

EOI (Expression of Interest) and DomainApplication (Pre-Registration) Reference

Version 1.1
October 12, 2012

Proprietary Information

This document is our property. It may be used by recipient only for the purpose for which it was transmitted and shall be returned upon request or when no longer needed by recipient. It may not be copied or communicated without the prior written consent of us.

COPYRIGHT NOTIFICATION

Copyright © 2012. All rights reserved.

DISCLAIMER AND LIMITATION OF LIABILITY

We made efforts to ensure the accuracy and completeness of all information in this document. However, we make no warranties of any kind (whether express, implied or statutory) with respect to the information herein. We assume no liability to any party for loss or damage (whether direct or indirect) caused by any errors, omissions or statements of any kind contained in this document. Further, we assume no liability arising from the application or use of the product or service described herein and specifically disclaims any representation that the products or services described herein do not infringe upon any existing or future intellectual property rights. Nothing herein grants the reader any license to make, use, or sell equipment or products constructed in accordance with this document. Finally, all rights and privileges related to any intellectual property right described herein are vested in the patent, trademark, or service mark owner, and no other person may exercise such rights without express permission, authority, or license secured from the patent, trademark, or service mark owner.

We reserve the right to make changes to any information herein without further notice.

NOTICE AND CAUTION concerning Patent or Trademark Rights

The inclusion in this document, the associated on-line file, or the associated software of any information covered by any patent, trademark, or service mark rights shall not constitute nor imply a grant of, or authority to exercise, any right or privilege protected by such patent, trademark, or service mark. All such rights and privileges are vested in the patent, trademark, or service mark owner, and no other person may exercise such rights without express permission, authority, or license secured from the patent, trademark, or service mark owner.

EOI and DomainApplication Manual 1.1
October 12, 2012

Table of Contents

1 INTRODUCTION	4
1.1 WHAT IS AN EOI AND A DOMAINAPPLICATION	5
1.2 PREPAID DEPOSITS AND REFUNDS FOR EOIS / DOMAINAPPLICATIONS	5
1.3 MULTIPLE DOMAINAPPLICATIONS FOR THE SAME DOMAIN NAME	6
1.4 PREMIUMSALE.COM FOR DOMAIN AUCTIONS	6
1.5 PROCESSING OF A DOMAINAPPLICATION AFTER SUCCESSFUL REGISTRATION	7
2 WORKFLOW AND PROCESSES	7
2.1 EOI / DOMAINAPPLICATION LIFE CYCLE	7
2.2 APPLICATION CLASS	8
2.3 ENABLE EOIS/DOMAINAPPLICATIONS FOR SUBUSERS	10
2.4 CREATING A NEW EOI	11
2.5 STATUS OF AN EOI	11
2.6 MODIFYING AN EOI	12
2.7 CONVERTING AN EOI INTO A DOMAINAPPLICATION	12
2.8 DELETING AN EOI	13
2.9 CREATING A NEW DOMAINAPPLICATION	13
2.10 STATUS OF A DOMAINAPPLICATION	13
2.11 MODIFYING A DOMAINAPPLICATION	14
2.12 DELETING A DOMAINAPPLICATION	14
3 API REFERENCE	15
3.1 COMMAND - CREATE AN EOI/DOMAINAPPLICATION	15
3.2 COMMAND - MODIFYDOMAINAPPLICATION - CONVERT EOI TO A DOMAINAPPLICATION	19
3.3 COMMAND - DELETE AN EOI/DOMAINAPPLICATION	23
3.4 COMMAND - QUERY DETAILS ABOUT A SINGLE EOI / DOMAINAPPLICATION	26
3.5 COMMAND - QUERY LIST ALL EOIS / DOMAINAPPLICATIONS	29

1 INTRODUCTION

In the first quarter of 2012, ICANN opened the newTLD (new Top-Level-DomainApplication) window, thus allowing entities worldwide to apply for their own individual TLD.

The newTLDs have been classified into two (2) different categories:

- 1.) generic newTLDs
- 2.) geo newTLDs

At this point in time, you can submit EOIs (Expression of Interests) for the following newTLDs:

1. generic newTLDs:

.Actor, .Agency, .Android, .Apartments, .App, .Arab, .Art, .Attorney, .Auction, .Audio, .Author, .Auto, .Baby, .Band, .Bar, .Baseball, .Basketball, .Beer, .Bet, .Bio, .Blog, .Book, .Boutique, .Broadway, .Buy, .Cafe, .Cars, .Casa, .Casino, .Charity, .Chat, .Christmas, .City, .Cloud, .Club, .Coffee, .Cool, .Corp, .Coupon, .CPA, .Data, .Deals, .Design, .Diet, .Discount, .Doctor, .Earth, .Eco, .Fail, .Family, .Fan, .Fashion, .Film, .Fitness, .Flowers, .Food, .Football, .Free, .Gallery, .Game, .Gay, .GmbH, .Golf, .Graphics, .Gratis, .Green, .Health, .Help, .Hiv, .Hockey, .Home, .Hotel, .Inc, .Insurance, .Kids, .Kiw, .Latino, .Law, .Lawyer, .Life, .Live, .LLC, .LLP, .Lol, .Love, .LTD, .Map, .Mail, .Marketing, .Med, .Media, .MLS, .Money, .Movie, .Music, .News, .NGO, .Now, .One, .Online, .Party, .Phone, .Photography, .Play, .Poker, .Property, .Radio, .Realestate, .Reise, .Rent, .Restaurant, .RIP, .Sale, .School, .Search, .Secure, .Sex, .Shop, .Show, .Site, .Ski, .Soccer, .Space, .Sport, .Store, .Style, .Sucks, .Support, .Taxi, .Team, .Tech, .Tennis, .Terra, .Theater, .Tickets, .Today, .Trust, .Tube, .Video, .VIP, .Web, .Website, .Wiki, .Wtf

