Canadian Internet
Registration Authority

Autorité canadienne pour
les enregistrements Internet

TECHNICAL SPECIFICATIONS FOR REGISTRARS

USER GUIDE

TECHNICAL SPECIFICATIONS FOR REGISTRARS

USER GUIDE

Publication: RPPG 00012
Document Version: 1.4
Document Date: November 1, 2002
Publication Date: December 3, 2002
Publication Time: 09:00 Ottawa Time

PUBLICATION HISTORY

JANUARY 08, 2002

Version 1.0 This document replaces Registration: Technical Rules and Specifications for Registrars. Although content remains the same, format has changed for ease of reading and comprehension.

JANUARY 25, 2002

Version 1.1 The following new error codes were added to Annex C: 01301 to 01342 and 05019 to 05022.

JUNE 13, 2002

Version 1.2 The following changes have been made to this document for this release:

Chapter 2: Addition of a new section describing CIRA System Maintenance;

Annex C: Addition of the following new error codes: 01236 to 01240; 01345; 05023.

SEPTEMBER 15, 2002

Version 1.3 The following changes were made to this document for this release:

Chapter 4 Whois Utility: Update to reflect the new whois architecture;

Chapter 6: Addition of new section .ca Name Server Validation. This utility enables registrars to query the CIRA database and get information on the validity of a .ca namer server;

Chapter 7: Addition of new section Cancel Applicant Registrant. The process of Applicant Registrant can be cancelled by the Registrar so long as the admin contact has not yet approved the registration agreement;

Annex A: Update to reflect the change in telephone number verification by the CIRA system;

Annex D: Addition of new Annex for the Process Name Abbreviations;

Annex E: Addition of new Annex for the Process Status Abbreviations.

DECEMBER 3, 2002

Version 1.4 The following changes were made to this document for this release:

Chapter 2: Addition of new section Token XML Interface

Chapter 4 Registration Procedures: Whois Utility, Domain Name Availability Check and Domain Name List updated to reflect the changes in Chapter 2.

Chapter 6 Getting Information: Getting Domain Name Registration Information updated to reflect the changes in Chapter 2.

Chapter 10 Registrar Reports: Addition of new section Registrar Deposit Account Statement

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION

CIRA is the not-for-profit organization mandated by the Government of Canada to operate the dot-ca top-level domain for all Canadians. It is responsible for setting policy and managing the dot-ca domain database, as well as registering domain names through its network of certified Registrars. For more information on CIRA, please visit www.cira.ca.

The following topics are discussed in this section:

- "Who should read this document and why"
- "What you need to know"
- "What's new in this document"
- "Related documents"
- "How to get more help"

WHO SHOULD READ THIS DOCUMENT AND WHY

This document is intended for CIRA Certified Registrars. The purpose of this document is to give Registrars the information and specifications required to build a system that can effectively communicate with CIRA's system.

WHAT YOU NEED TO KNOW

CIRA's system supports 3 interfaces. Most functions are available through these interfaces and they are as follows:

- Standard web interface
- Socket-based web interface
- Email

SECURITY

PGP (Pretty Good Privacy) software is used to encrypt and sign email based transaction requests between CIRA and CIRA Certified Registrars.

PGP is a powerful cryptographic tool that enables people to securely exchange messages. By providing the ability to encrypt messages, PGP provides protection against information being intercepted and allows for verification of the identity of the sender.

Anyone wishing to become a CIRA Certified Registrar must obtain their own private PGP key and forward it to CIRA with their completed application. For more information on PGP software go to: http://www.heureka.clara.net/sunrise/pgpwhat.htm

WHAT'S NEW IN THIS DOCUMENT

The following items have been added for this release:

- A new section on CIRA SYSTEM MAINTENACE has been added to Chapter 2, Technical Overview, page 2-4.
- The following new error codes have been added to Annex C:

  01236 to 01240, 01345 and 05023.

RELATED DOCUMENTS

Related documents for Registrars and Registrants can be found at: http://www.cira.ca/en/officials.html

or

http://www.cira.ca/fr/officials.html

HOW TO GET MORE HELP

Additional information can be obtained by calling the CIRA Customer Support Unit at 1-877-860-1411 from Monday to Friday 8:00 to 20:00 Ottawa time or by sending an email to regsupport@cira.ca

CHAPTER 2 TECHNICAL OVERVIEW

The purpose of this chapter is to provide CIRA Certified Registrars with technical information about CIRA's system. This will include the requirements for building a system to communicate with the CIRA system and how to access the CIRA system.

ACCESSING THE CIRA SYSTEM

CIRA's system supports three interfaces a standard web interface, socket-based web interface, and email. Most functions are available through these 3 interfaces. Registrars must use the user code and password provided by CIRA to access CIRA's system for all three interfaces. An additional user code and password can be obtained if required. To have a user code and password changed or to obtain a new user code and password the registrar must contact regadmin@cira.ca or call the helpline at: 1-877-860-1411.

SECURITY PROCEDURES

All email based transaction requests from a CIRA Certified Registrar must be signed with the Registrar's private PGP key and encrypted with CIRA's public PGP key and then forwarded to CIRA via standard Internet email to "transac@cira.ca".

PARAMETERS

A parameter is a value or reference passed to a function, procedure, subroutine, command or program by the caller. Parameter values used by the CIRA system are:

NUMBER	numeric value
VARCHAR2	variable length character string
DATE	date format YYYY/MM/DD (i.e.2000/08/10)

The following is an example of parameter ctc_nationality:

```
ctc_nationality        VARCHAR2(3)        OPTIONAL
  Nationality of administrative contact

Valid values:
 CND Canadian
 OTH Foreign citizenship
 RES Canadian resident
```

The first column represents the parameter name, the second column the parameter type and length, and the third column indicates whether the parameter is mandatory. If the value is longer than the specified length, the entry will be rejected.

The line below the parameter name gives a short description followed by any valid values. When a list of valid values is specified it must be respected.

STANDARD WEB INTERFACE

Registrars using the standard Web interface can access the registrar web site from the following: www.cira.ca/cira/registraires The registrar must then enter the user code and password provided by CIRA. Almost all transactions will be conducted from this registrar web site.

CIRA will also provide a URL, user code, and password to Registrants. The URL for registrants is the following: www.cira.ca/cira/detenteurs. If a Registrant should lose or forget the CIRA supplied user code and password, the Registrant's Registrar can request the CIRA system to send the Registrant an email with the original user code and password. However, to change passwords or request a new password, the Registrar, on behalf of the Registrant must contact regadmin@cira.ca .

SOCKET-BASED XML INTERFACE

Registrars using the socket-based web interface can submit their socket-based requests for all transactions to:

https: //cira2.cira.ca/cira/registraires

https://cira2.cira.ca/cira/rwhois

http: //cira2.cira.ca/cira/rwhois.token

http: //cira2.cira.ca/cira/rinfo

XML=Y is a mandatory parameter in all of these interfaces. If this parameter is present, the interface will display a plain text layout of the data in XML format in the latin1 character set. For eg.

- If XML='Y' has been specified in the request:

```
    <result>
<status>status code</status>
     <error>
        <code>(return code number)</code>
        <description></description>
     </error>
     <parameter-name>(parameter-value)</parameter-name>
     ...
     <parameter-name>(parameter-value)</parameter-name>
  </result>
```

To obtain XML response other than the plain text latin1 character set, traditional response should include the parameter pure = y with their request to cira2 XML socket. This will return "pure" XML format data in the UTF-8 character set. This would apply to all functions on registraires, rwhois, rwhois_token and rinfo XML interfaces.

For eg.

- If XML='Y'&pure=y has been specified in the request:

```
  <?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<result>
  <status>(status code)</status>
<error>
  <code>(error code)</code>
  <description>(error description)</description>
 </error>
  <parameter-name>(parameter value)</parameter-
name>
   <parameter-name>(parameter value)</parameter-
name>
  </result>
```

TOKEN XML INTERFACE

The token XML interface applies only to a set of Read-Only transactions such as:
WHOIS (on http://cira2.cira.ca/cira/rwhois.token), and
GET_DOM_INFO, VALID_DOM and LIST_DOM
(http://cira2.cira.ca/cira/rinfo)

To use a token interface you must first set the token, then call the same function by specifying token= parameter instead of user/pwd.

Rules:
1. A token expires 24 hours from the moment it is set.
2. It can only be used from the ip that set it.
3. A registrar can set no more than 10 live tokens at a time.

Using the token interface involves two steps:
1) Set token

Tokens can be set either at:

https://cira2.cira.ca/cira/rwhois.token (WHOIS token interface)
https://cira2.cira.ca/cira/rinfo (the other token interfaces)

With:

```
Request parameters:
set_token                            VARCHAR2(1)
MANDATORY
  Valid value
    Y
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
```

For eg. by using this sample call:
```
https://cira2.cira.ca/cira/rwhois.token?XML=Y&user=???&pwd
=??&set_token=y&transac_type=WHOIS
```

You will set one token:

Reply:
```
<result>
<status>ok</status>
<token>3150289.6220443.venus.64.230.122.119.22545956.1525.
2320780</token>
```

```
</result>
```

2) Use token

To use token just set call :

http://cira2.cira.ca/cira/rwhois.token (WHOIS token interface)
or
http://cira2.cira.ca/cira/rinfo (the other token interfaces)

```
Request parameters:
token                          VARCHAR2(60)
MANDATORY
  Registrar token
XML                            VARCHAR2(10)
MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
transac_type                   VARCHAR2(10)
MANDATORY
 Transaction type
<transac_type call spec>
```

For example: Whois call over token just set will be:

http://cira2.cira.ca/cira/rwhois.token?XML=Y&token=3150295.31780098.venus.64.230.122.119.22545962.1526.2320726&transac_type=WHOIS&subdomain=cira.ca

REAL-TIME COMPLETION OF REQUESTS FOR THE SOCKET-BASED WEB INTERFACE

Registrars who work with their Registrants via web sites and CIRA in real time may opt to complete the registration process in real time. If a Registrar decides to complete the registration process in real time, the email sent to the Registrant will acknowledge the registration completion instead of inviting the Registrant to complete it.

Registrars choosing this option must use the following procedure after receiving confirmation of a successful registration request from the CIRA system.

The response will include the following parameters:

```
<user_acc>(user-name)</user_acc>
<password>(password)</password>
```

These are the CIRA supplied Registrant user code and password that the Registrant will need to accept the CIRA Registrant Agreement on-line.

Registrars can use these codes to establish transactional control over the Registrant's acceptance of the CIRA Registrant Agreement.

Having given these values to the Registrant, the Registrar's system should re-direct the Registrant to https://cira2.cira.ca/cira/detenteurs and supply the following parameters:

URL: Contains the URL that the Registrar wishes CIRA to transfer the Registrant to after the Registrant has accepted the CIRA Registrant Agreement;

TID: Registrar generated transaction identifier that will be returned when transferring to the Registrar's URL (free format string);

TYPE=ENREG_RANT: confirms the transaction type.

Once the Registrant has accepted or rejected the CIRA Registrant Agreement, the Registrant's browser will be re-directed to the URL specified in the Registrar's request and the CIRA system will supply a result parameter RES = A (Registrant accepted) or R (Registrant refused) as well as the TID parameter.

The Registrar will now be able to complete its transaction with the Registrant.

CLASSIC EMAIL

All email based transaction requests from a CIRA Certified Registrar must be signed with the Registrar's private PGP key and encrypted with CIRA's public PGP key and then forwarded to CIRA via standard Internet email to "transac@cira.ca".

> ***Note:*** The Registrar’s PGP key is associated with the email address used to exchange mail with CIRA.

