional.

9.2.1. Transfers

Contact objects may be transferred between registrars in the same way as domain names.

Registrars may find it easier to simply recreate new contacts objects for recently transferred domain names, since this avoids the delay associated with contact transfers.

9.2.2. Purging of Unused Contact Objects

In order to comply with data protection legislation, CentralNic purges unused contact objects from the registry database. When a domain is updated to remove a contact, and that contact loses the "linked" status, then it will gain the "pendingDelete" status, and then will later be deleted from the database.

Registrars can opt-out of this system on the by logging a request to support@centralnic.com

9.2.2.1. Restoring Contacts on Pending Delete

Registrars can "restore" a contact object that is scheduled for deletion by submitting a "no-op" <contact:update> EPP command. An example of such a command is below.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <update>
      <contact:update
        xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
        <contact:id>sh8013</contact:id>
        <contact:chg />
      </contact:update>
    </update>
    <clTRID>ABC-12345</clTRID>
  </command>
</epp>
```

When the server receives this command, the pendingDelete status is removed from the contact object and it will not be deleted. However, the object may still be deleted in future if it is not linked to a domain name once restored.

9.3. Host Objects

Registrars may freely create host objects that are "out-of-bailiwick" (i.e. not subordinate to any of the zones for which CentralNic is authoritative). IP address information supplied during a <create> or <update> request will be ignored for such hosts, and our DNS servers will not publish glue for them.

Registrars may only create "in-bailiwick" (ie subordinate to a zone for which CentralNic is authoritative) if they are the sponsor for the superordinate domain name. That is to say, a registrar may only create a host object named NS0.EXAMPLE.UK.COM if that registrar is also the sponsor for EXAMPLE.UK.COM.

Registrars may not create host objects that are subordinate to domain names that are sponsored by other registrars, or that are subordinate to an unregistered domain name.

For in-bailiwick host objects, registrars must provide at least one IPv4 or IPv6 address.

9.3.1. Sponsorship and Transfers

In-bailiwick host objects are automatically transferred when the superordinate domain name is transferred. Out-of-bailiwick host objects may not be transferred.

9.3.2. Orphan Glue

Subordinate host objects are deleted when the parent domain is deleted. If other domains in the registry are delegated to these hosts, then the delegation is removed.

10. EPP System

EPP is a stateful XML protocol layered over TCP (see RFC 3734). Protected using lower-layer security protocols, clients exchange identification, authentication, and option information, and engage in a series of client-initiated command-response exchanges. All EPP commands are atomic (there is no partial success or partial failure) and designed so that they can be made idempotent (executing a command more than once has the same net effect on system state as successfully executing the command once).

EPP provides four basic service elements: service discovery, commands, responses, and an extension framework that supports the definition of managed objects and the relationship of protocol requests and responses to those objects.

EPP servers respond to client-initiated communication (which can be either a lower-layer connection request or an EPP service discovery message) by returning a greeting to a client. The server then responds to each EPP command with a coordinated response that describes the results of processing the command.

EPP commands fall into three categories: session management, queries, and transform commands. Session management commands are used to establish and end persistent sessions with an EPP server. Query commands perform read-only object information retrieval operations. Transform commands perform read-write object management operations.

The server processes commands in the order they are received from a client. The protocol includes features that allow for offline review of transform commands before the requested action is completed. In such situations, the response clearly notes that the command has been received but that the requested action is pending. The corresponding object then reflects the processing of the pending action. The server will also notify the client when offline processing of the action has been completed. Object mappings describe standard formats for notices that describe the completion of offline processing.

EPP uses XML namespaces to provide an extensible object management framework and to identify schemas required for XML instance parsing and validation. These namespaces and schema definitions are used to identify both the base protocol schema and the schemas for managed objects.

10.1. Objects supported

Registrars may create and manage the following object types in the CentralNic EPP system:

- Domains (RFC 5731)

- Host objects (RFC 5732)

- Contact objects (RFC 5733)

10.2. Commands supported

CentralNic supports the following EPP commands:

- <hello> - retrieve the <greeting> from the server

- <login> and <logout> - session management

- <poll> - message queue management

- <check> - availability check

- <info> - object information

- <create> - create object

- <update> - update object

- <renew> - renew object

- <delete> - delete object

- <transfer> - manage object transfer

The EPP Server has been designed to adhere to version 1.0 of the EPP specification, as defined in RFC 5730. In addition, we have developed our EPP implementation to operate in an identical manner to that operated by other registries, to minimise the amount of new software that must be written by registrars wishing to provision domain names in zones managed by CentralNic.

Access to the EPP server is via a TLS-encrypted TCP interface at **epp.centralnic.com**, port 700.

For examples of EPP frames for creating and management registry objects such as domain names, host objects and contact objects, please consult the relevant RFCs.

10.3. Operational Testing and Evaluation (OT&E)

CentralNic provides an Operational Testing and Evaluation (OT&E) environment to allow registrars to develop and test their client applications. This system identical to the production registry system but uses a separate database. Registrars are invited to use this system to test their client implementations, without fear of affecting the production registry system. Registrars can create multiple account credentials for use in the OT&E system via the Registrar Console.

Access to the OT&E EPP server is via a TCP interface (secured by TLS) at **epp-ote.centralnic.com**, port 700, and is not restricted by IP address. There is also a version of the Registrar Console for the OT&E system at:

https://ote-console.centralnic.com/

The OT&E whois service is provided at **whois-ote.centralnic.com**, port 43.

> *Note: Registrars are not required to complete an OT&E certification before gaining access to the production environment. All registrars have immediate access to the production environment once their account has been activated.*

10.3.1. OT&E Testing of TMCH Sunrise and EPP Launch Extension

In the OT&E environment, all domain applications received using the EPP Launch extension (see Section Launch Extension) are processed every 30 minutes. This means that registrars may rapidly test and iterate their client implementations, including receipt and processing of EPP messages, without the delays involved in waiting for a normally long-running launch phase to finish.

As of writing, the Trademark Clearinghouse has only issued a finite number of Signed Mark Data (SMD) files from its OT&E environment. This means that there are also a finite number of domains that registrars can use to test their implementations and integration with our registry.

To circumvent this limitation, all domain names in the OT&E environment, which were created as part of a Trademark-Clearinghouse backed sunrise phase, or which were created with Trademark Claims information, will be automatically purged from the system after five days. This allows these names to be reused for future testing.

If the Trademark Clearinghouse allows the creation of arbitrary numbers of SMD files in their OT&E environment, then this workaround will be removed.

10.3.2. special OT&E TLDs: .eclipse

In OT&E there is a special TLD operating which is called .eclipse referring to sunrise. This TLD provides the opportunity to test for TLD's which are creating domain in pendingCreate status. In normal circumstances the registry will then manually review each domain and approve or deny each domain.

In order to simulate that you can create 2 types of domains:
- domains ending with -reject.eclipse like starthere-reject.eclipse
- all other domains

All domains ending with -reject.eclipse will be rejected in the normal refresh cycle of OT&E. All other domains will be approved.

10.3.3. **special OTE&E TLDs: .eaptest1**

In order to help registrars prepare for EAP phases and the mechanics that our system will use.

The OT&E EAP is always 2x24hours long (2 days of EAP), followed by a 24-hour GA period, after which EAP starts again.

10.4. How to create your EPP client

10.4.1. Scope

EPP allows you to manage pretty much everything there is to manage about a domain name. This result is in a large set of commands and variations on commands which probably seem overwhelming when you think about implementing this.

This is why we recommend making a phased approach to your EPP implementation:

1. Start with a *domain check* command only which will get you familiar with logging in and connecting to EPP

2. Move on to a *golden path**
3. From here your own effort will guide you on where to focus next on your continuous improvement

* By golden path we mean the cornerstone of domain registration, which includes creating contacts, nameservers followed by a domain.

Or course other development best practices like test-driven development and Agile help in making effective use of your resources. We'll provide you with some sample test scenarios to help you on your way and leave the rest up to your own criteria.

10.4.2. Don't start from scratch

EPP has been around for a long time and many development years have gone into developing open-source clients. So we suggest starting there. Depending on which development language you are using, look for a popular, well maintained EPP project. For example this one, or our own php repo. These libraries will take care of the EPP protocol itself so you can concentrate on sending and receiving the XML frames.

We have a fully functional OTE or acceptance environment which you can use to validate your code against.

Get yourself a test account on our OTE/acceptance platform and start with getting a domain check command to work.

This implementation alone will save you heaps of time and allow you or your customers to automatically check the availability of a domain.

10.4.3. Sample domain check test frames

10.4.3.1. Command