2. geo newTLDs:

.Abudhabi, .Africa, .Amsterdam, .Barcelona, .Bayern, .Berlin, .Boston, .Brussels, .Budapest, .Capetown, .Cymru, .Dubai, .Durban, .Gent, .Hamburg, .Helsinki, .Istanbul, .Joburg, .Kyoto, .London, .Madrid, .Melbourne, .Miami, .Moscow, .Nagoya, .Nrw, .Nyc, .Okinawa, .Osaka, .Paris, .Quebec, .Roma, .Ruhr, .Scot, .Stockholm, .Sydney, .Taipei, .Tirol, .Tokyo, .Vegas, .Wales, .Wien, .Yokohama, .Zuerich

The above list of newTLDs is not yet final. Depending on the respective Policies of the specific TLD we may be forced to remove the respective nTLD from our DomainApplication product.

If and when a newTLD is approved by ICANN and the TLD is placed in the Internet's Root (required for a TLD's domains to resolve on the Internet), one can choose to convert an EOI into a standard DomainApplication or to delete the EOI completely.

1.1 What is an EOI and a DomainApplication

An EOI for a domain is a free and non-binding Expression of Interest, informing the service provider that the user has interest in a domain name and would like timely information and alerts sent to them. The EOI process was created only for newTLDs as details for many if not most of the newTLDs are still pending. If and when any newTLD gets entered into the Internet's Root, detailed information on that newTLD's respective launch phases (Sunrise A, Sunrise B, Landrush and Go Live, ...etc) will be immediately provided, as well as, information on how to convert an EOI into a standard DomainApplication.

A DomainApplication is exactly the same as a standard pre-registration and goes through the regular deposit/billing cycle. Users may delete a DomainApplication up to 48 hours prior to its submission without being charged.

1.2 Prepaid Deposits and Refunds for EOIs / DomainApplications

EOIs are free-of-charge and can be deleted at any point in time. EOIs that are converted into DomainApplications must then follow the standard deposit/billing cycle for DomainApplications for proper submission of an application.

DomainApplications deposits are withdrawn from user's accounts approximately two (2) weeks prior to their submission dates. Again, deposits are used to ensure funds are available to properly register a DomainApplication in the event it is successful.

Please ensure user account balances are properly funded to cover all pending DomainApplications, otherwise some or all DomainApplication requests will **NOT** be processed.

If a DomainApplication can not be submitted to the registry, reserved deposits are then automatically returned to the respective users account balances.

1.3 Multiple DomainApplications for the Same Domain Name

Some DomainApplications will be first-come-first-served while others will not. In the event that domains with multiple DomainApplications on them go to auction, a PRIVATE Auction which will be conducted by PremiumSale.com. Private auctions at PremiumSale.com are only for those users that originally placed a DomainApplication on the respective domain prior to submission.

1.4 PremiumSale.com for Domain Auctions

PremiumSale.com manages all invitations, registrations, auction proceedings, payment, and even notifications for resellers. For more information on private auctions or on the auction company, please visit www.premiumsale.com.

1.5 Processing of a DomainApplication After Successful Registration

DomainApplications with a single request that are successfully registered by the registration system, will be placed in the reseller's account automatically. Those DomainApplications requiring an auction for settling ownership, the domain will be placed in the reseller's account or the reseller's customer at the conclusion of the auction.

2 WORKFLOW AND PROCESSES

2.1 EOI / DomainApplication Life Cycle

As the EOI is free-of-charge and a non-binding, there is no life cycle for EOIs. The status of an EOI will always be 'REQUESTED'.

The initial status of a DomainApplication will either be 'REQUESTED' or 'INCOMPLETE' no matter if it was newly created or converted from an EOI.

INCOMPLETE:

A submission of a DomainApplication with status 'INCOMPLETE' is not possible. There will be no attempt to withdraw a deposit for a DomainApplication with status 'INCOMPLETE'. Further information has to be provided in order to change the status to 'REQUESTED'.

REQUESTED:

All required parameters have been completed. The next step will be the withdrawal of a deposit approximately two (2) weeks prior to the submission in order to ensure funds are available to properly register the DomainApplication in the event it is successful. If the deposit was withdrawn successfully the status will change to 'ACTIVE'. If the domain application stays in status 'REQUESTED' the reason may be insufficient funds.

ACTIVE:

The DomainApplication is ready for submission. A deletion of a DomainApplication with status 'ACTIVE' will automatically cause the reserved deposit to be released to the respective users account balances.