Messages must be formatted as follows:

```
(parameter-name)='(parameter-value)'
```

The parameter name must be followed by an equal sign and the parameter value. It must be enclosed by single quotes ('). To specify a single quote within the value, a backslash will precede the quote. For example a message containing the domain name Domains’ R us would be formatted as follows:

```
rgpc_description='Domains\'R us'
```

Parameters can be presented in random order. All required parameters must be present in a transaction request. The body of the request email will be in the following format:

```
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The “FROM:” field of these email transaction requests will be used to transmit the processing results to the sender. The response email is also encrypted as per policy. The body of the message will contain the return code(s) followed by a copy of the original request. The body of the response email will be in the following format:

- If XML='Y' has been specified in the request:

```
<result>
   <error>
      <code>(return code number)</code>
      <description></description>
   </error>
   <parameter-name>(parameter-value)</parameter-name>
   ...
   <parameter-name>(parameter-value)</parameter-name>
</result>
```

- If XML option has not been specified in the request or XML=N:

```
status:  (status code)
errors:
(return code number): (return code text, optional)
end of error
(parameter-name)= '(parameter-value)'
...
(parameter-name)= '(parameter-value)'
```

CIRA SYSTEM MAINTENANCE

Regular maintenance on the CIRA system is performed on a weekly basis and takes approximately one hour. This maintenance is currently scheduled every Sunday from 03:30 to 04:30, Ottawa time. During this time the CIRA system is unavailable to Registrars and the general public.

In the event the CIRA system requires more than the scheduled maintenance described above, Registrars will be given a 48 hour notice via email. Whenever possible, CIRA staff will schedule maintenance outside of normal business hours that is: between 07:00 and 21:00, Ottawa time.

CHAPTER 3 REGISTRATION OVERVIEW

All registrations must be processed through a CIRA Certified Registrar. In order to register a dot-ca domain name a registrant must first meet CIRA's Canadian Presence Requirements (http://www.cira.ca/en/cat_Registration.html).

REGISTRANT AGREEMENT APPROVAL PROCESS

Every new registrant must approve CIRA's Registrant Agreement. When a REGIS_DOM_RANT or ENREG_RANT transaction is accepted, an email is sent to the registrant with a user code and password. Domain names accepted for the registrant will be reserved but inactive until the registrant consents to CIRA's Registrant Agreement (https://www.cira.ca/cira/detenteurs). If the registration is not completed within the 7-day period the registrant will be removed from the system and all domain names reserved for this Registrant will become available. The Registrar can request a 7-day extension period for the registration process by submitting a DELAY_DOMAIN_APPROVAL transaction. Only one extension will be granted per registration process.

If a Registrar decides to complete the registration process in real time, the email sent to the Registrant will acknowledge the registration completion instead of inviting the Registrant to complete it.

EXISTING REGISTRANT, NEW DOMAIN

The REGIS_DOM transaction is used to register a domain name for an existing Registrant. You need the Registrant's identification number to submit a REGIS_DOM transaction. If the Registrant has already approved CIRA's Registrant Agreement and the domain name requested is available, the system will register and activate the requested domain name without further action.

CONFIRMATION EMAIL

For each transaction request submitted, an email containing a request's status will be sent to the Registrant.

However, no email is sent if the request is incomplete (status = 'INC'). Incomplete status results from mandatory parameters missing or parameters with invalid values in the request, or the Registrar's account balance is insufficient to cover the transaction fees.

REGISTRATION REQUEST VERIFICATION

If a Registrar wants only to check whether a registration request will be accepted or rejected, it can ask the system to verify the request. By adding the prefix VERIF_ to the normal value of the parameter transac_type, the system will only check the request without generating a transaction. This feature is available for the following transaction: ENREG_RANT, REGIS_DOM, and REGIS_DOM_RANT.

REAL-TIME COMPLETION OF REGISTRATION REQUESTS

Registrars who work with their Registrants via websites and CIRA may opt to complete the following functions in real time: ENREG_RANT and REGIS_DOM_RANT.

REGISTRANT ACCESS TO CIRA WEB SITE

When a request to register a new Registrant is entered into the system, a user account is created and an email from CIRA containing a user code and password is sent to the administrative contact. At the present time, only administrative contacts can have a user account.

The user code and password sent to the administrative contact at the time of registration will be needed by the Registrant to access restricted sections of CIRA's web site. Access to restricted sections of CIRA's web site is necessary to acknowledge information changes, approve transfers, or vote at CIRA elections.

SENDING USER CODE AND PASSWORD

This procedure sends an email message containing a user code and password to the Registrant's administrative contact. The result of the request is returned in the STATUS parameter. It can have one of the following values:

```
OK:  Success, an email message is sent.
ERR: Failure, error codes are returned.
```

Request parameters:

```
user                NUMBER(16)      MANDATORY
  Registrar user id

pwd                 VARCHAR2(30)    MANDATORY
  Registrar password

XML                 VARCHAR2(10)    MANDATORY
  Valid value
    Y       Response in XML format. Mandatory for
socket-based web interface
    N       Other interface

transac_type        VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    EMAIL_PWD

rgpc_no              NUMBER(16)      MANDATORY
 Registrant unique identifier.

ctc_id               NUMBER(16)      OPTIONAL
  Contact unique identifier.  If missing, the system
will use the administrative contact identifier.
```

CHAPTER 4 REGISTRATION PROCEDURES

This chapter will describe in detail the following registration procedures:

WHOIS UTILITY

The WHOIS utility enables registrars to query the CIRA database and get information on the availability of domain names. The status of the domain name will be returned in the 'status' parameter and will have one of the following values:

EXIST	Domain name exists and information on the domain is returned.
APPRVL	Domain name is awaiting approval by the registrant.
SUSPD	Domain name exists, but is suspended.
AVAIL	Domain name is available for registration.
UNAV	Domain name is not available.
TBR	Domain name is being withheld by the system.

If a domain name is awaiting approval by the Registrant, DNS and contact information are not returned. If the domain name is suspended, DNS information is not returned. If a domain name is not available, an error code explaining the reason is returned.

Socket-based Web interface details

Registrars can submit their socket-based WHOIS request to:
https://cira2.cira.ca/cira/rwhois
http://cira2.cira.ca/cira/rwhois.token

Request parameters:

For https://cira2.cira.ca/cira/rwhois

```
user NUMBER(16)  MANDATORY
Registrar user id

pwd VARCHAR2(30)           MANDATORY
Registrar password

XML VARCHAR2(10)        MANDATORY
 Valid value:
  Y Response in XML format.                MANDATORY


transac_type  VARCHAR2(5)          MANDATORY
  Transaction type
  Valid values: WHOIS

subdomain VARCHAR2(50)        MANDATORY
   Domain name.
```

If the SUBDOMAIN input value ends with a .ca the output will display whois information for that particular domain. If the input string does not end in .ca, a wildcard search on domains starting with the input string will be performed.

WHOIS using tokens:

To use a token interface you must first set the token, then call the same function by specifying token=parameter instead of user/pwd.

Rules:
1. A token expires 24 hours from the moment it is set.

2. It can only be used from the ip that set it.
3. A registrar can set no more than 10 live tokens at a time.

Using the token interface involves two steps:
1) Set token

Tokens can be set at https://cira2.cira.ca/cira/rwhois.token

With:

```
Request parameters:
set_token                       VARCHAR2(1)     MANDATORY
  Valid value
    Y
user                            NUMBER(16)       MANDATORY
  Registrar user id

pwd                             VARCHAR2(30)     MANDATORY
  Registrar password

XML                             VARCHAR2(10)     MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
transac_type                    VARCHAR2(10)     MANDATORY
 Transaction type
```

For eg. by using this sample call:
https://cira2.cira.ca/cira/rwhois.token?XML=Y&user=???&pwd=??&set_token=y&transac_type=WHOIS

You will set one token:

Reply:

```
<result>
<status>ok</status>
<token>3150289.6220443.venus.64.230.122.119.22545956.1525.
2320780</token>
</result>
```

2) Use token

To use token set call to http://cira2.cira.ca/cira/rwhois.token

```
Request parameters:
token       VARCHAR2(60)        MANDATORY
  Registrar token
XML         VARCHAR2(10)       MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
transac_type  VARCHAR2(5)       MANDATORY
  Transaction type
  Valid values: WHOIS

subdomain VARCHAR2(50)          MANDATORY
   Domain name.
```

For example: Whois call over token just set will be:

http://cira2.cira.ca/cira/rwhois.token?XML=Y&token=3150295.31780098.venus.64.230.122.119.22545962.1526.2320726&transac_type=WHOIS&subdomain=cira.ca

XML format details

A WHOIS request will generate the following return codes via XML:

```
 <result>
   <status>(status value)</status>
  <output_parameter_name>(VALUE)</output_parameter_name>
     ......
 <output_parameter_name>(VALUE)</output_parameter_name>
 </result>
```

If the domain name is not available, the response will be in the following format:

```
    <result>
       <status>(status value)</status>
       <error>
          <code>( ERROR CODE)</code>
          <description>( ERROR DESCRIPTION)</description>
       </error>
    <subdomain>(VALUE)</subdomain>
       ......
    </result>
```

If a wildcard search is performed:

```
    <result>
        <rec #> (subdomain)</rec>
         </result>
```

One of several error codes and descriptions will explain why the domain is not available. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for a successful request will be in the following format:

```
     status: (status code)
     (input parameter-name)='(input parameter-value)'
     ....
     (input parameter-name)='(input parameter-value)'
     (output parameter-name)='(output parameter-value)'
     ....
     (output parameter-name)='(output parameter-value)'
```

If the domain name is not available, the response will be in the following format:

```
status: RJC
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will explain why the domain is not available. In every case, all the parameters transmitted in the originating request will be returned.

Response additional parameters:

```
dom_no                   NUMBER(16)
  Domain unique identifier

rant_no                  NUMBER(16)
  Registrant unique identifier

rgar_no               NUMBER(16)
  Registrar unique identifier.

registrar             VARCHAR2(200)
  Registrar name.
```

```
date_received           DATE
  Registration request date.

date_approved            DATE
  Registration approval date.

date_modified            DATE
  Date of last modification of domain information.

organization             VARCHAR2(200)
  Registrant name.

description               VARCHAR2(2000)
  Registrant description.

ctc_id_adm                NUMBER(16)
  Administrative contact unique identifier.

admin_name                 VARCHAR2(300)
  Administrative contact name.

admin_title                 VARCHAR2(200)
  Administrative contact job title.

admin_postal                VARCHAR2(500)
  Administrative postal address.

admin_phone                 VARCHAR2(40)
  Administrative contact telephone number.

admin_fax                   VARCHAR2(40)
  Administrative contact fax number.

admin_mailbox                VARCHAR2(100)
  Administrative email address.

ctc_id_tec                    NUMBER(16)
  Technical contact unique identifier.

tech_name                     VARCHAR2(300)
  Technical contact name.

tech_title                    VARCHAR2(200)
  Technical contact job title.
```

```
tech_postal                    VARCHAR2(500)
  Technical contact postal address.

tech_phone                     VARCHAR2(40)
  Technical contact telephone number.

tech_fax                       VARCHAR2(40)
  Technical contact fax number.

tech_mailbox                   VARCHAR2(100)
  Technical contact email address.

ns1_hostname                   VARCHAR2(200)
  Name of the first name server

ns1_netaddress                 VARCHAR2(100)
  IP address of the first name server

ns2_hostname                   VARCHAR2(200)
  Name of second name server

ns2_netaddress                 VARCHAR2(100)
  IP address of second name server

ns3_hostname                   VARCHAR2(200)
  Name of third name server

ns3_netaddress                 VARCHAR2(100)
  IP address of third name server

ns4_hostname                   VARCHAR2(200)
  Name of fourth name server

ns4_netaddress                 VARCHAR2(100)
  IP address of fourth name server

ns5_hostname                   VARCHAR2(200)
  Name of fifth name server

ns5_netaddress                 VARCHAR2(100)
  IP address of fifth name server

ns6_hostname                   VARCHAR2(200)
  Name of sixth name server
```

```
ns6_netaddress                  VARCHAR2(100)
  IP address of sixth name server
```

NEW REGISTRANT WITHOUT A DOMAIN NAME

This procedure is used to register a new Registrant without a domain name. The new Registrant will be issued a Registrant identification number. Once this identification number has been issued the Registrar can then use the REGIS_DOM transaction to register domain names for the Registrant. The Registrant is given a period of 90 days to register a domain name.

> ***Note:*** If no domain name registration takes place within the 90-day period, the Registrant will be removed from the CIRA system.

At least one phone number must be provided for the administrative contact. Parameter ctc_phone will be published by the WHOIS, while ctc_others will not be published.

If a Registrar tries to register a Registrant that is already registered, the system will reject the request. The result of a registration request is returned in the STATUS parameter. Possible values are:

```
AAP: Registration request processed and accepted.
INC: Registration request is incomplete and was not
     processed.(Input parameters missing or with wrong
     values).
RJC: Registration request processed and rejected.
```

In the case of an accepted (AAP) or rejected (RJC) request, a confirmation email is sent to the Registrant and to the Registrar. If the request is incomplete, no email is sent.

A Registrar can check the status of a registration request by setting parameter transac_type to VERIF_ENREG_RANT. This function will tell the Registrar if the request will be accepted or rejected without generating a transaction.

> ***Note:*** Response parameters 'pris_id', 'rgpc_no', 'user_acc' and 'password' will contain invalid values since no registration takes place.