```
<command>
  <check>
    <domain:check xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
      <domain:name>${yourdomain.tld}</domain:name>
    </domain:check>
  </check>
  <clTRID>ABC-12345</clTRID>
</command>
```

10.4.3.2. Response

```
<response>
  <result code="1000">
    <msg>Command completed successfully.</msg>
  </result>
  <resData>
    <domain:chkData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
      <domain:cd>
        <domain:name avail="1">${yourdomain.tld}</domain:name>
      </domain:cd>
    </domain:chkData>
  </resData>
  <trID>
    <clTRID>ABC-12345</clTRID>
    <svTRID>TEST-xtld-5018</svTRID>
  </trID>
</response>
```

Once you get this working, move on to the golden flow. Here are some sample frames for creating a contact, nameserver and domain:

10.4.4. Sample "golden path" frames

10.4.4.1. Create Contact Command

```
<command>
  <create>
    <contact:create xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
      <contact:id>${handle}</contact:id>
      <contact:postalInfo type="int">
        <contact:name>John Doe</contact:name>
        <contact:org>Example Inc.</contact:org>
        <contact:addr>
          <contact:street>123 Example Dr.</contact:street>
          <contact:street>Suite 100</contact:street>
          <contact:street>A</contact:street>
          <contact:city>Dulles</contact:city>
          <contact:sp>VA</contact:sp>
          <contact:pc>20166-6503</contact:pc>
          <contact:cc>US</contact:cc>
        </contact:addr>
      </contact:postalInfo>
```

```
        <contact:voice x="1234">+1.7035555555</contact:voice>
        <contact:fax>+1.7035555556</contact:fax>
        <contact:email>jdoe@example.com</contact:email>
        <contact:authInfo>
          <contact:pw>${newcontactpw}</contact:pw>
        </contact:authInfo>
        <contact:disclose flag="0">
          <contact:voice/>
          <contact:email/>
        </contact:disclose>
      </contact:create>
    </create>
    <clTRID>ABC-12345</clTRID>
  </command>
```

10.4.4.2. Create Contact Response

```
<response>
  <result code="1000">
    <msg>Command completed successfully.</msg>
  </result>
  <resData>
    <contact:creData xmlns:contact="urn:ietf:params:xml:ns:contact-1.0">
      <contact:id>${handle}</contact:id>
      <contact:crDate>2012-03-12T15:54:00.781Z</contact:crDate>
    </contact:creData>
  </resData>
  <trID>
    <clTRID>ABC-12345</clTRID>
    <svTRID>TEST-xtld-979</svTRID>
  </trID>
</response>
```

10.4.4.3. Create Host Command

```
<command>
  <create>
    <host:create xmlns:host="urn:ietf:params:xml:ns:host-1.0">
      <host:name>${host.name}</host:name>
      <host:addr ip="v4">192.0.2.2</host:addr>
      <host:addr ip="v4">192.0.2.29</host:addr>
      <host:addr ip="v6">1080:0:0:0:8:800:200C:417A</host:addr>
```

```
        </host:create>
      </create>
      <clTRID>ABC-12345</clTRID>
    </command>
```

10.4.4.4. Create Host Response

```
    <response>
      <result code="1000">
        <msg>Command completed successfully.</msg>
      </result>
      <resData>
        <host:creData xmlns:host="urn:ietf:params:xml:ns:host-1.0">
          <host:name>${host.name}</host:name>
          <host:crDate>2020-01-15T00:00:00.0Z</host:crDate>
        </host:creData>
      </resData>
      <trID>
        <clTRID>ABC-12345</clTRID>
        <svTRID>TEST-xtld-913</svTRID>
      </trID>
    </response>
```

10.4.4.5. Create Domain Command

```
    <command>
      <create>
        <domain:create xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
          <domain:name>${domain.name)</domain:name>
          <domain:ns>
            <domain:hostObj>${host.name}</domain:hostObj>
          </domain:ns>
          <domain:registrant>${handle}</domain:registrant>
          <domain:contact type="tech">${handle}</domain:contact>
          <domain:contact type="admin">${handle}</domain:contact>
        </domain:create>
      </create>
      <clTRID>ABC-12345</clTRID>
    </command>
```

10.4.4.6. Create Domain Response

```
    <response>
```

```
<result code="1000">
        <msg>Domain ${domain.name} created successfully with ROID
1234566-CNIC</msg>
    </result>
    <resData>
     <domain:creData xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
      <domain:name>${domain.name}</domain:name>
      <domain:crDate>2020-01-10T15:13:41.0Z</domain:crDate>
      <domain:exDate>2021-01-10T23:59:59.0Z</domain:exDate>
     </domain:creData>
    </resData>
    <trID>
      <clTRID>ABC-12345</clTRID>
      <svTRID>TEST-xtld-17712</svTRID>
    </trID>
  </response>
```

As you can see it's not rocket science and with a bit of tinkering you can get this to work pretty quickly.

When you are familiar with the domain checks and golden flow, you will find plenty of information in this document that will allow you to dig a little deeper and expand as your business needs occur.

10.5. The Shared Registry System

10.5.1. Domain Names

10.5.1.1. Allocation of domain names using pre-generated authInfo codes

Certain TLDs may control the allocation of domain names by means of a pre-generated authInfo code. Once generated and associated with a domain name, that domain name may not be allocated unless the registrar inserts the correct code into the <domain:pw> element of the <create> command.

For example, say that the domain EXAMPLE.TLD has been associated with the authInfo code "abc12345". The domain remains available for registration, but a <create> command will only be successful if the authInfo code above is supplied in the <create> command like so:

```xml
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:period unit="y">1</domain:period>
        <domain:registrant>jd1234</domain:registrant>
        <domain:contact type="admin">sh8013</domain:contact>
        <domain:contact type="tech">sh8013</domain:contact>
        <domain:authInfo>
          <domain:pw>abc12345</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
  </command>
</epp>
```

Once generated, the authInfo code may be used by any registrar. authInfo codes will typically be generated by the registry and then passed to a registrant, who will then present the code to their preferred registrar.

Once the domain has successfully been allocated, the sponsoring registrar may use the <update> command to change the authInfo code for the domain as per its own policies.

10.5.1.2.Domain Update Command

EPP Registrars may add or remove the following EPP status codes to domain names:

- **clientHold** – NS records will not be entered into the zone for the domain name while this status code is applied.

- **clientTransferProhibited** – transfer requests for this domain name will be automatically rejected while this status code is applied.

- **clientRenewProhibited** – when this status code is applied, all requests by the sponsoring client to renew the domain are rejected.

- **clientUpdateProhibited** – when this status code is applied, all requests by the sponsoring client to update the domain name are rejected (except if the request includes removing this status from the domain name).

- **clientDeleteProhibited** – when this status code is applied, all requests by the sponsoring client to delete the domain name are rejected.

10.5.1.3.Domain status codes

When a domain is in one of the following statuses, it will be represented by following EPP codes:

- Status "P"
-

10.5.2. Contact Objects

When constructing an EPP <create> or <update> frame, registrars MUST supply an internationalized form for address data (see RFC 5733). Internationalized address information MUST be composed solely from characters from the ASCII range. Registrars MAY also provide localised address data, which MAY contain any valid UTF-8 code point.

Please note that contact IDs, whether server or client assigned, are case-insensitive. That is, you cannot create an object with an identifier of abc-12345 if an object with an identifier of ABC-12345 already exists. If you perform an EPP request for object abc-12345, information about ABC-12345 will be returned.

10.5.3. Disclosure of Contact Information

Registrars may use the <contact:disclose> element to enable display of contact attributes in whois and RDAP records. By default, all contact attributes are opted-out of display*.

When performing a <create> or <update> command, the <contact:disclose> element may be used to enable disclosure of specific contact attributes, as per RFC 5733 Section 2.9:

```
<contact:disclose flag="1">
      <contact:name type="int"/>
      <contact:name type="loc"/>
      <contact:org type="int"/>
      <contact:org type="loc"/>
      <contact:addr type="int"/>
      <contact:addr type="loc"/>
      <contact:voice/>
      <contact:fax/>
      <contact:email/>
</contact:disclose>
```

An <update> command which contains a <contact:disclose> with a "flag" attribute of "0" or "false" will remove the specified attributes from display.

Note: prior to July 22, 2018, the default was for all contact attributes to be opted-in to display by default.

10.5.3.1. Performing <info> commands on Contact Objects

The EPP system will refuse all <info> commands on contact objects for which you are not the sponsor, unless you provide a valid authInfo code in the <contact:pw>

∽ CentralNic
Registry

element (which may be that of a related domain object, signalled using the "roid" attribute).

10.5.4. Host Objects

Registrars can change the IP addresses associated with a host object using the EPP <host:update> command.

The <update> command also permits changing the DNS name of a host object, but with the following conditions:

1. Registrars may not rename a host object that is not subordinate to a domain name under their sponsorship. This includes "out-of-bailiwick" host objects.

2. Registrars may not rename a host object so that it becomes subordinate to a domain name that is sponsored by another registrar;

3. Registrars may not rename a host object so that it becomes subordinate to a non-existent domain name;

4. Registrars may rename a host object so that it is no longer "in-bailiwick", that is, not subordinate to any DNS zone for which CentralNic is the authority. Registrars should be aware that this means that IP address information will not be stored or published for this host object if this happens.

Changes to the DNS name of a given host are applied to the NS records of all domain names that are delegated to that host. This means that it is possible that domain names sponsored by other registrars could be affected by renaming a host object: as a result, registrars should exercise caution when using this feature, and consider using an alternative solution instead (such as creating a new host object and updating any domain names).

10.6. Credentials and Access Control

Registrars are issued with a username (their registrar ID) and a password. This password cannot be used to access any other service and only this password can be used to access the EPP system. Registrar officers with the "Management" access level can change their EPP password via the Registrar Console.

RFC 5730 requires "mutual, strong client-server authentication". CentralNic requires that all registrars connect using an SSL certificate. This certificate may be obtained from a recognised certificate authority, or it may be a self-signed certificate

registered with CentralNic via the Registrar Console. Registrar officers with the "Management" access level or above can upload SSL certificates for their account.

10.7. Session Limits and Transaction Volumes and Fair Use Policy

10.7.1. During Normal Operations

During normal operations, registrars may maintain up to **100 concurrent sessions** to the EPP system. Check commands are limited to 50 objects per command. Other command throughput is not throttled, meaning that the server will respond to commands immediately (once the command has been processed and the response has been prepared).

As an exception to the above, domain <create> commands which result in a 2302 "object exists" error will be subject to a minimum 2-second round-trip time. This policy serves to mitigate the operational impact of "drop-catching" activity on the registry system (see Section <u>Fair use policy</u> below).

10.7.2. During Launch Periods

CentralNic may impose limits on the number of concurrent connections available to each registrar during Launch Periods (the time shortly before and after a domain extension enters its "General Availability" phase). During these periods the following limitations will apply:

- Maximum number of concurrent sessions per registrar: **10 concurrent sessions**

- Maximum throughput per session: **5 domain <create> commands per second** *(no limit on other commands)*

The time periods during which the above restrictions will be in place will be communicated to registrar's operational point of contact prior to the date of the launch.

Note: these limits may be subject to revision, so registrars should always check the latest version of this document before participating in a launch.

10.7.3. Effect on Pipelining

Registrars can obtain a modest enhancement on performance by means of pipelining (see RFC 5734, Section 3). During launch periods, the EPP system will impose a maximum transaction rate as described above. Due to the way that this is implemented, commands that are pipelined by a client will be buffered in the client's send queue and will be processed by the server (in the order that they are received) at the above rate.

This means that, if the maximum throughput is 5 commands per second, if a registrar sends 200 <create> commands in a single burst, the minimum time required to execute all 200 commands will be 200 / 5 = 40 seconds.

Note: registrars should avoid pipelining so many commands that the session idle timeout (described below) is reached. If this occurs, the connection will drop, and any unprocessed commands will be discarded.

10.7.4. Session Timeouts

The system will terminate an idle session if no command has been received after 300 seconds. Registrars may wish to use the <hello> command to maintain connections for longer than this period. Please note that care should be taken to avoid a timeout when using pipelining (see section effect on pipelining).

10.7.5. Fair Use Policy

CentralNic has an obligation to ensure the availability, stability and reliability of the Shared Registry System for all registry partners and registrars. We reserve the right to take action to maintain the stability of the registry in the face of exceptional, abusive or malicious registrar activity, even if that activity falls within the limits outlined elsewhere in this document.

CentralNic may limit a registrar's access to any and all features of the Shared Registry System, on a temporary or indefinite basis, where required to meet our obligations to other registrars. This includes (but is not limited to):

- restricting the number of EPP connections that registrar may maintain;
- rate-limiting of specific EPP commands (globally or for specific objects); and
- enabling of response caching (so that the registrar does not receive real-time updates to query commands).

All registrars are expected to apply reasonable efforts to ensure that their EPP and Whois/RDAP implementations comply with the relevant protocol specifications and best practices. Such best practices include (but are not limited to):

- checking the response code received from the server before retrying a failed command (since the response code may indicate a permanent error, meaning that the command can never result in a successful response);
- delaying resubmission of commands when a "rate limiting" response (such as a 429 response to RDAP queries) is returned;
- Using the <check> command to determine availability before submitting a <create> command;
- not submitting more than one EPP <check> (or RDAP HEAD) request per second for the same object;
- use the response of a create command instead of doing an immediate check command to check how the domain was created
- Check the result of a login command as it might indicate that you are being rate limited because of abuse

10.8. Transaction Logging and Reporting

All "transform" commands are logged. Transform commands are: <create>, <renew>, <update>, <delete> and <transfer>. The system logs the time and date when the command was received, the registrar that submitted it, the result code, and message. All commands, whether successful or not, are logged (except for domain <create> commands which result in a 2302 response).

Registrars have access to the log for their account via the Registrar Console. The log viewer permits filtering by command, object type, object ID (domain, hostname, contact ID), result code and timestamp.

Query commands (<check>, <info>, <poll op="req">) and session commands (<login>, <logout> and <hello>) are not logged due to the large volume of such queries (particularly <check> queries).

10.9. Immediate re-registration of purged domains ("drop-catching")

"Drop-catching" is the practice of re-registering an expired domain name shortly after it has been purged from the registry database.

Due to the competitive nature of the "drop-catching" market place, registrars who offer this service typically compete with others to obtain names by submitting a large volume of EPP <create> commands, beginning shortly before the anticipated time that domains are purged from the database.

In order to mitigate any operational impact from this activity, to maintain the stability of the shared registry system, and ensure equal access to all registrars, the following policies are in place.

1. Each registrar account is restricted to 100 concurrent connections to the EPP system.

2. If the EPP system receives a domain <create> command, for which the response code is 2302 (object exists), the server will wait until 2,000 milliseconds have elapsed (from the time when the processing of the command began) before sending the response back to the client.

3. Rather than purging domains in a single batch once per day, domains are purged throughout the day, normally within 60-120 seconds of exiting the 5-day Redemption Hold Period (RHP).

4. Registrars who wish to perform drop-catching activity can use the Domain Drop List files (available from the FTP service and described in Section 6.3.2.1) to determine the approximate time when a given domain will be purged. The registrar can then attempt to "snipe" individual domains as they are purged.

CentralNic reserves the right to amend these policies at any time in order to preserve the stability of the registry system and ensure equal access to all registrars.

As a reference of good practice we have included an example of good practice regarding drop catching. Please also refer to the section fair use policy.

Dropcatching done well: good flow, no rate limiting
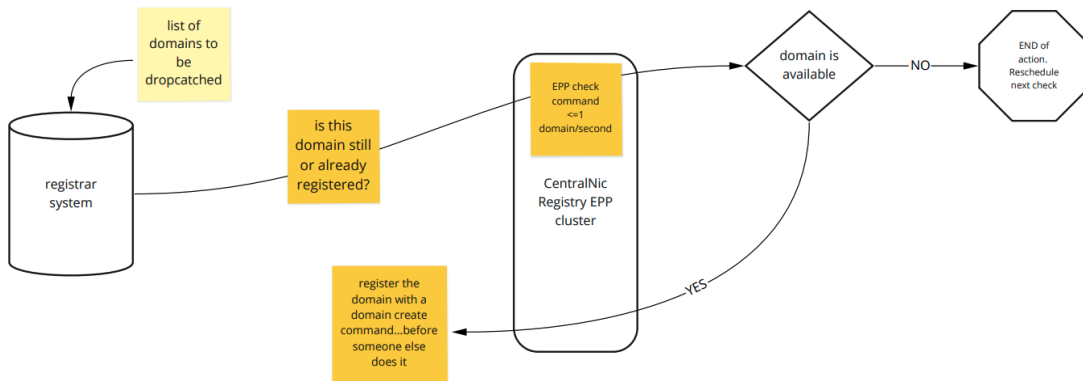
And also an example in what will result in being flagged as unfair use of the platform and the resulting penalty of only allow 5 sessions for your registrar account:

Dropcatching bad flow, rate limiting
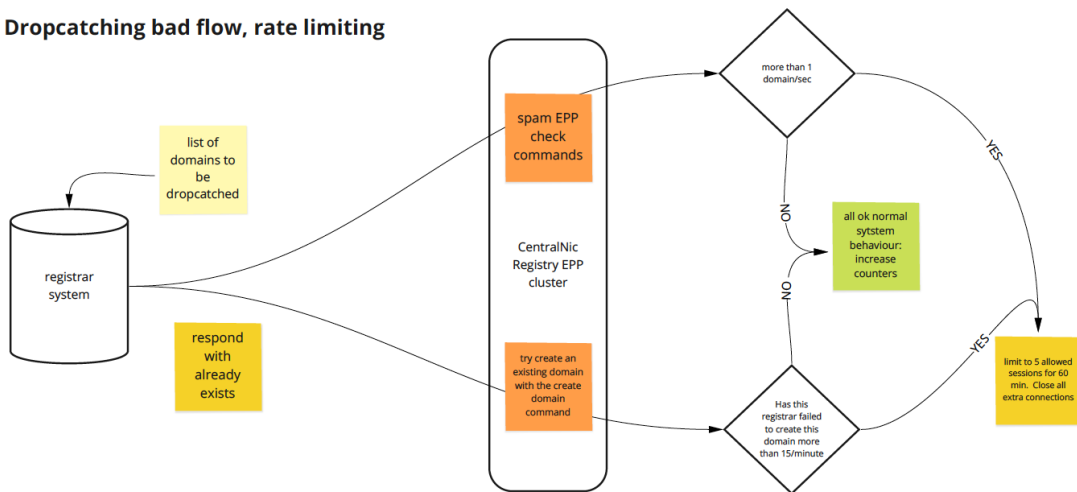
10.10. XML Schema Validation

The EPP system automatically validates all request frames against the XSD schema definitions provided in the RFCs. Should a non-validating request be received, an error is returned to the client.

Registrars can validate their commands using the OT&E environment, or by using the EPP Frame Validator on the Registrar Console.

10.11. Message Queue

EPP provides a message queue system to provide registrars with notifications for out-of-band events. Registrars can choose to enable or disable the use of the

message queue via the Account Settings page of the Registrar Console. The Registrar Console also provides access to the message queue.

CentralNic currently supports the following EPP messages:

1. Notification of approved inbound transfer

2. Notification of rejected inbound transfer

3. Notification of new outbound transfer

4. Notification of cancelled outbound transfer

5. Status change of a Launch Application

6. Status change of a domain on the `pendingCreate` status

7. Purge of a domain on the `pendingDelete` status

Example frames for these events are given below.

10.11.1.　　　Notification of approved inbound transfer

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully;
        ack to dequeue.</msg>
    </result>
    <msgQ count="1" id="12345" />
    <resData>
      <domain:trnData
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.uk.com</domain:name>
        <domain:trStatus>clientApproved</domain:trStatus>
        <domain:reID>H12345</domain:reID>
        <domain:reDate>2011-01-27T23:50:00.0Z</domain:reDate>
        <domain:acID>H54321</domain:acID>
        <domain:acDate>2011-02-01T23:50:00.0Z</domain:acDate>
      </domain:trnData>
    </resData>
    <trID>
      <clTRID>abc123</clTRID>
      <svTRID>321cba</svTRID>
    </trID>
```

```
    </response>
</epp>
```