PROCESSING:

Approx. 48 hour prior to submission the status of the DomainApplication changes to 'PROCESSING'. The submission is now being prepared and it is not possible to modify the DomainApplication any more. A deletion with refund is no longer permitted.

PENDING:

The DomainApplication was submitted successfully to the registry. This is no indication of a successful domain application. The DomainApplication can stay in status 'PENDING' until further information from the registry is received. The status of the DomainApplication can change to either 'SUCCESSFUL', 'FAILED', or 'AUCTION-PENDING'.

SUCCESSFUL:

The domain application was successful and the respective domain has been assigned to your user account.

FAILED:

The domain application was not successful.

AUCTION-PENDING:

There were multiple domain applications for the same domain and the domains were not allocated on a first-come-first-served basis. The domain will be auctioned in a private auction only for those users that originally placed a DomainApplication on the respective domain prior to submission.

AUCTION-WON:

The auction was won

AUCTION-LOST:

The auction was lost

2.2 Application Class

The application class has to be stated when creating an EOI/DomainApplication. It is also required in order to enable EOIs/DomainApplications for subusers, as well as when converting an EOI into a DomainApplication.

For EOIs the application class usually consists of the respective TLD followed

by '_EOI'.

Examples of application classes of EOIs:

```
CLASS = WEB_EOI  
CLASS = BERLIN_EOI
```

If you create a new EOI through the API the preceding TLD in the application class may be omitted as the system will automatically extend it depending on the respective TLD:

Simplified application class:

```
CLASS = EOI
```

For DomainApplications the application class usually consists of the respective TLD followed by a '_' and the desired launch phase. Typical launch phases are SUNRISE1, SUNRISE2, SUNRISE3, LANDRUSH, GOLIVE.

Examples of application classes of DomainApplications:

```
CLASS = SX_GOLIVE  
CLASS = WEB_LANDRUSH  
CLASS = BERLIN_SUNRISE1
```

Depending on the respective registry some launch phases may not be supported or additional launch phases may be offered. The TLD can be omitted when creating new DomainApplications or converting EOIs into DomainApplications. The system will automatically extend the application class based on the respective TLD.

Examples of simplified application classes:

```
CLASS = SUNRISE1  
CLASS = LANDRUSH  
CLASS = GOLIVE
```

Please note: The simplified application class must not be used when enabling EOIs for subusers or when managing price relations for DomainApplications for subusers (see below).

2.3 Enable EOIs/DomainApplications for Subusers

If you manage subusers you can enable EOIs for your subusers by TLD. The following relation has to be defined and set to 0.00 for each TLD that you want to enable:

```
PRICE_CLASS_DOMAIN_<APPLICATIONCLASS>_SETUP:0.00
```

Examples of EOI Relations:

```
PRICE_CLASS_DOMAIN_WEB_EOI_SETUP:0.00  
PRICE_CLASS_DOMAIN_BERLIN_EOI_SETUP:0.00
```

If you do not manage subusers, all EOIs are enabled for your account by default.

If you manage subusers you can also enable DomainApplications for your subusers on a TLD basis and set the respective fees for your subusers. A DomainApplication requires usually a set of four relations:

Relation set for DomainApplications:

```
PRICE_CLASS_DOMAIN_<APPLICATIONCLASS>_SETUP  
PRICE_CLASS_DOMAIN_<APPLICATIONCLASS>_ANNUAL  
PRICE_CLASS_DOMAIN_<APPLICATIONCLASS>_APPLICATION  
PRICE_CLASS_DOMAIN_<APPLICATIONCLASS>_CURRENCY
```

If PERIOD is the requested registration period in years the total fee is calculated as:

```
TOTAL FEE = SETUP + (PERIOD * ANNUAL) + APPLICATION
```

SETUP is a one-time fee, which will only be charged if the DomainApplication is successful.

ANNUAL is the annual fee, which will only be charged if the DomainApplication is successful.

APPLICATION is a non-refundable one-time fee, which will be charged for the successful submission to the registry. Most registries do not charge this fee

Example of a relation set for .SX GoLive DomainApplications:

```
PRICE_CLASS_DOMAIN_SX_GOLIVE_SETUP:0.00  
PRICE_CLASS_DOMAIN_SX_GOLIVE_ANNUAL:24.00  
PRICE_CLASS_DOMAIN_SX_GOLIVE_APPLICATION:0.00  
PRICE_CLASS_DOMAIN_SX_CURRENCY:USD
```

2.4 Creating a New EOI

To create a new EOI, simply use the command below:

```
COMMAND = AddDomainApplication  
CLASS = EOI  
  
Optional Parameters:  
AUTH = <AUTHCODE>  
OWNERCONTACT0 = <CONTACTHANDLE>  
ADMINCONTACT0 = <CONTACTHANDLE>  
TECHCONTACT0 = <CONTACTHANDLE>  
BILLINGCONTACT0 = <CONTACTHANDLE>  
NAMESERVER0 = <NAMESERVER>  
NAMESERVER1 = <NAMESERVER>  
NAMESERVER2 = <NAMESERVER>
```

Parameters used for this command are very similar to the parameters used for creating a new registration via AddDomain, however, in the case of an EOI, most parameters are optional as well as the periods are ignored.