Socket-based web interface details

Registrars can submit their socket-based registration request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A registration request will generate the following return codes in XML:

```
<result>
   <status>(STATUS)</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</ input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
   <output_parameter_name>(VALUE)</ output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</ output_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected or incomplete status. All the values transmitted in the originating request will follow (input parameters). If the request is accepted, some additional fields (output parameters) will be included in the response.

A successful transaction request will generate a status code with a value of 'AAP' in the following format:

```
<result>
   <status>AAP</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an accepted request will be in the following format:

```
status: AAP
     (input parameter-name)='(input parameter-value)'
     ....
     (input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status: RJC
errors:
     (return code number): (return code text, optional)
     end of error
     (input parameter-name)='(input parameter-value)'
     ....
     (input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Real-time completion of registration requests for socket-based web interface

Registrars working with their Registrants via web sites and CIRA in real time may opt to complete all registration requests in real time. To do so Registrars must use the following procedure after receiving a confirmation of a successful registration request from the CIRA system.

The returned parameters will include:

```
<user_acc>(user-name)</user_acc>
<password>(password)</password>
```

These are the CIRA supplied Registrant user code and password the Registrant will be required to use to accept the CIRA Registrant Agreement online. Registrars can use these codes to establish transactional control over the Registrant's approval of the CIRA Registrant Agreement. After passing on these values to the Registrant the Registrar's system should re-direct the Registrant to:

https://cira2.cira.ca/cira/detenteurs and supply the following parameters:

```
URL: Contains the URL that the Registrar wishes CIRA to
transfer to after the Registrant has completed its
approval of the CIRA Registrant Agreement;

TID: Registrar generated transaction identifier that will
be returned when transferring to the Registrar's URL (free
format string);

TYPE=ENREG_RANT: confirms the transaction type.
```

Once the Registrant has accepted or rejected the CIRA Registrant Agreement, the Registrant's browser will be re-directed to the URL specified in the Registrar's request and the CIRA system will supply a result parameter RES = A (Registrant accepted) or R (Registrant refused) as well as the TID parameter. The Registrar will now be able to complete its transaction with the Registrant. Once the transaction is completed the system will send a new password to the Registrant.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password
XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface

transac_id                     VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(10)    MANDATORY
  Transaction type
  Valid values:
    ENREG_RANT           Registration request transaction
    VERIF_ENREG_RANT     Registration request verification

rgpc_name                      VARCHAR2(200)   MANDATORY
  Name of the Registrant.

rgpc_legal_type                VARCHAR2(3)     MANDATORY
 Registrant legal type.

Valid values:

CCO   Corporation (Canada or Canadian province or
      territory)
CCT   Canadian citizen
RES   Permanent Resident of Canada
GOV   Government or government entity in Canada
EDU   Canadian Educational Institution
ASS   Canadian Unincorporated Association
HOP   Canadian Hospital
PRT   Partnership Registered in Canada
TDM   Trade-mark registered in Canada (by a non-Canadian
      owner)
TRD   Canadian Trade Union
PLT   Canadian Political Party
LAM   Canadian Library, Archive or Museum
TRS   Trust established in Canada
ABO   Aboriginal Peoples (individuals or groups)indigenous
      to Canada
INB   Indian Band recognized by the Indian Act of Canada
LGR   Legal Representative of a Canadian Citizen or
      Permanent Resident
OMK   Official mark registered in Canada
MAJ   Her Majesty the Queen

rgpc_member                  VARCHAR2(1)    OPTIONAL
  Indicates if the Registrant wants to be a member of
CIRA. Default value: Y
  Valid values:
    Y    Yes
    N    No

rgpc_description              VARCHAR2(2000) OPTIONAL
 Description of the Registrant.

rgpc_ref_info                  VARCHAR2(2000) OPTIONAL
 Reference information, free form text for Registrar use

ctc_language                   VARCHAR2(3)    MANDATORY
 Preferred language of administrative contact.
  Valid values:
    EN   English
    FR   French

ctc_last_name                  VARCHAR2(200)  MANDATORY
 Last name of administrative contact.

ctc_title                      VARCHAR2(3)    OPTIONAL
 Name title .
  Valid values:
    MR    Mr
    MRS   Mrs
    MS    Ms
    DR    Dr

ctc_first_name                 VARCHAR2(50)   MANDATORY
 First name of administrative contact.
```

```
ctc_middle_name                 VARCHAR2(50)   OPTIONAL
 Middle name of administrative contact.

ctc_job_title                  VARCHAR2(200)  OPTIONAL
 Job title of administrative contact.

ctc_nationality                VARCHAR2(3)    OPTIONAL
 Nationality of administrative contact.
  Valid values:
    CND  Canadian
    OTH  Foreign citizenship
    RES  Canadian resident

ctc_company                    VARCHAR2(200)  OPTIONAL
  Company of administrative contact.

ctc_phone                      VARCHAR2(40)   MANDATORY
  Phone number of administrative contact.

ctc_fax                        VARCHAR2(40)   OPTIONAL
  Fax number of administrative contact.

ctc_mobile                     VARCHAR2(40)   OPTIONAL
  Mobile phone number of administrative contact.

ctc_email                      VARCHAR2(100)  MANDATORY
  Email address of administrative contact.

ctc_second_email               VARCHAR2(100)  OPTIONAL
  Second email address of administrative contact.

ctc_other_phone                VARCHAR2(40)   OPTIONAL
  Other phone number of administrative contact. (Home,
etc..)

ctc_access_key                 VARCHAR2(2048) OPTIONAL
  PGP access key for encrypted message from CIRA.

addr_address_number            VARCHAR2(200)  OPTIONAL
  Civic number of administrative contact address.

addr_corporative               VARCHAR2(200)  OPTIONAL
  Additional delivery information. For example:  "Acme
Tower", "MIS Department"
```

```
addr_street_cat                VARCHAR2(3)    OPTIONAL
  Street type of administrative contact address
  Valid values:
    AV   Avenue
    BL   Boulevard
    CT   Court
    CR   Crescent
    DR   Drive
    HW   Highway
    PL   Place
    PZ   Plaza
    RD   Road
    SQ   Square
    ST   Street
    WY   Way

addr_street_name               VARCHAR2(200)  OPTIONAL
  Street name of administrative contact address.

addr_orientation               VARCHAR2(3)    OPTIONAL
  Street orientation of administrative contact address
  Valid values:
    E     East
    N     North
    NE    North-East
    NW    North-West
    S     South
    SE    South-East
    SW    South-West
    W     West

addr_office                    VARCHAR2(20)   OPTIONAL
  Office number of administrative contact address
  Example: Suite 340

addr_city                      VARCHAR2(200)  MANDATORY
  City of administrative contact address.

addr_province_state            VARCHAR2(50)   OPTIONAL
  Province or territory of administrative contact address,
  MANDATORY if in Canada.
  Valid values for Canada, Not validated in other
countries:
    AB   Alberta
```

BC	British Columbia
MB	Manitoba
NB	New Brunswick
NF	Newfoundland
NS	Nova Scotia
NT	Northwest Territories
NU	Nunavut
ON	Ontario
PE	Prince Edward Island
QC	Quebec
SK	Saskatchewan
YK	Yukon

addr_postal_code VARCHAR2(20) OPTIONAL

Postal code of administrative contact address, MANDATORY if in Canada.

addr_country VARCHAR2(200) MANDATORY

Country of administrative contact address. Valid values are ISO 3166 3-letter country codes. See annex B for the list of valid values.

ctc_tec_same_as_admin VARCHAR2(1) MANDATORY

Indicates if the administrative contact is also the technical contact.

If Y, the following ctc_tec_* and addr_tec_* parameters are ignored.

Valid values:

Y	Yes
N	No

ctc_tec_ctc_id NUMBER(16) OPTIONAL

Indicates the ID of an already registered contact that will be the technical contact for this registrant.

If filled, the following ctc_tec_* and addr_tec_* parameters are ignored.

ctc_tec_language VARCHAR2(3) MANDATORY

Preferred language of technical contact.

Valid values: c.f. ctc_language

ctc_tec_same_addr_admin VARCHAR2(1) OPTIONAL

Indicates if technical contact has the same address as the administrative contact. Default value: N

If Y, the following addr_tec_* parameters are ignored.

```
  Valid values:
    Y   Yes
    N   No

ctc_tec_last_name              VARCHAR2(200)  MANDATORY
  Last name of technical contact.

ctc_tec_title                  VARCHAR2(3)    OPTIONAL
  Name title
  Valid values: c.f. ctc_title

ctc_tec_first_name             VARCHAR2(50)   OPTIONAL
  First name of technical contact.

ctc_tec_midle_name             VARCHAR2(50)   OPTIONAL
  Middle name of technical contact.

ctc_tec_job_title              VARCHAR2(200)  OPTIONAL
  Job title of technical contact.

ctc_tec_company                VARCHAR2(200)  OPTIONAL
  Company of technical contact.

ctc_tec_phone                  VARCHAR2(40)   OPTIONAL
  Phone number of technical contact.

ctc_tec_fax                    VARCHAR2(40)   OPTIONAL
  Fax number of technical contact.

ctc_tec_mobile                 VARCHAR2(40)   OPTIONAL
  Mobile phone number of technical contact.


ctc_tec_email                  VARCHAR2(100)  MANDATORY
  Email address of technical contact.

ctc_tec_second_email           VARCHAR2(100)  OPTIONAL
  Second email address of technical contact.

ctc_tec_other_phone            VARCHAR2(40)   OPTIONAL
  Other phone number of technical contact. (Home, etc..)

addr_tec_address_number        VARCHAR2(200)  OPTIONAL
  Civic number of technical contact address.
```

```
addr_tec_corporative           VARCHAR2(200)  OPTIONAL
  Additional delivery information. For example:  "Acme
Tower", "MIS Department"

addr_tec_street_cat            VARCHAR2(3)    OPTIONAL
  Street type of technical contact address
  Valid values: c.f. addr_street_cat

addr_tec_street_name           VARCHAR2(200)  OPTIONAL
  Street name of technical contact address.

addr_tec_orientation           VARCHAR2(3)    OPTIONAL
  Street orientation of technical contact address
  Valid values: c.f. addr_street_orientation

addr_tec_office                VARCHAR2(20)   OPTIONAL
  Office number of technical contact address (Suite 340)

addr_tec_city                  VARCHAR2(200)  MANDATORY
  City of technical contact address.

addr_tec_province_state        VARCHAR2(50)   OPTIONAL
  Province or territory of technical contact address.
  Valid values for Canada, Not validated in other
countries,
  c.f. addr_province_state

addr_tec_postal_code           VARCHAR2(20)   OPTIONAL
  Postal code of technical contact address, MANDATORY if
in Canada.

addr_tec_country               VARCHAR2(200)  MANDATORY
  Country of technical contact address. Valid values are
ISO 3166 3-letter country codes. See annex B for the list
of valid values.
```

Request response additional parameters:

These parameters are present only if request status is AAP or URJ.

```
pris_id                        NUMBER(10)
 CIRA’s process instance number.

rgpc_no                        NUMBER(16)
```

```
 CIRA’s Registrant unique identifier.

user_acc                       NUMBER(16)
(socket-based web interface ONLY)
  Registrant user account to access CIRA’s site.

password                       VARCHAR2(20)
(socket-based web interface ONLY)
  Registrant password to access CIRA’s site.
```

EXISTING REGISTRANT NEW DOMAIN NAME

This procedure is used to register a domain name for an existing Registrant. The Registrant’s identification number is needed to submit this transaction. If the Registrant has approved CIRA’s Registrant Agreement and the requested domain name is available, the system will register and activate the domain name without further action. The status returned for an accepted request will be AAP or ACT.

The following values are possible:

```
AAP: Registration request processed and accepted. The
     domain name will be active when the registrant will
     have completed its registration
ACT: Registration request processed and accepted. The
     domain will be active within 24 hours.
MAN: Registration request processed, but needs
     verification by CIRA’s staff.
INC: Registration request is incomplete and was not
     processed. (Input parameters missing or with wrong
     values).
RJC: Registration request processed and rejected.
```

In the case of a processed request (AAP, ACT, MAN, RJC), a confirmation email is sent to the Registrant and to the Registrar. If the request is incomplete, no email is sent.

If a Registrar wants to check if a registration request will be accepted or rejected, it can ask the system to verify the request. By setting parameter transac_type to VERIF_REGIS_DOM, the system will only check the request without generating a transaction.

> ***Note:*** Response parameters 'pris_id' and 'effect_cost' will contain invalid values since no registration takes place.