Note that the <domain:trStatus> element may also have the value "serverApproved".

10.11.2.   Notification of rejected inbound transfer

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully;
        ack to dequeue.</msg>
    </result>
    <msgQ count="1" id="12345" />
    <resData>
      <domain:trnData
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.uk.com</domain:name>
        <domain:trStatus>clientRejected</domain:trStatus>
        <domain:reID>H12345</domain:reID>
        <domain:reDate>2011-01-27T23:50:00.0Z</domain:reDate>
        <domain:acID>H54321</domain:acID>
        <domain:acDate>2011-02-01T23:50:00.0Z</domain:acDate>
      </domain:trnData>
    </resData>
    <trID>
      <clTRID>abc123</clTRID>
      <svTRID>321cba</svTRID>
    </trID>
  </response>
</epp>
```

Note that the <domain:trStatus> element may also have the value "serverRejected".

10.11.3.   Notification of new outbound transfer

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully;
        ack to dequeue.</msg>
    </result>
    <msgQ count="1" id="12345" />
    <resData>
      <domain:trnData
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.uk.com</domain:name>
```

```
        <domain:trStatus>pending</domain:trStatus>
        <domain:reID>H12345</domain:reID>
        <domain:reDate>2011-01-27T23:50:00.0Z</domain:reDate>
        <domain:acID>H54321</domain:acID>
        <domain:acDate>2011-02-01T23:50:00.0Z</domain:acDate>
      </domain:trnData>
    </resData>
    <trID>
      <clTRID>abc123</clTRID>
      <svTRID>321cba</svTRID>
    </trID>
  </response>
</epp>
```

10.11.4.   Notification of cancelled outbound transfer

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully;
        ack to dequeue.</msg>
    </result>
    <msgQ count="1" id="12345" />
    <resData>
      <domain:trnData
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.uk.com</domain:name>
        <domain:trStatus>clientCancelled</domain:trStatus>
        <domain:reID>H12345</domain:reID>
        <domain:reDate>2011-01-27T23:50:00.0Z</domain:reDate>
        <domain:acID>H54321</domain:acID>
        <domain:acDate>2011-02-01T23:50:00.0Z</domain:acDate>
      </domain:trnData>
    </resData>
    <trID>
      <clTRID>abc123</clTRID>
      <svTRID>321cba</svTRID>
    </trID>
  </response>
</epp>
```

10.11.5.   Status change of a Launch Application

The following is an example poll message for a Launch Application that has transitioned to the "pendingAllocation" state.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
```

```
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully; ack to dequeue</msg>
    </result>
    <msgQ count="5" id="12345">
      <qDate>2013-04-04T22:01:00.0Z</qDate>
      <msg>Application pendingAllocation.</msg>
    </msgQ>
    <resData>
      <domain:infData
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:roid>EXAMPLE1-REP</domain:roid>
        <domain:clID>ClientX</domain:clID>
      </domain:infData>
    </resData>
    <extension>
      <launch:infData
        xmlns:launch="urn:ietf:params:xml:ns:launch-1.0">
        <launch:phase>sunrise</launch:phase>
        <launch:applicationID>abc123</launch:applicationID>
        <launch:status s="pendingAllocation" />
      </launch:infData>
    </extension>
    <trID>
      <clTRID>ABC-12345</clTRID>
      <svTRID>54322-XYZ</svTRID>
    </trID>
  </response>
</epp>
```