A newly created EOI will have the status REQUESTED and a unique APPLICATION ID.

```
STATUS - REQUESTED  
APPLICATION - 3723
```

2.5 Status of an EOI

As an EOI is free of charge and a non-binding expression of interest it will always have the status REQUESTED:

```
COMMAND = StatusDomainApplication  
APPLICATION = 3723
```

```
STATUS = REQUESTED
```

2.6 Modifying an EOI

Auth code, contact handles and name servers that can be assigned to an EOI or can be changed for any EOI at any point in time:

```
COMMAND = ModifyDomainApplication
APPLICATION = 3723
AUTH = <AUTHCODE>
OWNERCONTACT0 = <CONTACTHANDLE>
ADMINCONTACT0 = <CONTACTHANDLE>
TECHCONTACT0 = <CONTACTHANDLE>
BILLINGCONTACT0 = <CONTACTHANDLE>
NAMESERVER0 = <NAMESERVER>
NAMESERVER1 = <NAMESERVER>
NAMESERVER2 = <NAMESERVER>
```

2.7 Converting an EOI into a DomainApplication

If a newTLD is approved by ICANN and further information about the respective newTLD and the different launch phases is available, it will be possible to convert an EOI into a standard DomainApplication. This is also done with the ModifyDomainApplication command:

```
COMMAND = ModifyDomainApplication
APPLICATION = 3723
CLASS = LANDRUSH
```

Depending on the requirements the status will either stay 'REQUESTED' or change to 'INCOMPLETE'.

```
STATUS - REQUESTED
```

or

```
STATUS - INCOMPLETE
```

Please note that a submission of DomainApplications with status 'INCOMPLETE' is not possible. In that case further information has to be provided in order to change the status to 'REQUESTED'.

2.8 Deleting an EOI

An EOI can be deleted at any point in time with the DeleteDomainApplication command:

```
COMMAND = DeleteDomainApplication
APPLICATION = 3723
```

2.9 Creating a New DomainApplication

A new DomainApplication requires domain name, application class, period, and contact handles:

```
COMMAND = AddDomainApplication
CLASS = SX_GOLIVE
DOMAIN = example.sx
PERIOD = 1Y
OWNERCONTACT0 = <CONTACTHANDLE>
ADMINCONTACT0 = <CONTACTHANDLE>
TECHCONTACT0 = <CONTACTHANDLE>
BILLINGCONTACT0 = <CONTACTHANDLE>
```

Optional:

```
NAMESERVER0 = <NAMESERVER>
NAMESERVER1 = <NAMESERVER>
NAMESERVER2 = <NAMESERVER>
```

2.10 Status of a DomainApplication

A domain application can have a different status depending on the circumstances. The current status of a DomainApplication can be queried with the StatusDomainApplication command:

```
COMMAND = StatusDomainApplication
APPLICATION = 4678
```

```
STATUS - REQUESTED
```

The meaning of the different status values for DomainApplications is described below:

INCOMPLETE - required parameters are missing, submission not possible
REQUESTED - required parameters complete, deposit not yet withdrawn or not enough credit
ACTIVE - deposit withdrawn successfully, waiting for submission to be prepared
PROCESSING - submission has been prepared, waiting for submission
PENDING - application has been submitted
SUCCESSFUL - domain application was successful
FAILED - domain application was not successful
AUCTION-PENDING - multiple domain applications are being auctioned
AUCTION-WON - auction was won
AUCTION-LOST - auction was lost

2.11 Modifying a DomainApplication

DomainApplications can be modified up to 48 hours prior to their submission. Auth code, contact handles and name servers that can added or modified:

```
COMMAND = ModifyDomainApplication
APPLICATION = 4678
OWNERCONTACT0 = (CONTACT)
ADMINCONTACT0 = (CONTACT)
TECHCONTACT0 = (CONTACT)
BILLINGCONTACT0 = (CONTACT)
NAMESERVER0 = (NAMESERVER)
NAMESERVER1 = (NAMESERVER)
NAMESERVER2 = (NAMESERVER)
```

2.12 Deleting a DomainApplication

DomainApplications can be deleted without charge up to 48 hours prior to their submission.

```
COMMAND = DeleteDomainApplication
APPLICATION = 4678
```

3 API REFERENCE

This API command reference below details the API commands for the EOI / DomainApplication Service. The API itself is NOT object oriented but is objected related and can be encapsulated into various Object-Oriented languages.