Socket-based Web interface details

Registrars can submit their socket-based registration request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A registration request will generate the following return codes in XML:

```
<result>
   <status>(STATUS)</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected or incomplete status. All the values transmitted in the originating request will follow (input parameters). If the request is accepted, some additional fields (output parameters) will be included in the response. A successful transaction request will generate a status code with a value of AAP or ACT in the following format:

```
<result>
   <status>AAP</status>
   <input_parameter_name>(VALUE)</ input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
   <output_parameter_name>(VALUE)</ output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</ output_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an accepted request will generate a status code of AAP or ACT in the following format:

```
status: AAP
     (input parameter-name)='(input parameter-value)'
     ....
     (input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status: RJC
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(1)     MANDATORY
  Valid value
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface

transac_id                     VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    REGIS_DOM          Registration request transaction
    VERIF_REGIS_DOM    Registration request verification

force_manual_verif             VARCHAR2(1)     OPTIONAL
 Indicates the Registrant wants this request to be
reviewed by CIRA's staff. This option is used when the
Registrant qualifies to obtain a domain name registration,
but for some reason, the system will not allow its
```

```
registration. This parameter is ignored if the transaction
type is VERIF_REGIS_DOM. Default value: N
  Valid values
    Y   Force review of this request by CIRA’s staff.
    N   Use normal processing

rgpc_rant                      NUMBER(16)      MANDATORY
  Registrant identification number.  This number is
returned in the rgpc_no parameters of REGIS_DOM_RANT,
ENREG_RANT and LIST_RANT functions.

rgpc_subdomain                 VARCHAR2(50)    MANDATORY
  Domain name to register.

rgpc_trade_mark_registered     VARCHAR2(1)    OPTIONAL
 Indicates if the submitted domain name is a trademark.
Default value: N
Valid values:
    Y   Yes
    N   No

rgpc_ref_info                  VARCHAR2(2000) OPTIONAL
 Reference information, free form text for Registrar use

length_of_term                 NUMBER         OPTIONAL
 Number of years the domain will be registered.
  Valid values: integer between 1 and 10, default 1

nmsv_1_name                    VARCHAR2(200)  OPTIONAL
 Name of the first name server

nmsv_1_ip_address              VARCHAR2(100)  OPTIONAL
 First IP address

nmsv_1_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the first IP address, only IPV4 is
currently supported.

nmsv_2_name                    VARCHAR2(200)  OPTIONAL
 Name of a second name server

nmsv_2_ip_address              VARCHAR2(100)  OPTIONAL
 Second IP address
```

```
nmsv_2_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the second IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_3_name                    VARCHAR2(200)  OPTIONAL
 Name of a third name server

nmsv_3_ip_address              VARCHAR2(100)  OPTIONAL
 Third IP address
nmsv_3_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the third IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_4_name                    VARCHAR2(200)  OPTIONAL
 Name of a fourth name server

nmsv_4_ip_address              VARCHAR2(100)  OPTIONAL
 Fourth IP address

nmsv_4_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the fourth IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_5_name                    VARCHAR2(200)  OPTIONAL
 Name of a fifth name server

nmsv_5_ip_address              VARCHAR2(100)  OPTIONAL
 Fifth IP address

nmsv_5_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the fifth IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_6_name                    VARCHAR2(200)  OPTIONAL
 Name of a sixth name server

nmsv_6_ip_address              VARCHAR2(100)  OPTIONAL
```

```
 Sixth IP address

nmsv_6_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the sixth IP address, only IPV4 is
currently supported.
  Default IPV4
```

Request response additional parameters:

These parameters are present only if the request status is AAP.

```
pris_id                        NUMBER(10)
 CIRA's process instance number.

effect_cost                    NUMBER (3.2)
  Cost to Registrar, including applicable taxes.
```

NEW REGISTRANT WITH A NEW DOMAIN NAME

This procedure is used to register a new Registrant and a new domain name.

> ***Note:*** This procedure should not be used to register a domain name for an existing Registrant.

The result of the registration request is returned in the STATUS parameter. A rejected request will return RJC in the status parameter.

Returned values can be one of the following:

```
AAP: Registration request processed and accepted. The
     domain name will be active when the registrant will
     have completed its registration
ACT: Registration request processed and accepted. The
     domain will be active within 24 hours.
MAN: Registration request processed, but needs
     verification by CIRA's staff.
INC: Registration request is incomplete and was not
     processed. (Input parameters missing or with wrong
     values).
RJC: Registration request processed and rejected.
```

In the case of a processed request (AAP, ACT, MAN, RJC), a confirmation email is sent to the Registrant and to the Registrar. If the request is incomplete no email is sent.

If a Registrar wants to check if a registration request will be accepted or not, it can ask the system to verify the request. By setting parameter transac_type to VERIF_REGIS_DOM_RANT, the system will check the request without generating a transaction.

> ***Note:*** Response parameters 'pris_id', 'rgpc_no', 'effect_cost', 'user_acc' and 'password' will contain invalid values since no registration takes place.

Socket-based web interface details

Registrars can submit their socket-based registration request to:
https://cira2.cira.ca/cira/registraires

XML format details

A registration request will generate the following return codes in XML:

```
<result>
   <status>(STATUS)</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</ input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
   <output_parameter_name>(VALUE)</ output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</ output_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected or incomplete status. All the values transmitted in the originating request will follow (input parameters). If the request is accepted, some additional fields (output parameters) will be included in the response. A successful transaction request will generate a status code with a value of AAP or ACT in the following format:

```
<result>
   <status>AAP</status>
   <input_parameter_name>(VALUE)</ input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
   <output_parameter_name>(VALUE)</ output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</ output_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an accepted request will generate a status code with a value of AAP or ACT in the following format:

```
status: AAP
     (input parameter-name)='(input parameter-value)'
     ....
     (input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status: RJC
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Real-time completion of registration requests for socket-based web interface

Registrars who work with their Registrants via web sites and CIRA in real time may opt to complete all registration requests in real time. To do so Registrars must use the following procedure after receiving confirmation of a successful registration request from the CIRA system.

The returned parameters will include:

```
<user_acc>(user-name)</user_acc>
<password>(password)</password>
```

These are the CIRA supplied Registrant user code and password that the Registrant will be required to use to accept the CIRA Registrant Agreement on-line. Registrars can use these codes to establish transactional control over the Registrant's acceptance of the CIRA Registrant Agreement. Having passed on these values to the Registrant the Registrar's system should re-direct the Registrant to:

https://cira2.cira.ca/cira/detenteurs and supply the following parameters:

```
  URL: Contains the URL that the Registrar wishes CIRA to
       transfer to after the Registrant has completed its
       approval of the CIRA Registrant Agreement;
  TID: Registrar generated transaction identifier that
       will be returned when transferring to the
       Registrar's URL (free format string);

  TYPE=REGIS_DOM_RANT: confirms the transaction type.
```

Once the Registrant has accepted or rejected the CIRA Registrant Agreement, the Registrant's browser will be re-directed to the URL specified in the Registrar's request and the CIRA system will supply a result parameter RES = A (Registrant accepted) or R (Registrant refused) as well as the TID parameter.

The Registrar will now be able to complete its transaction with the Registrant.

Request parameters:

```
user                          NUMBER(16)      MANDATORY
  Registrar user id

pwd                           VARCHAR2(30)    MANDATORY
  Registrar password

XML                           VARCHAR2(10)    MANDATORY
  Valid value
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface

transac_id                    VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                  VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    REGIS_DOM_RANT        Registration request transaction
    VERIF_REGIS_DOM_RANT  Registration request
                          verification

look_for_existing_rant        VARCHAR2(1)     OPTIONAL
  This indicates that the registrant may be already
registered. If set to 'Y', the system will look for an
existing registrant having the matching rgpc_name and
ctc_email or ctc_phone. If only one existing registrant
matches these parameter values, the system will act as if
a REGIS_DOM transaction was called. In this case,
parameters absent from the REGIS_DOM parameters list will
```

be ignored and response parameters 'user_acc' and 'password' will be empty.

Valid values:

Y	Look for existing registrant
N	Create new registrant without checking

rgpc_name VARCHAR2(200) MANDATORY
Name of the Registrant.

rgpc_legal_type VARCHAR2(3) MANDATORY
Registrant legal type.

Valid values:

CCO	Corporation (Canada or Canadian province or territory)
CCT	Canadian citizen
RES	Permanent Resident of Canada
GOV	Government or government entity in Canada
EDU	Canadian Educational Institution
ASS	Canadian Unincorporated Association
HOP	Canadian Hospital
PRT	Partnership Registered in Canada
TDM	Trade-mark registered in Canada (by a non-Canadian owner)
TRD	Canadian Trade Union
PLT	Canadian Political Party
LAM	Canadian Library, Archive or Museum
TRS	Trust established in Canada
ABO	Aboriginal Peoples (individuals or groups)indigenous to Canada
INB	Indian Band recognized by the Indian Act of Canada
LGR	Legal Representative of a Canadian Citizen or Permanent Resident
OMK	Official mark registered in Canada
MAJ	Her Majesty the Queen

rgpc_subdomain VARCHAR2(50) MANDATORY
Domain name to register.

rgpc_member VARCHAR2(1) OPTIONAL
Default 'Y'
Indicates if the Registrant wants to be a member of CIRA.

```
  Valid values:
    Y   Yes
    N   No

rgpc_description               VARCHAR2(2000) OPTIONAL
 Description of the Registrant.

rgpc_trade_mark_registered     VARCHAR2(1)    OPTIONAL
 Indicates if the submitted domain name is a trademark.
  Valid values:
    Y    Yes
    N    No

rgpc_ref_info                  VARCHAR2(2000) OPTIONAL
 Reference information, free form text for registrar use

length_of_term                 NUMBER         OPTIONAL
 Number of years the domain will be registered.
  Valid values: integer between 1 and 10, default 1

nmsv_1_name                    VARCHAR2(200)  OPTIONAL
 Name of the first name server

nmsv_1_ip_address              VARCHAR2(100)  OPTIONAL
 First IP address

nmsv_1_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the first IP address, only IPV4 is
currently supported.

nmsv_2_name                    VARCHAR2(200)  OPTIONAL
 Name of a second name server

nmsv_2_ip_address              VARCHAR2(100)  OPTIONAL
 Second IP address

nmsv_2_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the second IP address, only IPV4 is
currently supported.
  Default IPV4
nmsv_3_name                    VARCHAR2(200)  OPTIONAL
 Name of a third name server
nmsv_3_ip_address              VARCHAR2(100)  OPTIONAL
```

```
 Third IP address

nmsv_3_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the third IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_4_name                    VARCHAR2(200)  OPTIONAL
 Name of a fourth name server

nmsv_4_ip_address              VARCHAR2(100)  OPTIONAL
 Fourth IP address

nmsv_4_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the fourth IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_5_name                    VARCHAR2(200)  OPTIONAL
 Name of a fifth name server

nmsv_5_ip_address              VARCHAR2(100)  OPTIONAL
 Fifth IP address

nmsv_5_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the fifth IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_6_name                    VARCHAR2(200)  OPTIONAL
 Name of a sixth name server

nmsv_6_ip_address              VARCHAR2(100)  OPTIONAL
 Sixth IP address

nmsv_6_ip_addr_type            VARCHAR2(4)    OPTIONAL
Default IPV4
 IP address type for the sixth IP address, only IPV4 is
currently supported.
  Default IPV4

ctc_language                   VARCHAR2(3)    MANDATORY
```

```
 Preferred language of administrative contact.
  Valid values:
    EN   English
    FR   French

ctc_last_name                  VARCHAR2(200)  MANDATORY
 Last name of administrative contact.

ctc_title                      VARCHAR2(3)    OPTIONAL
 Name title .
  Valid values:
    MR    Mr
    MRS   Mrs
    MS    Ms
    DR    Dr

ctc_first_name                 VARCHAR2(50)   MANDATORY
 First name of administrative contact.

ctc_midle_name                 VARCHAR2(50)   OPTIONAL
 Middle name of administrative contact.

ctc_job_title                  VARCHAR2(200)  OPTIONAL
 Job title of administrative contact.

ctc_nationality                VARCHAR2(3)    OPTIONAL
 Nationality of administrative contact.
  Valid values:
    CND   Canadian
    OTH   Foreign citizenship
    RES   Canadian resident

ctc_company                    VARCHAR2(200)  OPTIONAL
  Company of administrative contact.

ctc_phone                      VARCHAR2(40)   MANDATORY
  Phone number of administrative contact. This phone
number will be published by the WHOIS.

ctc_fax                        VARCHAR2(40)   OPTIONAL
  Fax number of administrative contact.

ctc_mobile                     VARCHAR2(40)   OPTIONAL
  Mobile phone number of administrative contact.
```

```
ctc_email                      VARCHAR2(100)  MANDATORY
  Email address of administrative contact.

ctc_second_email               VARCHAR2(100)  OPTIONAL
  Second email address of administrative contact.

ctc_other_phone                VARCHAR2(40)   OPTIONAL
  Other phone number of administrative contact. (Home,
etc..) This phone number will not be published by the
WHOIS.

ctc_access_key                 VARCHAR2(2048) OPTIONAL
  PGP access key for encrypted message from CIRA.

addr_address_number            VARCHAR2(200)  OPTIONAL
  Civic number of administrative contact address.