The following is an example <domain:panData> poll message for an "allocated"
Launch Application.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully; ack to dequeue</msg>
    </result>
    <msgQ count="5" id="12345">
      <qDate>2013-04-04T22:01:00.0Z</qDate>
      <msg>Application successfully allocated.</msg>
    </msgQ>
    <resData>
      <domain:panData
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name paResult="1">example.tld</domain:name>
        <domain:paTRID>
          <clTRID>ABC-12345</clTRID>
```

```
            <svTRID>54321-XYZ</svTRID>
          </domain:paTRID>
          <domain:paDate>2013-04-04T22:00:00.0Z</domain:paDate>
        </domain:panData>
      </resData>
      <extension>
        <launch:infData
          xmlns:launch="urn:ietf:params:xml:ns:launch-1.0">
          <launch:phase>sunrise</launch:phase>
          <launch:applicationID>abc123</launch:applicationID>
          <launch:status s="allocated" />
        </launch:infData>
      </extension>
      <trID>
        <clTRID>BCD-23456</clTRID>
        <svTRID>65432-WXY</svTRID>
      </trID>
    </response>
</epp>
```

> ***Important Note:*** *as per the specification for the Launch extension (see Section* [*Launch Extension*](#)*), the message has a different format if the application has entered a "final" status (either "allocated" or "rejected"). For these statuses, the <domain:infData> element is omitted, and the <domain:panData> is included instead.*

10.11.6.   Status change of a domain on the `pendingCreate` status

For further information, please see Section [Launch Extension](#).

10.11.7.   Status change of a domain on the `pendingDelete` status

When a domain is purged, our system will look in the EPP transaction log for the most recent successful (response code 1001) <delete> command for the domain, submitted by the current sponsoring registrar. If a log entry is found, a message will be queued to confirm that the domain has finally been purged.

The following is an example message:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1301">
      <msg>Command completed successfully; ack to dequeue</msg>
    </result>
```

```
<msgQ count="5" id="12345">
  <qDate>2016-04-04T22:01:00.0Z</qDate>
  <msg>Pending action completed successfully.</msg>
</msgQ>
<resData>
  <domain:panData
   xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
    <domain:name paResult="1">example.uk.com</domain:name>
    <domain:paTRID>
      <clTRID>ABC-12345</clTRID>
      <svTRID>54321-XYZ</svTRID>
    </domain:paTRID>
    <domain:paDate>2016-04-04T22:00:00.0Z</domain:paDate>
  </domain:panData>
</resData>
<trID>
  <clTRID>BCD-23456</clTRID>
  <svTRID>65432-WXY</svTRID>
</trID>
  </response>
</epp>
```

10.12. Extensions

CentralNic's EPP system supports a number of extensions. These are listed below.

10.12.1.   Registry Grace Period Mapping

This extension (described in RFC 3915) implements the "grace period" policies defined by ICANN. Grace period policies exist to allow protocol actions to be reversed or otherwise revoked during a short period of time after the protocol action has been performed.

The grace periods supported are fully described in Section 10.1.5. The state diagram below describes how the RGP state of a domain changed during the RGP lifecycle.
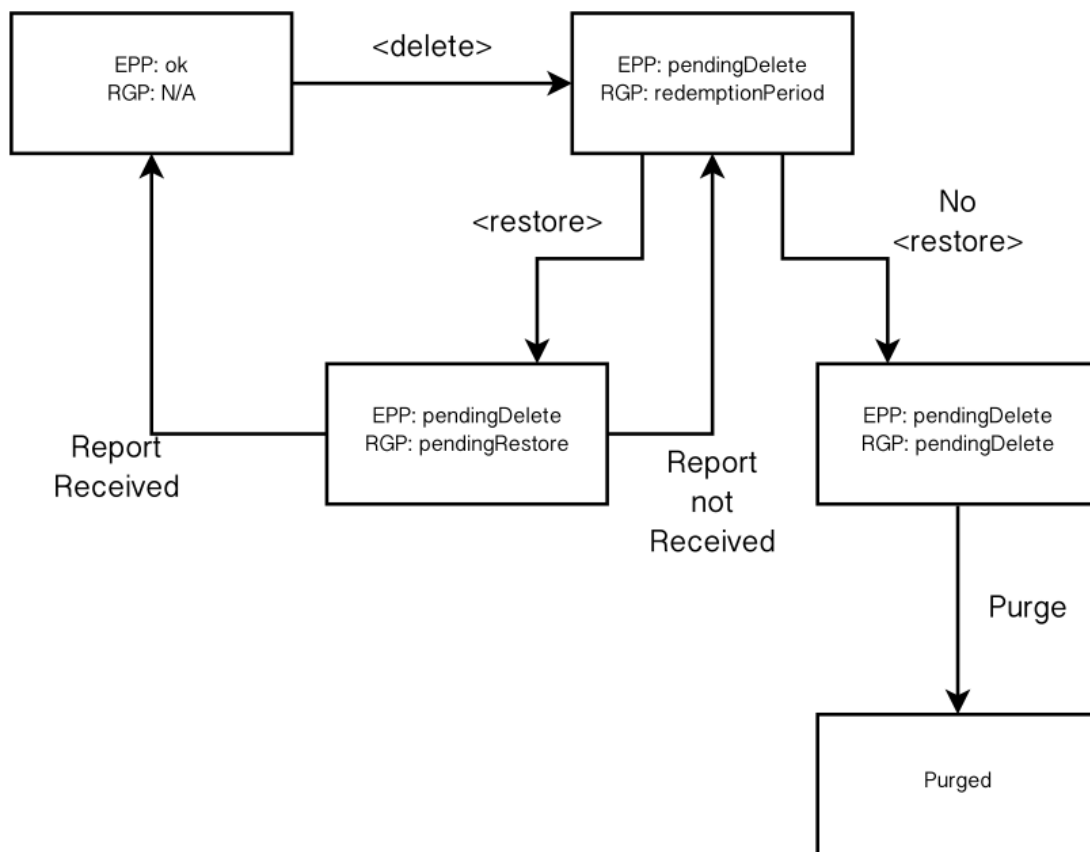
Figure 2: RGP State Diagram

10.12.1.1. Domain Info Command

If a domain name has an RGP status, then the EPP <info> response frame is extended to describe the status. An example response frame is below.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <response>
    <result code="1000">
      <msg>Command completed successfully</msg>
    </result>
    <resData>
      <domain:infData
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.com</domain:name>
        ...
      </domain:infData>
    </resData>
    <extension>
      <rgp:infData xmlns:rgp="urn:ietf:params:xml:ns:rgp-1.0">
```

```
        <rgp:rgpStatus s="addPeriod" />
      </rgp:infData>
    </extension>
    <trID>
      <clTRID>ABC-12345</clTRID>
      <svTRID>54322-XYZ</svTRID>
    </trID>
  </response>
</epp>
```

10.12.1.2.　Restoring a domain on the redemptionPeriod (Redemption Grace Period) status

If a domain name has the redemptionPeriod RGP status, then it can be restored from Pending Delete using an extended <update> command. An example is below.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <update>
      <domain:update
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.com</domain:name>
        <domain:chg />
      </domain:update>
    </update>
    <extension>
      <rgp:update xmlns:rgp="urn:ietf:params:xml:ns:rgp-1.0">
        <rgp:restore op="request" />
      </rgp:update>
    </extension>
    <clTRID>ABC-12345</clTRID>
  </command>
</epp>
```

10.12.1.2.1　Restore Reports

CentralNic does not currently require registrars to submit a "restore report" as part of an RGP restore.