Please adhere to the following BOX COLORS:

GREEN – Request and Response Parameters

BLUE – Example Command Use

GREY – EPP Request and Response if Applicable

3.1 COMMAND - Create an EOI/DomainApplication

Request Parameters:

Required:

COMMAND = AddDomainApplication
CLASS = EOI
DOMAIN = <DOMAIN NAME>

Optional:

AUTH = AUTH code for the domain name (NOTE 1)
OWNERCONTACT0 = contact handle ID of registrant
ADMINCONTACT0 = contact handle ID of the administrative contact
TECHCONTACT0 = contact handle ID of the technical contact
BILLINGCONTACT0 = contact handle ID of the billing contact
NAMESERVER0 = nameserver #1
NAMESERVER1 = nameserver #2
...
NAMESERVER12 = nameserver #13

Note:

(1) If the EOI is converted to a DomainApplication and the DomainApplication has been completed successfully this AUTH code will be set for this domain

Response Parameters:

Parameters:

CLASS = application class (syntax: <TLD>_EOI)
APPLICATION = unique application ID
STATUS = application status (NOTE 1)
DOMAIN = domain name
USER = user id the DomainApplication is assigned to
PEERUSER = User ID hierarchy (tree) that DomainApplication is assigned to
CREATEDDATE = application created date
UPDATEDDATE = last application updated date
AUTH = AUTH code for the domain name (NOTE 2)

Note:

(1) Possible values are:

- INCOMPLETE - required parameters are missing, submission not possible
- REQUESTED - final status for EOIs, deposit not yet withdrawn for applications
- ACTIVE - deposit withdrawn successfully, waiting for submission to be prepared
- PROCESSING - submission prepared, waiting for submission
- PENDING - submitted, waiting for results
- SUCCESSFUL - DomainApplication successful
- FAILED - DomainApplication failed
- AUCTION-PENDING - multiple DomainApplications are being auctioned
- AUCTION-WON - auction was won
- AUCTION-LOST - auction was lost

(2) If the EOI is converted to a DomainApplication and the DomainApplication has been completed successfully this AUTH code will be set for the domain name.

Example:

- To create an EOI for an .APP Domain (only the domain name is given)

RRP Request/Response:

```
[COMMAND]
command=AddDomainApplication
class=EOI
domain=example.app
EOF
```

[RESPONSE]

```
CODE=200
DESCRIPTION=Command completed successfully
PROPERTY[APPLICATION][0]=3723
PROPERTY[AUTH][0]=TXU3InYoEv
PROPERTY[CLASS][0]=APP_EOI
PROPERTY[CREATEDBY][0]=SYSTEM
PROPERTY[CREATEDDATE][0]=2012-09-03 04:53:41
PROPERTY[DOMAIN][0]=example.app
PROPERTY[DOMAINUMLAUT][0]=example.app
PROPERTY[PEERUSER][0]=
PROPERTY[STATUS][0]=REQUESTED
PROPERTY[UPDATEBY][0]=SYSTEM
PROPERTY[UPDATEDDATE][0]=2012-09-03 04:53:41
PROPERTY[USER][0]=test.user
```

EPP:

EPP request:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key='COMMAND' value='AddDomainApplication' />
<keyvalue:kv key='CLASS' value='EOI' />
<keyvalue:kv key='DOMAIN' value='example2.app' />
</keyvalue:extension>
</extension>
</epp>
```

EPP response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<response>
<result code="1001">
<msg>Command completed successfully; action pending</msg>
<extValue>
<value xmlns:epp="urn:ietf:params:xml:ns:epp-1.0">
<epp:undef/>
</value>
<reason>200 Command completed successfully</reason>
</extValue>
</result>
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key="APPLICATION" value="3725"/>
<keyvalue:kv key="AUTH" value="ZBh5ralfPI"/>
<keyvalue:kv key="CLASS" value="APP_EOI"/>
<keyvalue:kv key="CREATEDBY" value="SYSTEM"/>
<keyvalue:kv key="CREATEDDATE" value="2012-09-03 09:47:10"/>
<keyvalue:kv key="DOMAIN" value="example2.app"/>
<keyvalue:kv key="DOMAINUMLAUT" value="example2.app"/>
<keyvalue:kv key="PEERUSER" value=""/>
<keyvalue:kv key="STATUS" value="REQUESTED"/>
<keyvalue:kv key="UPDATEBY" value="SYSTEM"/>
<keyvalue:kv key="UPDATEDDATE" value="2012-09-03 09:47:10"/>
<keyvalue:kv key="USER" value="test.user"/>
</keyvalue:extension>
</extension>
<trID>
<svTRID>RW-22720-1346665630348630</svTRID>
</trID>
</response>
</epp>
```

3.2 COMMAND – ModifyDomainApplication – Convert EOI to a DomainApplication

At any time the Authorization Code, contact handles and name servers that are assigned to an EOI can be updated free of charge. This command can also be used to convert an EOI to a DomainApplication if the respective newTLD is approved by ICANN.

Request Parameters:

Required:

COMMAND = ModifyDomainApplication
APPLICATION = <unique application ID>

Optional:

CLASS = <NEW_CLASS> e.g. APP_SUNRISE only to convert an EOI to a domain application if the respective newTLD is approved by ICANN
AUTH = AUTH code for the domain name (NOTE 1)
OWNERCONTACT0 = contact handle ID of registrant
ADMINCONTACT0 = contact handle ID of the administrative contact
TECHCONTACT0 = contact handle ID of the technical contact
BILLINGCONTACT0 = contact handle ID of the billing contact
NAMESERVER0 = nameserver #1
NAMESERVER1 = nameserver #2
...
NAMESERVER12 = nameserver #13

Note:

(1) If the EOI is converted to a DomainApplication and the DomainApplication has been completed successfully this AUTH code will be set for this domain

Response Parameters:

Parameters:

CLASS = application class (syntax: <TLD>_EOI)
APPLICATION = unique application ID
STATUS = application status (NOTE 1)
DOMAIN = domain name
USER = user id the DomainApplication is assigned to
PEERUSER = User ID hierarchy (tree) that DomainApplication is assigned to
CREATEDDATE = application created date
UPDATEDDATE = last application updated date
AUTH = AUTH code for the domain name (NOTE 2)

Note:

(1) Possible values are:

- INCOMPLETE - required parameters are missing, submission not possible
- REQUESTED - final status for EOIs, deposit not yet withdrawn for applications
- ACTIVE - deposit withdrawn successfully, waiting for submission to be prepared
- PROCESSING - submission prepared, waiting for submission
- PENDING - submitted, waiting for results
- SUCCESSFUL - DomainApplication successful
- FAILED - DomainApplication failed
- AUCTION-PENDING - multiple DomainApplications are being auctioned
- AUCTION-WON - auction was won
- AUCTION-LOST - auction was lost

(2) If the EOI is converted to a DomainApplication and the DomainApplication has been completed successfully this AUTH code will be set for the domain name.

Example:

- To convert an EOI to a Sunrise Application

RRP Request/Response:

```
[COMMAND]
command=ModifyDomainApplication
application=3723
class=APP_SUNRISE
EOF
```

[RESPONSE]

```
CODE=200
DESCRIPTION=Command completed successfully
PROPERTY[APPLICATION][0]=3723
PROPERTY[AUTH][0]=TXU3InYoEv
PROPERTY[CLASS][0]=APP_SUNRISE
PROPERTY[CREATEDBY][0]=SYSTEM
PROPERTY[CREATEDDATE][0]=2012-09-03 04:53:41
PROPERTY[DOMAIN][0]=example.app
PROPERTY[DOMAINUMLAUT][0]=example.app
PROPERTY[PEERUSER][0]=
PROPERTY[STATUS][0]=REQUESTED
PROPERTY[UPDATEBY][0]=SYSTEM
PROPERTY[UPDATEDDATE][0]=2012-09-12 07:29:15
PROPERTY[USER][0]=test.user
```

EPP:

EPP request:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key='COMMAND' value='ModifyDomainApplication' />
<keyvalue:kv key='APPLICATION' value='3725' />
<keyvalue:kv key='CLASS' value='APP_SUNRISE' />
</keyvalue:extension>
</extension>
/epp>
```

EPP response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<response>
<result code="1001">
<msg>Command completed successfully; action pending</msg>
<extValue>
<value xmlns:epp="urn:ietf:params:xml:ns:epp-1.0">
<epp:undef/>
</value>
<reason>200 Command completed successfully</reason>
</extValue>
</result>
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key="APPLICATION" value="3725"/>
<keyvalue:kv key="AUTH" value="ZBh5ralfPI"/>
<keyvalue:kv key="CLASS" value="APP_SUNRISE"/>
<keyvalue:kv key="CREATEDBY" value="SYSTEM"/>
<keyvalue:kv key="CREATEDDATE" value="2012-09-03 09:47:10"/>
<keyvalue:kv key="DOMAIN" value="example2.app"/>
<keyvalue:kv key="DOMAINUMLAUT" value="example2.app"/>
<keyvalue:kv key="PEERUSER" value=""/>
<keyvalue:kv key="STATUS" value="REQUESTED"/>
<keyvalue:kv key="UPDATEBY" value="SYSTEM"/>
<keyvalue:kv key="UPDATEDDATE" value="2012-09-12 06:17:50"/>
<keyvalue:kv key="USER" value="test.user"/>
</keyvalue:extension>
</extension>
<trID>
<svTRID>RW-22720-1346665630348630</svTRID>
</trID>
</response>
</epp>
```

3.3 COMMAND - Delete an EOI/DomainApplication

Request Parameters:

Required:

COMMAND = DeleteDomainApplication
CLASS = EOI
APPLICATION = <unique application ID>

Response Parameters:

Parameters:

CLASS = application class (syntax: <TLD>_EOI)
APPLICATION = unique application ID
STATUS = application status (NOTE 1)
DOMAIN = domain name
USER = user id the DomainApplication is assigned to
PEERUSER = User ID hierarchy (tree) that DomainApplication is assigned to
CREATEDDATE = application created date
UPDATEDDATE = last application updated date
AUTH = AUTH code for the domain name (NOTE 2)

Note:

(1) Possible values are:

- INCOMPLETE - required parameters are missing, submission not possible
- REQUESTED - final status for EOIs, deposit not yet withdrawn for applications
- ACTIVE - deposit withdrawn successfully, waiting for submission to be prepared
- PROCESSING - submission prepared, waiting for submission
- PENDING - submitted, waiting for results
- SUCCESSFUL - DomainApplication successful
- FAILED - DomainApplication failed
- AUCTION-PENDING - multiple DomainApplications are being auctioned
- AUCTION-WON - auction was won
- AUCTION-LOST - auction was lost

(2) If the EOI is converted to a DomainApplication and the DomainApplication has been completed successfully this AUTH code will be set for the domain name.