addr_corporative               VARCHAR2(200)  OPTIONAL
  Additional delivery information. For example:  "Acme
Tower", "MIS Department"

addr_street_cat                VARCHAR2(3)    OPTIONAL
  Street type of administrative contact address
  Valid values:
    AV   Avenue
    BL   Boulevard
    CT   Court
    CR   Crescent
    DR   Drive
    HW   Highway
    PL   Place
    PZ   Plaza
    RD   Road
    SQ   Square
    ST   Street
    WY   Way


addr_street_name               VARCHAR2(200)  OPTIONAL
  Street name of administrative contact address.

addr_orientation               VARCHAR2(3)    OPTIONAL
  Street orientation of administrative contact address
  Valid values:
    E    East
```

N	North
NE	North-East
NW	North-West
S	South
SE	South-East
SW	South-West
W	West

```
addr_office                   VARCHAR2(20)   OPTIONAL
  Office number of administrative contact address
  Example: Suite 340

addr_city                     VARCHAR2(200)  MANDATORY
  City of administrative contact address.

addr_province_state           VARCHAR2(50)   OPTIONAL
  Province or territory of administrative contact address,
  MANDATORY if in Canada.
```

Valid values for Canada, not validated in other countries:

AB	Alberta
BC	British Columbia
MB	Manitoba
NB	New Brunswick
NF	Newfoundland
NS	Nova Scotia
NT	Northwest Territories
NU	Nunavut
ON	Ontario
PE	Prince Edward Island
QC	Quebec
SK	Saskatchewan
YK	Yukon

```
addr_postal_code              VARCHAR2(20)   OPTIONAL
  Postal code of administrative contact address, MANDATORY
if in Canada.

addr_country                  VARCHAR2(200)  MANDATORY
  Country of administrative contact address. Valid values
are ISO 3166 3-letter country codes. See annex B for the
list of valid values. Code for Canada is CAN.

ctc_tec_same_as_admin         VARCHAR2(1)    MANDATORY
```

```
  Indicates if the administrative contact is also the
technical contact.
  If Y, the following ctc_tec_* and addr_tec_* parameters
are ignored.
  Valid values:
    Y   Yes
    N   No

ctc_tec_ctc_id                NUMBER(16)     OPTIONAL
  Indicates the ID of an already registered contact that
will be the technical contact for this Registrant. If
filled, the following ctc_tec_* and addr_tec_* parameters
are ignored.

ctc_tec_language              VARCHAR2(3)    MANDATORY
  Preferred language of technical contact.
  Valid values: c.f. ctc_language

ctc_tec_same_addr_admin       VARCHAR2(1)    OPTIONAL
Default value: N
  Indicates if the technical contact has the same address
  as the administrative contact.
  If Y, the following addr_tec_* parameters are ignored.
  Valid values:
    Y   Yes
    N   No

ctc_tec_last_name             VARCHAR2(200)  MANDATORY
  Last name of technical contact.

ctc_tec_title                 VARCHAR2(3)    OPTIONAL
  Name title
  Valid values: c.f. ctc_title

ctc_tec_first_name            VARCHAR2(50)   OPTIONAL
  First name of technical contact.

ctc_tec_midle_name            VARCHAR2(50)   OPTIONAL
  Middle name of technical contact.

ctc_tec_job_title             VARCHAR2(200)  OPTIONAL
  Job title of technical contact.

ctc_tec_company               VARCHAR2(200)  OPTIONAL
  Company of technical contact.
```

```
ctc_tec_phone                  VARCHAR2(40)   OPTIONAL
  Phone number of technical contact.

ctc_tec_fax                    VARCHAR2(40)   OPTIONAL
  Fax number of technical contact.

ctc_tec_mobile                 VARCHAR2(40)   OPTIONAL
  Mobile phone number of technical contact.

ctc_tec_email                  VARCHAR2(100)  MANDATORY
  Email address of technical contact.

ctc_tec_second_email           VARCHAR2(100)  OPTIONAL
  Second email address of technical contact.

ctc_tec_other_phone            VARCHAR2(40)   OPTIONAL
  Other phone number of technical contact. (Home, etc..)
addr_tec_address_number        VARCHAR2(200)  OPTIONAL
  Civic number of technical contact address.

addr_tec_corporative           VARCHAR2(200)  OPTIONAL
  Additional delivery information. For example:  "Acme
Tower", "MIS Department"

addr_tec_street_cat            VARCHAR2(3)    OPTIONAL
  Street type of technical contact address
  Valid values: c.f. addr_street_cat

addr_tec_street_name           VARCHAR2(200)  OPTIONAL
  Street name of technical contact address

addr_tec_orientation           VARCHAR2(3)    OPTIONAL
  Street orientation of technical contact address
  Valid values: c.f. addr_street_orientation

addr_tec_office                VARCHAR2(20)   OPTIONAL
  Office number of technical contact address

addr_tec_city                  VARCHAR2(200)  MANDATORY
  City of technical contact address.

addr_tec_province_state        VARCHAR2(50)   OPTIONAL
  Province or territory of technical contact address.
  Valid values for Canada, Not validated in other
countries,
  c.f. addr_province_state

addr_tec_postal_code           VARCHAR2(20)   OPTIONAL
  Postal code of technical contact address, MANDATORY if
in Canada.

addr_tec_country               VARCHAR2(200)  MANDATORY
  Country of technical contact address. Valid values are
ISO 3166 3-letter country codes. See annex B for the list
of valid values.
```

Request response additional parameters:

These parameters are present only if request status is AAP or URJ.

```
pris_id                        NUMBER(10)
 CIRA's process instance number.

rgpc_no                        NUMBER(16)
 CIRA's Registrant unique identifier.

effect_cost                    NUMBER(3.2)
  Cost to Registrar, including applicable taxes.

user_acc                       NUMBER(16)
(Socket-based Web forms interface ONLY)
  Registrant user account to access CIRA's site.

password                       VARCHAR2(20)
(Socket-based Web forms interface ONLY)
  Registrant password to access CIRA's site.
```

EXTENSION PERIOD FOR REGISTRATION PROCESS

The Registrar can submit a request for a 7-day extension period of a registration process by submitting a DELAY_DOMAIN_APPROVAL transaction. Only one extension will be granted per registration process. The Registrant will be given an additional 7 days to accept the CIRA Registrant Agreement.

The result of the approval process extension request will be returned in the STATUS parameter and will have one of the following values:

```
AAP: Approval process extension request accepted.
RJC: Approval process extension request rejected.
```

Socket-based web interface details

Registrars can submit their socket-based extension request to:

https://cira2.cira.ca/cira/registraires

XML format details

If the requested is accepted, the response will be in the following format:

```
  <result>
      <status>AAP</status>
      <input_parameter_name>(VALUE)</input_parameter_name>
       ......
      <input_parameter_name>(VALUE)</input_parameter_name>
  </result>
```

If the requested is rejected, the response will be in the following format:

```
<result>
      <status>RJC</status>
      <error>
         <code>(ERROR CODE)</code>
         <description>(ERROR DESCRIPTION)</description>
      </error>
      <input_parameter_name>(VALUE)</input_parameter_name>
       ......
      <input_parameter_name>(VALUE)</input_parameter_name>
   </result>
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for an accepted request will be in the following format:

```
status: AAP
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status: RJC
errors:
(return code number): (return code text, optional)
```

```
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Request parameters:

```
user                           NUMBER(16)     MANDATORY
Registrar user id

pwd                            VARCHAR2(30)   MANDATORY
Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y  Response in XML format. Mandatory for Socket-based
       web interface
    N  Other interface

transac_id                     VARCHAR2(50)   OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(20)   MANDATORY
  Transaction type
  Valid value:
    DELAY_DOMAIN_APPROVAL

rgpc_subdomain               VARCHAR2(50)    MANDATORY
   Domain name to accept
reason                         VARCHAR2(200)  OPTIONAL
   Reason for delay request
```

DOMAIN NAME AVAILABILITY CHECK

This procedure is used to determine the availability of a domain name and is available through socket-based web interface or email interface.

Socket-based web interface details

Registrars can submit their socket-based domain name availability check request to: https://cira2.cira.ca/cira/rinfo.

Request parameters:

```
user                          NUMBER(16)      MANADTORY
Registar user id

pwd                           VARCHAR2(30)    MANADTORY
Registrar password

XML                            VARCHAR2(10)    MANDATORY
 Valid value:
    Y  Response in XML format. Mandatory for socket-based
       web interface

transac_type                   VARCHAR2(10)    MANDATORY
  Transaction type
  Valid values:
    VALID_DOM

DOM                            VARCHAR2(50)    MANDATORY
   Domain name to check.
```

VALID_DOM using tokens:

To use a token interface you must first set the token, then call the same function by specifying token=parameter instead of user/pwd.

Rules:

4. A token expires 24 hours from the moment it is set.
5. It can only be used from the ip that set it.
6. A registrar can set no more than 10 live tokens at a time.

Using the token interface involves two steps:

1) Set token

Tokens can be set at https://cira2.cira.ca/cira/rinfo

With:

```
Request parameters:
set_token                      VARCHAR2(1)    MANDATORY
  Valid value
    Y
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
```

```
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
```

For eg. by using this sample call:
https://cira2.cira.ca/cira/rinfo?XML=Y&user=???&pwd=??&set_token=y&transac_type=VALID_DOM
You will set one token:

Reply:

```
<result>
<status>ok</status>
<token>3150289.6220443.venus.64.230.122.119.22545956.1525.
2320780</token>
</result>
```

2) Use token

To use token set call to http://cira2.cira.ca/cira/rinfo

```
Request parameters:
token                          VARCHAR2(60)
MANDATORY
  Registrar token
XML                            VARCHAR2(10)    MANDATORY
 Valid value:
    Y  Response in XML format. Mandatory for socket-based
       web interface

transac_type                   VARCHAR2(10)    MANDATORY
  Transaction type
  Valid values:
    VALID_DOM

DOM                            VARCHAR2(50)    MANDATORY
   Domain name to check.
```

For example: VALID_DOM call over token just set will be:

http://cira2.cira.ca/cira/rinfo?XML=Y&token=3150295.31780098.venus.64
.230.122.119.22545962.1526.2320726&transac_type=VALID_DOM&DOM=this.ca

XML format details

If the requested domain name is available, the response will be in the following format:

```
<result>
       <status>0</status>
       <DOM>(VALUE)</DOM>
</result>
```

If the requested domain name is not available, then the response will be in the following format:

```
<result>
  <error>
     <code>(ERROR CODE)</code>
     <description>(ERROR DESCRIPTION)</description>
  </error>
  <DOM>(VALUE)</DOM>
</result>
```

Classic email interface details

If the requested domain name is available, the response will be in the following format:

```
status: 0
DOM='(domain name)'
```

If the domain name is not available, the response will be in the following format:

```
errors:
(return code number): (return code text, optional)
end of error
DOM='(domain name)'
```

One of several error codes and descriptions will explain why the domain is not available.

SENDING USER CODE AND PASSWORD

This procedure sends an email message to a Registrant’s contact with its user code and password.

When a request to register a new Registrant is made, a user account is created for its administrative contact and an email containing a user code and password is sent to that administrative contact. Only administrative contacts can have a user account.

A registrant may need to access restricted sections of the CIRA web site to acknowledge information changes, approve transfers or vote at CIRA elections. To access these restricted sections, a registrant needs a user code and password.

The result of the request is returned in the STATUS parameter. It can have one of the following values:

```
OK:  Success, an email message will be sent.
ERR: Failure, error codes are returned.
```

Socket-based web interface details

Registrars can submit their socket-based request to:

https://cira2.cira.ca/cira/registraires

XML format details

A request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed request will generate a response in the following format:

```
<result>
   <status>ERR</status>

   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    EMAIL_PWD

rgpc_no                        NUMBER(16)      MANDATORY
 Registrant unique identifier.



ctc_id                         NUMBER(16)      OPTIONAL
  Contact unique identifier.  If missing, the system will
use the administrative contact identifier.
```

REGISTRANT LIST

This procedure searches for one or more Registrar's Registrants. Search criteria can be on the Registrant name or the domain name. The procedure returns a list of Registrants that match the search criteria. Information returned includes a Registrant's name, identification number, and status.

Socket-based web interface details

Registrars can submit their socket-based search request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A search request will generate the following return codes in XML:

```
<result>
   <no>( number of Registrant found, can be zero)</no>
   <rgpc>
      <rgpc_name>(Registrant name)</rgpc_name>
      <rgpc_no>(Registrant identification number)</rgpc_no>
      <status>(Registrant status)</status>
   </rgpc>
     ....
   <rgpc>
      <rgpc_name>(Registrant name)</rgpc_name>
      <rgpc_no>(Registrant identification number)</rgpc_no>
      <status>(Registrant status)</status>
   </rgpc>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If the request cannot be processed, the system will generate the following return codes in XML:

```
<result>
   <no>0</no>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for a search request will be in the following format:

```
no: (number of Registrant found, can be zero)
rgpc_name_1='(Registrant name)'
rgpc_no_1='(Registrant identification number)'
status_1='(Registrant status)'
```

```
rgpc_name_2='(Registrant name)'
rgpc_no_2='(Registrant identification number)'
status_2='(Registrant status)''
....
rgpc_name_N='(Registrant name)'
rgpc_no_N='(Registrant identification number)'
status_N='(Registrant status)'
```

If the request cannot be processed, the body of the response email will be in the following format:

```
 no: 0
 errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Request parameters:

```
user                           NUMBER(16)     MANDATORY
Registrar user id

pwd                            VARCHAR2(30)   MANDATORY
Registrar password

XML                            VARCHAR2(10)   MANDATORY
  Valid values:
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface
transac_type                   VARCHAR2(10)   MANDATORY
  Transaction type
  Valid values:
    LIST_RANT

dom                           VARCHAR2(50)    MANDATORY
  Domain name of Registrant being searched.  Character '*'
can be used at the end of the parameter as a wildcard
character representing one or several characters. For
example, 'Cos*' will bring back a list of Registrants
having a domain beginning with 'Cos'.
```

```
If domain name is not a criterion, do not put '*',
otherwise registrants without domains will not be
retrieved.

rant                      VARCHAR2(200)   MANDATORY
  Name of Registrant being searched.  Character '*' can be
used at the end of the parameter as a wildcard character
representing one or several characters.  For example,
'Syl*' will bring back a list of Registrants having a name
beginning with 'Syl'.

If Registrant name is not a criterion, do not put '*' in
the parameter.

usac_id              NUMBER(16)     OPTIONAL
  User id of the Registrant
```

Response additional parameters:

```
rgpc_name                      VARCHAR2 (200)
  Registrant name.

rgpc_no                        NUMBER(16)
  Registrant identification number.

status                         VARCHAR2 (3)
  Registrant status.
    Valid values:
     AAP  Awaiting Registrant approval of CIRA agreement
     ACT  Registered Registrant
```

DOMAIN NAME LIST

The purpose of this procedure is to search for the domains of a Registrant. The procedure returns a list of domains with their identification number and status.

Socket-based web interface details

Registrars can submit their socket-based search request to:

https://cira2.cira.ca/cira/rinfo

Request parameters:

```
user                           NUMBER(16)     MANDATORY
Registrar user id

pwd                            VARCHAR2(30)   MANDATORY
Registrar password

XML                            VARCHAR2(10)   MANDATORY
  Valid values:
    Y  Response in XML format. Mandatory for socket-based
       web interface

transac_type                   VARCHAR2(10)   MANDATORY
  Transaction type
  Valid values:
    LIST_DOM

rant_no                        NUMBER(16)
  Registrant identification number.
```

LIST_DOM using tokens:

To use a token interface you must first set the token, then call the same function by specifying token=parameter instead of user/pwd.

Rules:

7. A token expires 24 hours from the moment it is set.
8. It can only be used from the ip that set it.
9. A registrar can set no more than 10 live tokens at a time.

Using the token interface involves two steps:

1) Set token

Tokens can be set at https://cira2.cira.ca/cira/rinfo

With:

```
Request parameters:
set_token                      VARCHAR2(1)    MANDATORY
  Valid value
    Y
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
```

```
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
```

For eg. by using this sample call:
`https://cira2.cira.ca/cira/rinfo?XML=Y&user=???&pwd=??&set_token=y&transac_type=LIST_DOM`

You will set one token:

Reply:

```
<result>
<status>ok</status>
<token>3150289.6220443.venus.64.230.122.119.22545956.1525.
2320780</token>
</result>
```

2) Use token

To use token set call to http://cira2.cira.ca/cira/rinfo

```
Request parameters:
token                          VARCHAR2(60)
MANDATORY
  Registrar token
XML                            VARCHAR2(10)   MANDATORY
  Valid values:
    Y  Response in XML format. Mandatory for socket-based
       web interface

transac_type                   VARCHAR2(10)   MANDATORY
  Transaction type
  Valid values:
    LIST_DOM

rant_no                        NUMBER(16)
  Registrant identification number.
```

For example: LIST_DOM call over token just set will be:

http://cira2.cira.ca/cira/rinfo?XML=Y&token=3150295.31780098.venus.64
.230.122.119.22545962.1526.2320726&transac_type=LIST_DOM&rant_ no=1

XML format details

A search request will generate the following return codes in XML:

```
<result>
   <no>( number of domain found, can be zero)</no>
   <rgpc>
      <rgpc_subdomain>(domain name)</rgpc_subdomain>
      <rgpc_no>(domain identification number)</rgpc_no>
      <status>(domain status)</status>
   </rgpc>
     ....
   <rgpc>
      <rgpc_subdomain>(domain name)</rgpc_subdomain>
      <rgpc_no>(domain identification number)</rgpc_no>
      <status>(domain status)</status>
   </rgpc>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If the request can't be processed the system will generate the following return codes in XML:

```
<result>
   <no>0</no>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for a search request will be in the following format:

```
 no: (number of domain found, can be zero)
 rgpc_name_1='(domain name)'
 rgpc_no_1='(domain identification number)'
 status_1='(domain status)'
 rgpc_name_2='(domain name)'
 rgpc_no_2='(domain identification number)'
 status_2='(domain status)''
....
 rgpc_name_N='(domain name)'
 rgpc_no_N='(domain identification number)'
 status_N='(domain status)'
```

If the request cannot be processed, the body of the response email will be in the following format:

```
 no: 0
 errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Request response additional parameters:

```
rgpc_subdomain                 VARCHAR2(50)
  Domain name.

rgpc_no                        NUMBER(16)
 Domain identification number.

status                         VARCHAR2(3)
 Domain status.
    Valid values:
     AAP   Awaiting Registrant approval of CIRA agreement
     ACT   Active domain
     CNF   Active domain being in process of
           annual confirmation
     MAN   Domain awaiting CIRA approval
     SUS   Suspended domain
     ANR   Active domain that will not be renewed
```

CHAPTER 5 REGISTRATION OF CANCELLED DOMAIN NAMES

On each day starting December 31, 2001 at 23:59:59 Ottawa time, all domain name registrations that were suspended 30 days ago, and were not renewed will be cancelled. They will get a status of TBR (to be released). Registrations with a status TBR will be considered as lapsed (LPS) by the system. This means that the domain will not be available for registration and WHOIS inquiries will respond “Name to be released at 12:00 on <date>”. Starting January 24, 2002 at 09:00 Ottawa time, CIRA will make these cancelled domain names available for registration on a daily basis.

CIRA has developed two new functions for making the domain names that are the subject of cancelled registrations available to others for registration. Both new functions, VALID_TBR_DOM and REGIS_TBR_DOM, will be available through the CIRA system January 24, 2002.

The domain names subject of cancelled registrations will be made available for registration using the REGIS_TBR_DOM function during a special registration period that extends from 12:00 to 18:00 Ottawa time on the following day. It is important to note that only one transaction every 5 seconds will be allowed during this time period when registering domain names using VAILD_TBR_DOM and REGIS_TBR_DOM.

In anticipation of high volume of registrations expected on the first day, an exception will be made on January 24, 2002 when the first special registration period will run for 30 hours that is from 09:00 Ottawa time on January 24 to 15:00 Ottawa time on January 25, 2002.

To further facilitate this process, a new server will be dedicated to the task of supporting the registration of TBR domains. It will support all 3 interfaces (Web, Socket-based web, and Email interfaces). Registrars who work with their Registrants via websites and CIRA in real time can access this new server from the following location: www.cira.ca/cira/tbr.

Registrars can submit their socket-based requests via https://cira3.cira.ca/cira/tbr.

Registrars can submit their email-based requests to: tbr@cira.ca.

A new validation function VALID_TBR_DOM has also been developed. Both REGIS_TBR_DOM and VALID_TBR_DOM will be available from the new server. To save time, CIRA will not encrypt registry responses to email based registrar requests for registration using this special procedure. If a domain name is not registered according to this new procedure (REGIS_TBR_DOM) during its special registration period, then immediately following that period, it will be made available for registration according to normal procedures.

> ***Note:*** Although this new procedure can only be used to register domain names for existing registrants, it can be used to register any domain name, meaning domain names that have not been subject to suspension and cancellation.

AVAILABILITY CHECK REQUEST FOR DOMAIN NAMES TO BE RELEASED

This function indicates the availability of a domain name.

> ***Note:*** It is important to note that until 18:00 hours of the day the domain is cancelled, a WHOIS request through email or web socket will return a status of ‘TBR’ (to be released) even if the domain name was registered between 12:00 and 18:00 hours. To get the real status of such a domain name, you must use the VALID_TBR_DOM function.

Socket-based Web Interface Details

Registrars can submit their SSL socket-based domain name availability check requests to: https://cira3.cira.ca/cira/tbr

Email Interface Details

Registrars can submit their email based domain name availability check requests to: tbr@cira.ca.

XML format details

If the requested domain name is available, the response will be in the following format:

```
<result>
   <status>0</status>
   <DOM>(VALUE)</DOM>
</result>

If the requested domain name is not available, then the
response
will be in the following format:
<result>
   <error>
          <code>(ERROR CODE)</code>
           <description>(ERROR DESCRIPTION)</description>
   </error>
   <DOM>(VALUE)</DOM>
</result>
```

Classic email interface details

If the requested domain name is available, the response will be in the following format:

```
status: 0
DOM='(domain name)'

If the domain name is not available, the response will be
in the
following format:

errors:(return code number): (return code text, optional)
end of error
DOM='(domain name)'
```

One of several error codes and descriptions will indicate why the domain is not available.

Request parameters:

```
user                       NUMBER(16)         MANDATORY
Registar user id

pwd                        VARCHAR2(30)       MANADTORY
Registrar password

XML                        VARCHAR2(10)       MANDATORY
 Valid value:
    Y   Response in XML format. Mandatory for socket-based
```

```
        web interface

transac_type                    VARCHAR2(10)    MANDATORY
  Transaction type
  Valid values:
    VALID_TBR_DOM

LANGUE                          VARCHAR2(2)     MANDATORY
  Description's language
   Valid values:
    en   English
    fr   French

DOM                             VARCHAR2(50)    MANDATORY
   Domain name to check.
```

REGISTRATION OF DOMAIN NAMES SUBJECT OF CANCELLED REGISTRATIONS

This function registers a new domain name for an existing Registrant.

> ***Note:*** This is the only function that allows registration of a domain name subject of a cancelled registration.

The result of such a registration request is returned in the STATUS parameter. It can have one of the following values:

```
AAP: Registration request processed and accepted. The
     domain name will be active when the registrant will
     have completed its registration
ACT: Registration request processed and accepted. The
     domain will be active within 24 hours.
MAN: Registration request processed, but needs
     verification by CIRA's staff.
INC: Registration request is incomplete and was not
     processed. (Input parameters missing or have wrong
     values).
RJC: Registration request processed and rejected.
```

In case of a processed request (AAP, ACT, MAN, RJC), a confirmation email is sent to the Registrant and to the Registrar. If the request is incomplete, no email is sent.

If a Registrar wants only to check whether a registration request will be accepted or not, he/she can ask the system to only verify the request. By setting parameter transac_type to VERIF_REGIS_TBR_DOM, the function will only check the request without generating a transaction.

Note: Response parameters 'pris_id' and 'effect_cost' will contain invalid values since no registration takes place.

Socket-based web interface details

Registrars can submit their SSL socket-based domain name availability check requests to: https://cira3.cira.ca/cira/tbr

Email interface details

Registrars can submit their email based domain name availability check request to: tbr@cira.ca.

XML format details

A registration request will generate the following return codes in XML:

```
<result>
   <status>(STATUS)</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</ input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
   <output_parameter_name>(VALUE)</ output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</ output_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected or incomplete status. All the values transmitted in the originating request will follow (input parameters). If the request is accepted, some additional fields (output parameters) will conclude the response.