Important Note: depending on TLD policy, registrars may be charged a fee to restore a domain.

10.12.2.　DNSSEC Extension

This extension (described in RFC 5910) provides an interface for registrars to submit DNSSEC Delegation Signer (DS) records for domain names in the registry. These are then published in the DNS alongside the delegation for the domain, allowing a resolver to establish a chain of trust from the root zone to the domain.

CentralNic's implementation of this extension uses the "thin" DS Data Interface, rather than the "thick" Key Data Interface (see RFC 5910, §4). Registrars should insert the <secDNS:dsData> (and its child elements) element into <create> and <update> commands. If the <secDNS:keyData> element is included, it will be stored, but is not validated or used.

The <secDNS:maxSigLife> element is not supported and will be ignored.

10.12.3.   IDN Extension

CentralNic supports the extension for supplying IDN table information specified at:

> http://tools.ietf.org/html/draft-obispo-epp-idn-03

Use of this extension is OPTIONAL: it is not required for non-IDN domains. However, if the client includes the XML namespace for this extension in the <svcExtension> element of its login frame, it MUST be used when submitting a <create> command for an IDN domain.

An example of an extended EPP <create> command is below.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>xn--espaol-zwa.example.com</domain:name>
        <domain:period unit="y">1</domain:period>
        <domain:registrant>jd1234</domain:registrant>
        <domain:contact type="admin">sh8013</domain:contact>
        <domain:contact type="tech">sh8013</domain:contact>
        <domain:authInfo>
          <domain:pw>2fooBAR</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
    <extension>
      <idn:data xmlns:idn="urn:ietf:params:xml:ns:idn-1.0">
        <idn:table>es</idn:table>
        <idn:uname>espa&#241;ol.example.com</idn:uname>
```

```
        </idn:data>
      </extension>
      <clTRID>123456</clTRID>
   </command>
</epp>
```

The <idn:uname> element is optional, but if included, MUST match the A-label in the <domain:name> element.

10.12.3.1.　　IDN Table codes

CentralNic maintains a list of the supported IDN tables at the following URL. This page also contains links to the IDN tables themselves, which can be downloaded in a machine-readable format. The page also provides the mnemonic codes for each table.

https://centralnicregistry.com/support/information/idn

10.12.4.　　Launch Extension

The Launch Extension is used to manage the release of new domains during Sunrise and Landrush, and the Trademark Claims period for gTLDs. It is specified at the following URL:

http://tools.ietf.org/html/draft-tan-epp-launchphase

Unless otherwise noted, all sunrise and landrush phases involve the creation of "Launch Applications": multiple requests for the same domain are accepted, and are resolved using an out-of-band process, typically an auction.

10.12.4.1.   Launch Phases

10.12.4.1.1   Sunrise

During a TLD's Sunrise Phase, the registry accepts domain name registration requests on the basis of a claim relating to a registered trademark. A third-party agent validates these requests before the domains are allocated.

The Sunrise Phase does not (usually) operate on a "first-come, first-served" basis: instead, all registration requests are queued in the registry database and processed at the end of the phase. All unique domain name registrations are registered, but all duplicate domain names are sent to auction, in which the registrants who submitted the request are invited to participate.

The validation agent must successfully validate all requests submitted during the Sunrise Phase. Depending on the policy established for the TLD, registrars may be required (or choose to) establish a direct relationship with the organisation (such as the Trademark Clearinghouse) selected as the validation agent for the launch and "pre-validate" their domain requests, or they may (depending on the policy) be able to submit requests directly to the registry.

Note: for all gTLD sunrises, only the Trademark Clearinghouse will be used.

10.12.4.1.2   Landrush

The Landrush Phase operates under a similar model to the Sunrise Phase, except that registrants may submit requests for any domain name (that was not previously reserved or registered during the Sunrise Process), and trademark information is not required.

Like the Sunrise Phase, the Landrush Phase does not (normally) operate on a "first-come, first-served" basis: instead, all registration requests are queued in the registry database and processed at the end of the phase. All unique domain name registrations are registered, but all duplicate domain names are sent to auction, in which the registrants who submitted the request are invited to participate.

10.12.4.1.3   Other Phases

Depending on the policy for the domain extension, other phases may also be operated. These include additional sunrise phases for different classes of entitlement

or "grandfathering" phases where registrants of domains in different TLDs may obtain the identical domain in the new extension.

Please consult the relevant registration policy document for details of any additional launch phases.

10.12.4.1.4   General Availability

Once the Landrush Phase has ended, the domain extension goes into "General Availability", and registrars may freely submit domain registrations, which are allocated on a "first-come, first-served" basis. This is the normal operating mode for the TLD thereon.

10.12.4.2.   Trademark Claims

The Trademark Claims period may, in some circumstances, overlap the Landrush phase. The Trademark Claims period is Rights Protection Mechanism that is mandatory for all gTLDs. During the Trademark Claims period, which is a 60-day window following the completion of the Sunrise, registrars are required to present a notice to a prospective registrant if the domain name matches a label in the Clearinghouse. The registrar must capture the acknowledgement of this notice and provide it to the registry at the point of registration.
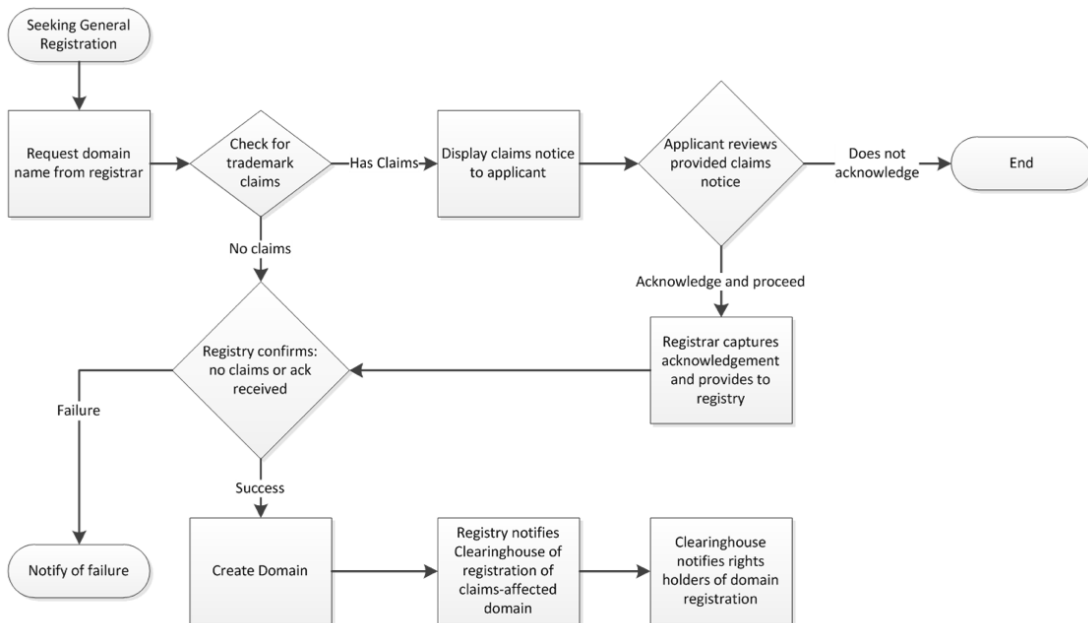
Figure 3: Trademark Claims Process

10.12.4.3. Sunrise Validation Models

Depending on the policy of the TLD, the following sunrise validation models will be supported:

- **Signed Mark Data:** this model is mandatory for all gTLD sunrises. Registrars must obtain a Signed Mark Data (SMD) file from the Trademark Clearinghouse and include this file in the <domain:create> command.

The EPP system will accept either a <smd:signedMark> or <smd:encodedSignedMark> element. An Encoded Signed Mark Data element will be decoded and displayed as XML in subsequent <info> responses.

> *Important Note: XML Digital Signatures are sensitive to whitespace, so to avoid any issues where your XML parser fails to preserve whitespace, you may wish to only use the <smd:encodedSignedMark> element.*

- **Code:** for non-gTLD sunrises, the validation agent may issue a code or key, which is typically an ASCII hexadecimal token.

- **Code With Mark:** for non-gTLD sunrises, which have additional eligibility requirements, it may be necessary to supply mark data in addition to the code.

- **Code Or Mark:** this model supports non-gTLD sunrises, which may allow the registrar to submit trademark data directly to the registry, which acts as a proxy to the validation agent (or which performs the validation itself).

Please check which validation models are supported for sunrise before participating. As mentioned above, all gTLD sunrises use the Trademark Clearinghouse and the Signed Mark Data model.

10.12.4.4.   Timeline of Sunrise and Landrush Phases
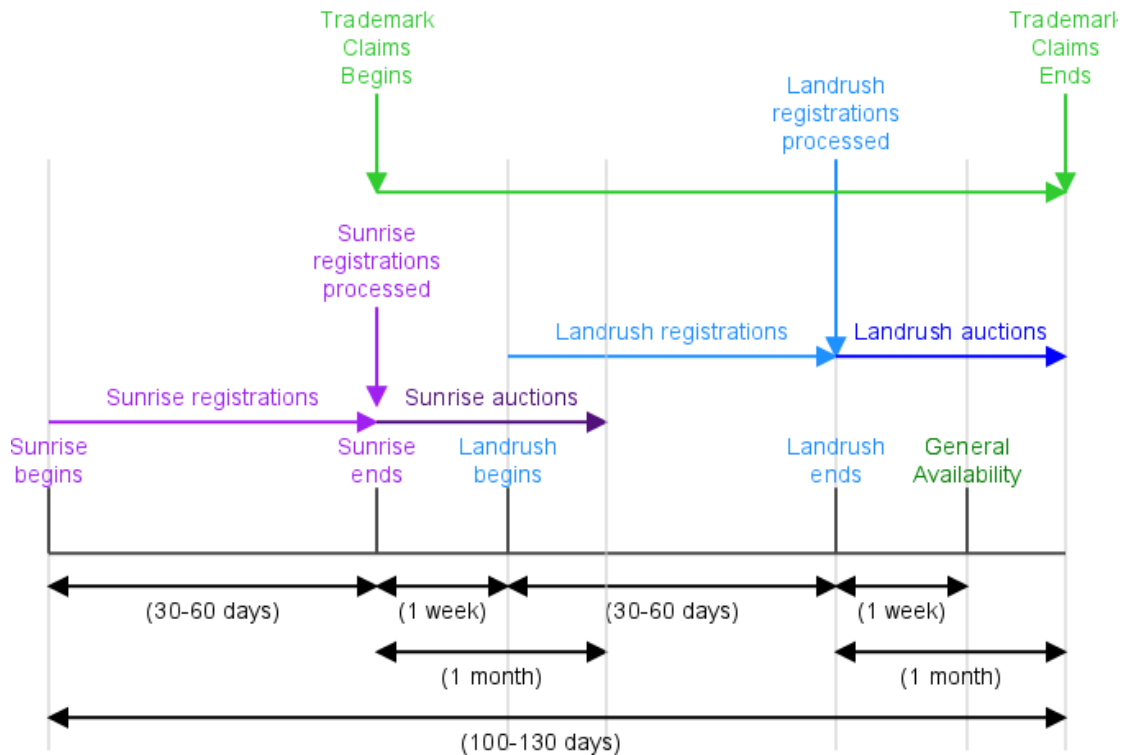
Figure 4: TLD Launch Timeline

Note: the times in the diagram above are approximate and will vary depending on the TLD.

Typically, a "quiet period" is added between each phase. During this period, the registry remains functioning normally (other TLDs will continue to be available) and registrars may freely create contact objects and out-of-bailiwick host objects in anticipation of the start of the next phase.

10.12.4.5.   End-of-Phase Processing

Once a phase has closed, the applications received during the phase will be processed.

● If only a single request for a domain has been received, then the domain is allocated and an EPP message will be sent to the registrar (see Section Message Queue).

● If more than one request is received, then the domain is reserved. At this point, all the applicants for the domain will be invited to resolve the

contention set by means of an auction. The application for each losing bidder will be marked as "rejected". The winning bidder's application will be marked as "allocated". In all cases, the registrar receives an EPP message as per Section 11.10.5).

10.12.4.6.   Example EPP Command Frames

10.12.4.6.1   Sunrise Create (with Code validation)