Example:

- Delete DomainApplication with ID 3723 (domain example.app)

RRP Request/Response:

```
[COMMAND]
command=DeleteDomainApplication
application=3723
EOF
```

[RESPONSE]

```
PROPERTY[DOMAIN][0]=example.app
DESCRIPTION=Command completed successfully
CODE=200
```

EPP:

EPP request:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key='COMMAND' value='DeleteDomainApplication' />
<keyvalue:kv key='APPLICATION' value='3723' />
</keyvalue:extension>
</extension>
</epp>
```

EPP response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<response>
<result code="1000">
<msg>Command completed successfully</msg>
<extValue>
<value xmlns:epp="urn:ietf:params:xml:ns:epp-1.0">
<epp:undef/>
</value>
<reason>200 Command completed successfully</reason>
</extValue>
</result>
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key="DOMAIN" value="example921.app"/>
</keyvalue:extension>
</extension>
<trID>
<svTRID>RW-22720-1346669564857839</svTRID>
</trID>
</response>
</epp>
```

3.4 COMMAND - Query Details about a single EOI / DomainApplication

Request Parameters:

Required:

COMMAND = StatusDomainApplication
APPLICATION = <unique application ID>

Response Parameters:

Parameters:

CLASS = application class (syntax: <TLD>_EOI)
APPLICATION = unique application ID
STATUS = application status (**NOTE 1**)
DOMAIN = domain name
USER = user id the DomainApplication is assigned to
PEERUSER = User ID hierarchy (tree) that DomainApplication is assigned
CREATEDDATE = application created date
UPDATEDDATE = last application updated date
AUTH = AUTH code for the domain name (**NOTE 2**)

Note:

(1) Possible values are:

- INCOMPLETE - required parameters are missing, submission not possible
- REQUESTED - final status for EOIs, deposit not yet withdrawn for applications
- ACTIVE - deposit withdrawn successfully, waiting for submission to be prepared
- PROCESSING - submission prepared, waiting for submission
- PENDING - submitted, waiting for results
- SUCCESSFUL - DomainApplication successful
- FAILED - DomainApplication failed
- AUCTION-PENDING - multiple DomainApplications are being auctioned
- AUCTION-WON - auction was won
- AUCTION-LOST - auction was lost

(2) If the EOI is converted to a DomainApplication and the DomainApplication has been completed successfully this AUTH code will be set for the domain name.

Example:

- To query details about a single EOI / DomainApplication

RRP Request/Response:

```
[COMMAND]
command=StatusDomainApplication
application=3723
EOF
```

[RESPONSE]

```
CODE=200
DESCRIPTION=Command completed successfully
PROPERTY[APPLICATION][0]=3725
PROPERTY[AUTH][0]=TXU3InYoEv
PROPERTY[CLASS][0]=APP_EOI
PROPERTY[CREATEDBY][0]=SYSTEM
PROPERTY[CREATEDDATE][0]=2012-09-03 04:53:41
PROPERTY[DOMAIN][0]=example.app
PROPERTY[DOMAINUMLAUT][0]=example.app
PROPERTY[PEERUSER][0]=
PROPERTY[STATUS][0]=REQUESTED
PROPERTY[SUBCLASS][0]=APP_EOI
PROPERTY[UPDATEBY][0]=SYSTEM
PROPERTY[UPDATEDDATE][0]=2012-09-03 04:53:41
PROPERTY[USER][0]=test.user
```

EPP:

EPP request:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key='COMMAND' value='StatusDomainApplication' />
<keyvalue:kv key='APPLICATION' value='3725' />
</keyvalue:extension>
</extension>
</epp>
```

EPP response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<epp xmlns="urn:ietf:params:xml:ns:epp-1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="urn:ietf:params:xml:ns:epp-1.0 epp-1.0.xsd">
<response>
<result code="1001">
<msg>Command completed successfully; action pending</msg>
<extValue>
<value xmlns:epp="urn:ietf:params:xml:ns:epp-1.0">
<epp:undef/>
</value>
<reason>200 Command completed successfully</reason>
</extValue>
</result>
<extension>
<keyvalue:extension xmlns:keyvalue="http://schema.ispapi.net/epp/xml/keyvalue-1.0"
xsi:schemaLocation="http://schema.ispapi.net/epp/xml/keyvalue-1.0 keyvalue-1.0.xsd">
<keyvalue:kv key="APPLICATION" value="3725"/>
<keyvalue:kv key="AUTH" value="ZBh5ralfPI"/>
<keyvalue:kv key="CLASS" value="APP_EOI"/>
<keyvalue:kv key="CREATEDBY" value="SYSTEM"/>
<keyvalue:kv key="CREATEDDATE" value="2012-09-03 09:47:10"/>
<keyvalue:kv key="DOMAIN" value="example2.app"/>
<keyvalue:kv key="DOMAINUMLAUT" value="example2.app"/>
<keyvalue:kv key="PEERUSER" value=""/>
<keyvalue:kv key="STATUS" value="REQUESTED"/>
<keyvalue:kv key="SUBCLASS" value="APP_EOI"/>
<keyvalue:kv key="UPDATEBY" value="SYSTEM"/>
<keyvalue:kv key="UPDATEDDATE" value="2012-09-03 09:47:10"/>
<keyvalue:kv key="USER" value="test.user"/>
</keyvalue:extension>
</extension>
<trID>
<svTRID>RW-22720-1347544683889437</svTRID>
</trID>
</response>
</epp>
```