A successful transaction request will generate a status code with a value of 'AAP' in the following format:

```
<result>
   <status>AAP</status>
   <input_parameter_name>(VALUE)</ input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
   <output_parameter_name>(VALUE)</ output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</ output_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an accepted request will be in the following format:

```
 status: AAP
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
 status: RJC
 errors:
(return code number): (return code text, optional)
 end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(1)     MANDATORY
  Valid value
    Y    Response in XML format. Mandatory for
         Socket-based web interface
    N    Other interface
```

```
transac_id                     VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    REGIS_TBR_DOM          Registration request
transaction
    VERIF_REGIS_TBR_DOM    Registration request
verification

force_manual_verif             VARCHAR2(1)     OPTIONAL
 Indicates the Registrant wants this request to be
reviewed by CIRA’s staff. This option is used when the
Registrant qualifies to obtain a domain name registration,
but for some reason, the system will not allow its
registration. This parameter is ignored if the transaction
type is VERIF_REGIS_TBR_DOM.
  Valid values
    Y   Force review of this request by CIRA's staff.
    N   Use normal processing

rgpc_rant                      NUMBER(16)      MANDATORY
  Registrant identification number.  This number is
returned in the rgpc_no parameters of REGIS_DOM_RANT,
ENREG_RANT and LIST_RANT
functions.

rgpc_subdomain                 VARCHAR2(50)    MANDATORY
  Domain name to register.

rgpc_trade_mark_registered     VARCHAR2(1)    OPTIONAL
 Indicates if the submitted domain name is a trademark.
  Valid values:
    Y   Yes
    N   No

rgpc_ref_info                  VARCHAR2(2000) OPTIONAL
 Reference information, free form text for Registrar use

length_of_term                 NUMBER         OPTIONAL
 Number of years the domain will be registered.
  Valid values: integer between 1 and 10, default 1

nmsv_1_name                    VARCHAR2(200)  OPTIONAL
```

```
      Name of the first name server

nmsv_1_ip_address              VARCHAR2(100)  OPTIONAL
 First IP address
nmsv_1_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the first IP address, only IPV4 is
currently supported.

nmsv_2_name                    VARCHAR2(200)  OPTIONAL
 Name of a second name server

nmsv_2_ip_address              VARCHAR2(100)  OPTIONAL
 Second IP address

nmsv_2_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the second IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_3_name                    VARCHAR2(200)  OPTIONAL
 Name of a third name server

nmsv_3_ip_address              VARCHAR2(100)  OPTIONAL
 Third IP address

nmsv_3_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the third IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_4_name                    VARCHAR2(200)  OPTIONAL
 Name of a fourth name server

nmsv_4_ip_address              VARCHAR2(100)  OPTIONAL
 Fourth IP address

nmsv_4_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 IP address type for the fourth IP address, only IPV4 is
currently supported.
  Default IPV4
```

```
nmsv_5_name                     VARCHAR2(200)  OPTIONAL
 Name of a fifth name server

nmsv_5_ip_address               VARCHAR2(100)  OPTIONAL
 Fifth IP address

nmsv_5_ip_addr_type             VARCHAR2(4)    OPTIONAL
                                               Default IPV4
 IP address type for the fifth IP address, only IPV4 is
currently supported.
  Default IPV4

nmsv_6_name                     VARCHAR2(200)  OPTIONAL
 Name of a sixth name server

nmsv_6_ip_address               VARCHAR2(100)  OPTIONAL
 Sixth IP address

nmsv_6_ip_addr_type             VARCHAR2(4)    OPTIONAL
                                               Default IPV4
 IP address type for the sixth IP address, only IPV4 is
currently supported.
  Default IPV4
```

Request Response Additional Parameters:

These parameters are present only if request status is AAP.

```
pris_id                         NUMBER(10)
 CIRA's process instance.

effect_cost                     NUMBER(3.2)
  Cost to Registrar, including applicable taxes.
```

CHAPTER 6 GETTING INFORMATION

REGISTRANT GENERAL INFORMATION

This function returns general information on a Registrant. It can be use to retrieve current registered information or to retrieve new and unconfirmed information.

Socket-based web interface details

Registrars can submit their socket-based information requests to:

https: //cira2.cira.ca/cira/registraires

XML format details

A registration request will generate the following return codes in XML:

```
<result>
   <input_parameter_name>(VALUE)</ input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
   <output_parameter_name>(VALUE)</ output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</ output_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed request will generate a response in the following format:

```
<result>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an accepted information request will be in the following format:

```
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface
```

```
transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    GET_RANT_INFO        Get Registrant information
    GET_RANT_INFO_NC     Get modified and non-confirmed
                         Registrant information
rgpc_no                        NUMBER(16)      MANDATORY
 Registrant unique identifier.
```

Request response additional parameters:

```
rgpc_name                      VARCHAR2(200)
  Name of the Registrant.

rgpc_registrant                VARCHAR2 (3)
  Registrant status.
    Valid values:
     AAP   Awaiting Registrant approval of CIRA agreement
     ACT   Registered Registrant


rgpc_master_rant               NUMBER (16)
  Master registrant unique identifier.  A master
registrant is used to identify registrants that are
registered more than once. It represents the registrant
unique identifier of its first registration. The value
returned is NULL if the function is called with
transac_type = GET_RANT_INFO_NC

rgpc_legal_type                VARCHAR2(3)
 Registrant legal type.

Valid values:
CCO   Corporation (Canada or Canadian province or
      territory)
CCT   Canadian citizen
RES   Permanent Resident of Canada
GOV   Government or government entity in Canada
EDU   Canadian Educational Institution
ASS   Canadian Unincorporated Association
HOP   Canadian Hospital
PRT   Partnership Registered in Canada
TDM   Trade-mark registered in Canada (by a non-Canadian
      owner)
TRD   Canadian Trade Union
PLT   Canadian Political Party
LAM   Canadian Library, Archive or Museum
TRS   Trust established in Canada
ABO   Aboriginal Peoples (individuals or groups)indigenous
      to Canada
INB   Indian Band recognized by the Indian Act of Canada
LGR   Legal Representative of a Canadian Citizen or
      Permanent Resident
OMK   Official mark registered in Canada
MAJ   Her Majesty the Queen
      rgpc_member                    VARCHAR2(1)
  Indicates if the Registrant is a member of CIRA.
  Valid values:
    Y   Yes
    N   No

rgpc_description               VARCHAR2(2000)
 Description of the Registrant.

rgpc_ref_info                  VARCHAR2(2000)
 Reference information, free form text for registrar use

rgpc_no_rgar                   NUMBER(16)
 Registrar of record unique identifier
```

GETTING DOMAIN NAME REGISTRATION INFORMATION

This procedure is used to retrieve registration information on a domain name. It can be used to retrieve current registered information.

Socket-based web interface details

Registrars can submit their socket-based information request to:

https://cira2.cira.ca/cira/rinfo

Request parameters:

```
user                 NUMBER(16)      MANDATORY
  Registrar user id

pwd                  VARCHAR2(30)    MANDATORY
```

```
  Registrar password

XML                    VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface

transac_type           VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    GET_DOM_INFO       Get Registration information

rgpc_no                 NUMBER(16)     MANDATORY
 Domain Name unique identifier.
```

GET_DOM_INFO using tokens:

To use a token interface you must first set the token, then call the same function by specifying token=parameter instead of user/pwd.

Rules:

10. A token expires 24 hours from the moment it is set.
11. It can only be used from the ip that set it.
12. A registrar can set no more than 10 live tokens at a time.

Using the token interface involves two steps:

1) Set token

Tokens can be set at https://cira2.cira.ca/cira/rinfo

With:

```
Request parameters:
set_token                         VARCHAR2(1)     MANDATORY
  Valid value
    Y
user                              NUMBER(16)       MANDATORY
  Registrar user id

pwd                               VARCHAR2(30)     MANDATORY
  Registrar password

XML                               VARCHAR2(10)     MANDATORY
  Valid value
```

```
    Y   Response in XML format. Mandatory for socket-based
        web interface
transac_type                 VARCHAR2(10)    MANDATORY
 Transaction type
```

For eg. by using this sample call:

```
https://cira2.cira.ca/cira/rinfo?XML=Y&user=???&pwd=??&set
_token=y&transac_type=GET_DOM_INFO
```

You will set one token:

Reply:

```
<result>
<status>ok</status>
<token>3150289.6220443.venus.64.230.122.119.22545956.1525.
2320780</token>
</result>
```

2) Use token

To use token set call to http://cira2.cira.ca/cira/rinfo

```
Request parameters:
token                        VARCHAR2(60)
MANDATORY
  Registrar token

XML                 VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface

transac_type        VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    GET_DOM_INFO       Get Registration information

rgpc_no              NUMBER(16)     MANDATORY
 Domain Name unique identifier.
```

For example: GET_DOM_LIST call over token just set will be:

http://cira2.cira.ca/cira/rinfo?XML=Y&token=3150295.31780098.venus.64
.230.122.119.22545962.1526.2320726&transac_type=GET_DOM_LIST&rgpc_no=1

XML format details

An information request will generate the following return codes in XML:

```
<result>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If any problem is encountered while completing the request, one or several error code and description will be returned. A failed request will generate a response in the following format:

```
<result>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
 <input_parameter_name>(VALUE)</input_parameter_name>
    ......
<input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an accepted information request will be in the following format:

```
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
     errors:
     (return code number): (return code text, optional)
     end of error
```

```
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request response additional parameters:

```
rgpc_subdomain        VARCHAR2(50)
  Domain name.

rgpc_registration      VARCHAR2(3)
 Indicates domain name registration status.
  Valid values:
     AAP   Awaiting Registrant approval of CIRA agreement
     ACT   Active domain name
     CNF   Active domain name being in process of
           annual confirmation.
     MAN   Domain name awaiting CIRA approval
     SUS   Suspended domain name
     ANR   Active domain name, will not be renewed

rgpc_trade_mark_registered     VARCHAR2(1)
 Indicates if the domain name is a trademark.
  Valid values:
    Y   Yes
    N   No

rgpc_adr_process               VARCHAR2(1)
  Indicates if the domain name in an ADR process.

rgpc_date_received             DATE
  Registration arrival date.

rgpc_date_approved             DATE
  Registration approval date.

rgpc_date_rejected             DATE
  Registration rejection date

rgpc_date_suspended            DATE
  Registration suspension date
```

```
rgpc_date_lapsed               DATE
  Registration deactivation date

rgpc_date_last_renew           DATE
  Registration next renewal date

rgpc_anniv_date                DATE
  Registration next anniversary date

rgpc_no_rant                   NUMBER(16)
  Registrant unique identifier

rgpc_no_rgar                   NUMBER(16)
 Registrar unique identifier
nmsv_1_name                    VARCHAR2(200)
 Name of the first name server

nmsv_1_ip_address              VARCHAR2(100)
 IP address of the first name server

nmsv_1_ip_addr_type            VARCHAR2(4)
  Address type for the first name server IP address, only
IPV4 is currently supported.

nmsv_2_name                    VARCHAR2(200)
 Name of the second name server

nmsv_2_ip_address              VARCHAR2(100)
 IP address of the second name server

nmsv_2_ip_addr_type            VARCHAR2(4)
  Address type for the second IP address, only IPV4 is
currently supported.

nmsv_3_name                    VARCHAR2(200)
 Name of the third name server

nmsv_3_ip_address              VARCHAR2(100)
 IP address of the third name server

nmsv_3_ip_addr_type            VARCHAR2(4)
  Address type for the third name server IP address, only
IPV4 is currently supported.
```

```
nmsv_4_name                    VARCHAR2(200)
 Name of a fourth name server

nmsv_4_ip_address              VARCHAR2(100)
 IP address of the fourth name server

nmsv_4_ip_addr_type            VARCHAR2(4)
  Address type for the fourth name server IP address, only
IPV4 is currently supported.

nmsv_5_name                    VARCHAR2(200)
 Name of the fifth name server
nmsv_5_ip_address              VARCHAR2(100)  OPTIONAL
 IP address of the fifth name server

nmsv_5_ip_addr_type     VARCHAR2(4)    OPTIONAL
  Address type for the fifth name server IP address, only
IPV4 is currently supported.

nmsv_6_name          VARCHAR2(200)  OPTIONAL
 Name of a sixth name server
nmsv_6_ip_address   VARCHAR2(100)  OPTIONAL
 IP address of the sixth name server

nmsv_6_ip_addr_type  VARCHAR2(4)    OPTIONAL
  Address type for the sixth name server IP address, only
IPV4 is currently supported.
```

REGISTRANT CONTACT LIST

This procedure returns a list of all contacts for a specific Registrant.