```xml
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:registrant>jd1234</domain:registrant>
        <domain:contact type="admin">sh8013</domain:contact>
        <domain:contact type="tech">sh8013</domain:contact>
        <domain:authInfo>
          <domain:pw>2fooBAR</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
    <extension>
      <launch:create
        xmlns:launch="urn:ietf:params:xml:ns:launch-1.0">
        <launch:phase>sunrise</launch:phase>
        <launch:codeMark>
          <launch:code>49FD46E6C4B45C55D4AC</launch:code>
        </launch:codeMark>
      </launch:create>
    </extension>
    <clTRID>ABC-12345</clTRID>
  </command>
</epp>
```

10.12.4.6.2   Sunrise Create (with SMD validation)

```xml
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:registrant>jd1234</domain:registrant>
```

```
        <domain:contact type="admin">sh8013</domain:contact>
        <domain:contact type="tech">sh8013</domain:contact>
        <domain:authInfo>
          <domain:pw>2fooBAR</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
    <extension>
      <launch:create
        xmlns:launch="urn:ietf:params:xml:ns:launch-1.0">
        <launch:phase>sunrise</launch:phase>
        <smd:encodedSignedMark
          xmlns:smd="urn:ietf:params:xml:ns:signedMark-1.0">
          PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4KP
          ...
          Xlgo8L3NtZDpzaWduZWRNYXJrPgo=</smd:encodedSignedMark>
      </launch:create>
    </extension>
    <clTRID>ABC-12345</clTRID>
  </command>
</epp>
```

10.12.4.6.3　Landrush Create (with Trademark Claims)

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:registrant>jd1234</domain:registrant>
        <domain:contact type="admin">sh8013</domain:contact>
        <domain:contact type="tech">sh8013</domain:contact>
        <domain:authInfo>
          <domain:pw>2fooBAR</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
    <extension>
      <launch:create
        xmlns:launch="urn:ietf:params:xml:ns:launch-1.0">
        <launch:phase name="landrush">claims</launch:phase>
        <launch:notice>
          <launch:noticeID>
            49FD46E6C4B45C55D4AC
          </launch:noticeID>
          <launch:notAfter>
            2012-06-19T10:00:10.0Z
          </launch:notAfter>
```

```
            <launch:acceptedDate>
               2012-06-19T09:01:30.0Z
            </launch:acceptedDate>
         </launch:notice>
      </launch:create>
    </extension>
    <clTRID>ABC-12345</clTRID>
  </command>
</epp>
```