3.5 COMMAND - Query List All EOIs / DomainApplications

Request Parameters:

Required:

COMMAND = QueryDomainApplicationList
CLASS = EOI

Optional:

WIDE = 1
STATUS = filter the list by STATUS value (**NOTE 1**)
ORDERBY = order by parameter, suffix "DESC" in descending order (**NOTE 2**)
USERDEPTH = filter the list by choosing user depth (**NOTE 3**)

Note:

(1) Possible values are:

- INCOMPLETE - required parameters are missing, submission not possible
- REQUESTED - final status for EOIs, deposit not yet withdrawn for applications
- ACTIVE - deposit withdrawn successfully, waiting for submission to be prepared
- PROCESSING - submission prepared, waiting for submission
- PENDING - submitted, waiting for results
- SUCCESSFUL - DomainApplication successful
- FAILED - DomainApplication failed
- AUCTION-PENDING - multiple DomainApplications are being auctioned
- AUCTION-WON - auction was won
- AUCTION-LOST - auction was lost

(2) Possible values are:

- CLASS
- CLASSDESC
- DOMAIN
- DOMAINDESC
- USER
- USERDESC
- CREATEDDATE
- CREATEDDATEDESC
- UPDATEDDATE
- UPDATEDDATEDESC

(3) Possible values are:

- SELF (only own EOIs/Applications)
- SUBUSER (only subusers EOIs/Applications)
- ALL (All EOIs/Applications)

Response Parameters:

Parameters:

Common List of Parameters

Returned as an array with only one element

- TOTAL = number of total list entries
- FIRST = number of first returned list entry
- COUNT = number of returned entries
- LAST = number of last returned list entry
- LIMIT = limitation of returned list entries

List entries returned as an array:

- CLASS = application class (syntax: <TLD>_EOI)
- APPLICATION = unique application ID
- STATUS = application status (**NOTE 1**)
- DOMAIN = domain name
- USER = user id the DomainApplication is assigned to
- PEERUSER = User ID hierarchy (tree) that DomainApplication is assigned
- CREATEDDATE = application created date
- UPDATEDDATE = last application updated date
- AUTH = AUTH code for the domain name (**NOTE 2**)
- OWNERCONTACT = contact handle ID of registrant
- ADMINCONTACT = contact handle ID of administrative contact
- TECHCONTACT = contact handle ID of technical contact
- BILLINGCONTACT = contact handle ID of billing contact
- EXPIRATIONDATE = TBD
- X-DOMAIN-ROID = TBD
- MISSINGPARAMETER = TBD
- INVALIDPARAMETER = TBD

Note:

(1) Possible values are:

- INCOMPLETE - required parameters are missing, submission not possible
- REQUESTED - final status for EOIs, deposit not yet withdrawn for applications
- ACTIVE - deposit withdrawn successfully, waiting for submission to be prepared
- PROCESSING - submission prepared, waiting for submission
- PENDING - submitted, waiting for results
- SUCCESSFUL - DomainApplication successful
- FAILED - DomainApplication failed
- AUCTION-PENDING - multiple DomainApplications are being auctioned
- AUCTION-WON - auction was won
- AUCTION-LOST - auction was lost

(2) If the EOI is converted to a DomainApplication and the DomainApplication has been completed successfully this AUTH code will be set for the domain name.

Example:

- To query a detailed list of all EOIs / DomainApplications

RRP Request/Response:

```
[COMMAND]
command=QueryDomainApplicationList
class=EOI
wide=1
EOF
```

[RESPONSE]

```
CODE=200
DESCRIPTION=Command completed successfully
PROPERTY[ADMINCONTACT][0]=
PROPERTY[APPLICATION][0]=3725
PROPERTY[BILLINGCONTACT][0]=
PROPERTY[CLASS][0]=APP_EOI
PROPERTY[COUNT][0]=1
PROPERTY[CREATEDDATE][0]=2012-09-03 09:47:10
PROPERTY[DOMAIN][0]=example2.app
PROPERTY[DOMAINUMLAUT][0]=example2.app
PROPERTY[EXPIRATIONDATE][0]=
PROPERTY[FIRST][0]=0
PROPERTY[INVALIDPARAMETER][0]=
PROPERTY[LAST][0]=0
PROPERTY[LIMIT][0]=10000
PROPERTY[MISSINGPARAMETER][0]=
PROPERTY[NAMESERVER][0]=
PROPERTY[OWNERCONTACT][0]=
PROPERTY[PARENTUSER][0]=demo.hexonet.net
PROPERTY[PEERUSER][0]=
PROPERTY[STATUS][0]=REQUESTED
PROPERTY[TECHCONTACT][0]=
PROPERTY[TOTAL][0]=1
PROPERTY[UPDATEDDATE][0]=2012-09-03 09:47:10
PROPERTY[USER][0]=test.user
PROPERTY[X-DOMAIN-ROID][0]=
```