10.12.4.6.4   Landrush Create (without Trademark Claims)

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:registrant>jd1234</domain:registrant>
        <domain:contact type="admin">sh8013</domain:contact>
        <domain:contact type="tech">sh8013</domain:contact>
        <domain:authInfo>
          <domain:pw>2fooBAR</domain:pw>
        </domain:authInfo>
      </domain:create>
    </create>
    <extension>
      <launch:create
        xmlns:launch="urn:ietf:params:xml:ns:launch-1.0">
        <launch:phase>landrush</launch:phase>
      </launch:create>
    </extension>
    <clTRID>ABC-12345</clTRID>
  </command>
</epp>
```

10.12.4.6.5   General Availability Create (with Trademark Claims)

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <create>
      <domain:create
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example.tld</domain:name>
        <domain:registrant>jd1234</domain:registrant>
        <domain:contact type="admin">sh8013</domain:contact>
```

```
            <domain:contact type="tech">sh8013</domain:contact>
            <domain:authInfo>
              <domain:pw>2fooBAR</domain:pw>
            </domain:authInfo>
          </domain:create>
      </create>
      <extension>
        <launch:create
          xmlns:launch="urn:ietf:params:xml:ns:launch-1.0">
          <launch:phase>claims</launch:phase>
          <launch:notice>
            <launch:noticeID>
              49FD46E6C4B45C55D4AC
            </launch:noticeID>
            <launch:notAfter>
              2012-06-19T10:00:10.0Z
            </launch:notAfter>
            <launch:acceptedDate>
              2012-06-19T09:01:30.0Z
            </launch:acceptedDate>
          </launch:notice>
        </launch:create>
      </extension>
      <clTRID>ABC-12345</clTRID>
   </command>
</epp>
```

10.12.4.6.6   Claims Check Command

This command retrieves the lookup key required to obtain the Claims Notice Information (CNI) from the Trademark Clearinghouse.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
  <command>
    <check>
      <domain:check
        xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
        <domain:name>example1.tld</domain:name>
      </domain:check>
    </check>
    <extension>
      <launch:check
        xmlns:launch="urn:ietf:params:xml:ns:launch-1.0"
        type="claims">
        <launch:phase>claims</launch:phase>
      </launch:check>
    </extension>
    <clTRID>ABC-12345</clTRID>
```

```
    </command>
</epp>
```

10.12.4.7.   OT&E Testing

Due to the limited number of SMD files issued by the TMCH OT&E environment, there is a workaround in place to purge domain names registered using OT&E SMD files. Please see Section **OT&E Testing of TMCH Sunrise and EPP Launch Extension** for further details.

10.12.5.   Fee Extension

The Fee extension can be used to query for the fee associated with a given transaction for a domain name and command. The extension is specified at:

> http://tools.ietf.org/html/draft-brown-epp-fees-02

10.12.5.1.   Use of the Fee Extension with Premium Domain Names

The Fee extension MUST be used when performing a billable transaction with premium domain names. A premium domain is defined as any domain that is not billed at the base price for registrations, renewals and transfers.

The following rules apply to premium names:

- Registrars who wish to register or manage premium names MUST include the XML namespace of the Fee extension in the <login> frame of a new EPP session.

- Participating registrars MUST include the applicable extension element in all billable transactions for premium domains. These commands are:

    o <create>

    o <transfer> (with op="request")

    o <renew>

See the URL above for examples of how to extend these command frames to include the required element.

- The value of the <fee:fee> element MUST match the value stored in the registry database for this domain. If the value does not match, then the transaction will be rejected.

- Registrars who do not wish to support premium names can simply ignore or omit the Fee extension from their command frames, and our system will reject any billable commands for premium domains.

10.12.5.2.   Use of the Fee Extension for Standard Domain Names

Registrars are not required to use the fee extension to submit fee information for standard price domain names. However, if the fee information *is* submitted, it MUST be correct.

If a domain is subject to a promotional discount, then this discount must be taken into account when calculating the fee submitted in the <create> command. Please see Section Promotional Pricing for further details.

10.12.5.3.   Supported Fee Descriptions and Domain Classes

Our server supports the following values for the "description" attribute of <fee:fee> elements:

Value	Description
"Application Fee"	The fee charged in addition to the registration fee for domains applied for during sunrise or landrush phases.
"Registration Fee"	The fee charged for the first year of registration.
"Renewal Fee"	The fee charged for all years subsequent to the first year.
"Restore Fee"	The fee charged to restore a domain from the Redemption Grace Period.

In <check> and <info> responses, our server will use the following values for the <fee:class> element:

Class	Description
standard	the domain name has the standard price

premiu m	the domain has a non-standard premium price

10.12.5.3.1   Determining whether a domain name has a premium price

Registrars who use the Fee extension can determine whether a domain has a premium price by using the <fee:class> element described in Section 11.11.5.3 above. If the value of this element is "premium", then the domain is a premium domain.

We also publish complete lists of all premium domains via FTP: please see Section 6.3.2.3 for further information. If a domain appears in a pricing file, then it has a premium price.

10.12.5.4.   OT&E Testing

Premium pricing data is replicated to the OT&E system, so registrars can test their client implementation using the premium pricing data obtained from the production data files.

11. Additional Registry Services

11.1. Registry Lock Service

The Registry Lock Service provides an additional layer of protection for "high-value domains", protecting them from unauthorised modification as a result of a hijacked customer account, or a compromise of a registrar's website or another technical system.

Under the service, domain names are placed in a "Registry Lock" which prohibits any updates by the registrar. When a change is required, the registrar must submit a request for the domain to be unlocked: this request is authenticated out-of-band using pre-established communication channels.

CentralNic charges a per-domain fee to use this service. Please contact your account manager for details.

11.1.1. Procedure

1. The registrar provides an initial list of domain names to be placed under a lock and one or more Points of Contact (POC) who must authenticate unlock requests. Additional domains may be added to the list by submitting a support ticket.

2. The registry will then place these domains under Registry Lock. This will result in the following EPP status codes being applied to the domain:

   - serverUpdateProhibited
   - serverTransferProhibited
   - serverDeleteProhibited

3. Once locked, any attempt by the registrar to update, renew or delete the domain will be refused by the registry, as will any attempt by another registrar to transfer it.

4. When the registrar wishes to change a domain that is locked or remove a domain from the list, they must submit a support ticket to request that the domain be unlocked.

5. The registry will authenticate the request by contacting the registrar by telephone and verifying the details of the request. Once verified, the lock will be removed from the domain.

6. The registrar may then make the changes required via the Registrar Console or EPP system.

7. When the required changes have been made, the registrar submits a further support ticket to request that the domain be re-locked. The registry will automatically re-lock the domain after 24 hours.

11.1.2. Adding a Registry Lock via the Registrar Console

Registrars can *add* a registry lock for domains that are eligible for this service via the "Manage Domains" page of the Registrar Console. Simply click the "Add Registry Lock" button on the domain information page.

Obviously, these status codes cannot be *removed* from a domain that already has them (as this would defeat the point of the Registry Lock service). Use the procedure described above to remove a lock.

11.1.3. Adding a Registry Lock via EPP

Submit a <update> command to add the following three status codes:

- `serverDeleteProhibited`
- `serverTransferProhibited`
- `serverUpdateProhibited`

Since the registry lock service is subject to a fee, registrars must request the EPP Fee Extension (see Section 11.11.5) on login for the registry lock to be applied. If the fee extension is not used on the update command, the lock will be applied and the cost returned in the response frame. If the fee information is submitted, it MUST be correct. An example command to add a lock with the fee extension:

```
<?xml version="1.0" encoding="UTF-8"?>
 <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
   <command>
     <update>
       <domain:update
         xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
         <domain:name>mary-testing.online</domain:name>
```

```
        <domain:add>
          <domain:status s="serverUpdateProhibited"/>
          <domain:status s="serverTransferProhibited"/>
          <domain:status s="serverDeleteProhibited"/>
        </domain:add>
      </domain:update>
    </update>
    <extension>
      <fee:updData xmlns:fee="urn:ietf:params:xml:ns:fee-0.5">
        <fee:currency>USD</fee:currency>
       <fee:fee>500</fee:fee>
      </fee:updData>
    </extension>
    <clTRID>ABC-12345</clTRID>
  </command>
</epp>
```

11.1.4. Billing

Registrars will be charged on a monthly basis for each domain that is subject to a registry lock. These fees will be added to the standard monthly invoice.