Socket-based web interface details

Registrars can submit their contact list request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A contact list request will generate the following return codes in XML:

```
<result>
```

```
   <no>(number of contact found)</no>
   <ctc>
      <ctc_id>(contact ID)</ ctc_id>
      <ctc_name>(contact name)</ctc_name>
      <crpc_type>(contact type)</crpc_type>
      <crpc_main_rgpc>(contact main rgpc)</crpc_main_rgpc>
   </ctc>
     ....
   <ctc_id>(contact ID)</ ctc_id>
      <ctc_id>(contact ID)</ ctc_id>
      <ctc_name>(contact name)</ctc_name>
      <crpc_type>(contact type)</crpc_type>
      <crpc_main_rgpc>(contact main 
registrant)</crpc_main_rgpc>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If the request can’t be processed, the system will generate the following return codes in XML:

```
<result>
   <no>0</no>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for a contact list request will be in the following format:

```
no: (number of contact found)
ctc_id_1='(contact ID)'
ctc_name_1='(contact type)'
crpc _type_1='(domain status)'
```

```
crpc_main_rgpc_1='(contact main registrant)'
ctc_id_2='(contact ID)'
ctc_name_2='(contact type)'
crpc_type_2='(domain status)'
crpc_main_rgpc_2='(contact main registrant)'
....
ctc_id_N='(contact ID)'
ctc_name_N='(contact type)'
crpc_type_N='(domain status)'
crpc_main_rgpc_N='(contact main registrant)'
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

If the system can't process the request, the body of the response email will be in the following format:

```
no: 0
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Request parameters:

```
user                           NUMBER(16)       MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)     MANDATORY
  Registrar password

XML                            VARCHAR2(10)     MANDATORY
  Valid value
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface

transac_type                   VARCHAR2(10)     MANDATORY
 Transaction type
  Valid values
```

```
    GET_CTC_LIST

rgpc_no                         NUMBER(16)      MANDATORY
 Registrant unique identifier.
```

Request response additional parameters:

```
ctc_id                          NUMBER(16)
  Contact unique identifier.

ctc_name                        VARCHAR2(300)
  Name of contact.

crpc_type                        VARCHAR2(3)
  Contact role
  Valid values:
    ADM    Administrative
    TEC    Technical

crpc_main_rgpc                   VARCHAR2(1)
  A contact can serve more than one Registrant.  This is
often the case for a technical contact supporting several
Registrants.  However, the CIRA system requires each
contact to be “owned” by only one Registrant. If the
contact “belongs” to the Registrar, the parameter contains
‘Y’, otherwise it contains ‘N’.
```

CONTACT INFORMATION

This procedure retrieves information on a contact. It can be use to retrieve current registered information or new and unconfirmed information.

Socket-based web interface details

Registrars can submit their socket-based information request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A request for contact information will generate the following return codes in XML:

```
<result>
```

```
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed request will generate a response in the following format:

```
<result>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</ input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</ input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a request for contact information will be in the following format:

```
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for Socket-based
Web
        interface
    N   Other interface

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    GET_CTC_INFO        Get contact information
    GET_CTC_INFO_NC     Get modified and non-confirmed
                        contact information

rgpc_no                        NUMBER(16)      MANDATORY
 Registrant unique identifier.

ctc_id                         NUMBER(16)      MANDATORY
  Contact unique identifier.
```

Request response additional parameters:

```
ctc_type                       VARCHAR2(3)
  Type of contact
  Valid values:
    ADM    Administrative
    TEC    Technical

ctc_language                   VARCHAR2(3)
 Preferred language of contact.
  Valid values:
    EN    English
    FR    French
ctc_last_name                  VARCHAR2(200)
 Last name of contact.
```

```
ctc_first_name                 VARCHAR2(50)
 First name of contact.

ctc_midle_name                 VARCHAR2(50)
 Middle name of contact.

ctc_title                      VARCHAR2(3)
 Name title .
  Valid values:
    MR    Mr
    MRS   Mrs
    MS    Ms
    DR    Dr

ctc_job_title                  VARCHAR2(200)
 Job title of contact.

ctc_nationality                VARCHAR2(3)
 Nationality of contact.
  Valid values:
    CND     Canadian
    OTH     Foreign citizenship
    RES      Canadian resident

ctc_company                    VARCHAR2(200)
  Company of contact.

ctc_phone                      VARCHAR2(40)
  Phone number of contact.

ctc_fax                        VARCHAR2(40)
  Fax number of contact.

ctc_mobile                     VARCHAR2(40)
  Mobile phone number of contact.

ctc_email                      VARCHAR2(100)
  Email address of contact.

ctc_second_email               VARCHAR2(100)
  Second email address of contact.

ctc_other_phone                VARCHAR2(40)
  Other phone number of contact. (Home, etc..)
```

```
ctc_access_key                 VARCHAR2(2048)
  PGP access key for encrypted message from CIRA.

addr_address_number            VARCHAR2(200)
  Civic number of contact address.

addr_corporative               VARCHAR2(200)
  Additional delivery information. For example:  "Acme
Tower", "MIS Department"

addr_street_cat                VARCHAR2(3)
  Street type of contact address
  Valid values:
    AV     Avenue
    BL     Boulevard
    CT     Court
    CR     Crescent
    DR     Drive
    HW     Highway
    PL     Place
    PZ     Plaza
    RD     Road
    SQ     Square
    ST     Street
    WY     Way

addr_street_name               VARCHAR2(200)
  Street name of contact address.

addr_orientation               VARCHAR2(3)
  Street orientation of contact address
  Valid values:
    E      East
    N      North
    NE     North-East
    NW     North-West
    S      South
    SE     South-East
    SW     South-West
    W      West

addr_office                    VARCHAR2(20)
  Office number of contact address (Suite 340)

addr_city                      VARCHAR2(200)
```

```
  City of contact address.
addr_province_state            VARCHAR2(50)
  Province or territory of contact address,
  Valid values for Canada:
    AB      Alberta
    BC      British Columbia
    MB      Manitoba
    NB      New Brunswick
    NF      Newfoundland
    NS      Nova Scotia
    NT      Northwest Territories
    NU      Nunavut
    ON      Ontario
    PE      Prince Edward Island
    QC      Quebec
    SK      Saskatchewan
    YK      Yukon

addr_postal_code               VARCHAR2(20)
  Postal code of contact address.

addr_country                   VARCHAR2(200)
  Country of contact address. Valid values are ISO 3166
3-letter country codes. See annex B for the list of valid
values.
```

.CA NAME SERVER VALIDATION

This utility enables registrars to query the CIRA database and get information on the validity of a .ca name server. The validity of the .ca name server will be returned in the 'is_valid' parameter and will have one of the following values:

Y Domain portion of .ca Name Server is registered

N Domain portion of .ca Name Server is not registered.

This validation is intended for use with .ca name servers only. Any other types will return an error.

Socket-based Web interface details

Registrars can submit their socket-based WHOIS request to:
http://cira2.cira.ca/cira/registraires

XML format details

A WHOIS request will generate the following return codes via XML:

```
<result>
  <status>(status value)</status>
  <input_parameter_name>(VALUE)</input_parameter_name>
     ......
  <input_parameter_name>(VALUE)</input_parameter_name>
<output_parameter_name>(VALUE)</output_parameter_name>
    ......
<output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If the .ca name server is not a valid .ca domain, the response will be in the following format:

```
<result>
   <status>(status value)</status>
   <error>
      <code>( ERROR CODE)</code>
      <description>( ERROR DESCRIPTION)</description>
   </error>
<input_parameter_name>(VALUE)</input_parameter_name>
   ......
<input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

One of several error codes and descriptions will explain why the .ca name server is not valid. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for a successful request will be in the following format:

```
status: (status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

If the .ca name server is not valid, the response will be in the following format:

```
status: RJC
errors:
(return code number): (return code text, optional)
end of error
```

```
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will explain why the domain is not available. In every case, all the parameters transmitted in the originating request will be returned.

Request parameters:

```
User    NUMBER(16)MANDATORY
Registrar user id

pwdVARCHAR2(30)           MANADTORY
Registrar password

XML                  VARCHAR2(10)          MANDATORY
 Valid value:
  Y Response in XML format.                MANDATORY
  N Other format


transac_type         VARCHAR2(10)          MANDATORY
  Transaction type
  Valid values: CHECK_NMSV

NMSV              VARCHAR2(50)        MANDATORY
  Name server.
```

Request response additional parameters:

```
is_valid                     VARCHAR2(1)
  Status of domain portion of .ca name server
```

CHAPTER 7 REGISTRATION CHANGES, UPDATES, AND MODIFICATIONS

CRITICAL CHANGES

Changes made to a Registrant's profile are considered to be critical changes. There are two categories of critical changes. The first category which includes changing the name, telephone number, title or address of either the administrative or technical contact of a Registrant is considered a standard critical change and can be done by the managing Registrar.

Other changes such as the email address of the administrative contact or a change or correction relating to the name of an organisation are considered to be highly critical changes and can only be done by CIRA. The managing Registrar must request these highly critical changes via email to ecac@cira.ca

PROCESS APPROVAL

Some processes need approval from the Registrar. For example, the Registrar must approve a domain name change for a Registrant. This procedure allows Registrars to accept or reject these processes.

Socket-based web interface details

Registrars can submit their socket-based approval request to:

https: //cira2.cira.ca/cira/registraires

XML format details

An approval request will generate the following return codes in XML:

```
<result>
```

```
   <status>OK</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed approval request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for an approval request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)     MANDATORY
Registrar user id

pwd                            VARCHAR2(30)   MANDATORY
Registrar password

XML                            VARCHAR2(10)   MANDATORY
  Valid values:
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface

transac_type                   VARCHAR2(10)   MANDATORY
  Transaction type
  Valid values:
    APPROVE_PROCESS

pris_list                       NUMBER(10)    MANDATORY
  List of process instance identification number to accept
or reject. Identification number must be separated by
comma.

If parameter 'all' contains 'Y', this parameter should
contain only one process instance identification number.
The function will apply the decision to all processes
needing approval starting with the one with the lowest
process instance identification number up to the one
specified with this parameter.

all                            VARCHAR2(1)    OPTIONAL
                                             Default 'N'
  Indicates which process instance to approve.
  Valid values:
    N   apply the decision to processes listed
        in pris_list
    Y   apply the decision to all process to be
        approved, starting with the one with the
        lowest process instance identification
        number up to the one specified in pris_list.

process_type                   VARCHAR2(3)    OPTIONAL
                                             Default 'ALL'
  Indicates the type of process to approve. This parameter
is ignored if 'all' parameter contains 'N'.
Valid values:
    ALL  Every process type
    CRT  Change of Registrant process

decision                       VARCHAR2(3)    MANDATORY
  Approval decision. Warning, the decision applies to all
processes.  If you want to reject a specific process, make
sure pris_list is filled only with the process id and
parameter 'all' equals 'N'.
  Valid values:
    OK    Process accepted
    RJC   Process rejected

 reason                        VARCHAR2(200)   OPTIONAL
  Reason for the decision.
```

UPDATING REGISTRANT INFORMATION

This procedure updates a Registrant’s general information.

If parameters rgpc_legal_type contain changed information and the Registrant has completed its registration process, then this update will be considered critical and the Registrant will be notified to confirm the changes. It is important to note the Registrant’s name can only be changed if the Registrant has not yet completed its registration process.

The result of an update request is returned in the STATUS parameter. It can have one of the following values:

```
OK:  Update request processed and accepted. A confirmation
     email will be sent to the Registrant.
CRT: Update request processed. An email will be sent to
     the Registrant to invite him to confirm the new
     information.
ERR: Update request could not be processed.  Error codes
are returned.
```

Socket-based web interface details

Registrars can submit their socket-based information update to:

https: //cira2.cira.ca/cira/registraires

XML format details

An update request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If any problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an information update request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface
transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    UPD_RANT_INFO

rgpc_no                        NUMBER(16)      MANDATORY
 Registrant unique identifier.

rgpc_name                      VARCHAR2(200)   OPTIONAL
  Name of the Registrant. This parameter is ignored if
Registrant has completed its registration process,
otherwise, the parameter is mandatory.

rgpc_legal_type                VARCHAR2(3)     MANDATORY
 Registrant legal type.

Valid values:

CCO   Corporation (Canada or Canadian province or
      territory)
CCT   Canadian citizen
RES   Permanent Resident of Canada
GOV   Government or government entity in Canada
EDU   Canadian Educational Institution
ASS   Canadian Unincorporated Association
HOP   Canadian Hospital
PRT   Partnership Registered in Canada
```

```
TDM   Trade-mark registered in Canada (by a non-Canadian
      owner)
TRD   Canadian Trade Union
PLT   Canadian Political Party
LAM   Canadian Library, Archive or Museum
TRS   Trust established in Canada
ABO   Aboriginal Peoples (individuals or groups)indigenous
      to Canada
INB   Indian Band recognized by the Indian Act of Canada
LGR   Legal Representative of a Canadian Citizen or
      Permanent Resident
OMK   Official mark registered in Canada
MAJ   Her Majesty the Queen

rgpc_member                   VARCHAR2(1)    OPTIONAL
                                             Default 'Y'
  Indicates if the Registrant wants to be a member of
CIRA.
  Valid values:
    Y   Yes
    N    No

rgpc_description              VARCHAR2(2000) OPTIONAL
 Description of the Registrant.
rgpc_ref_info                 VARCHAR2(2000) OPTIONAL
 Reference information, free form text for registrar use.
```

UPDATE CONTACT INFORMATION

This procedure is used to update contact information. If the contact is a Registrant’s administrative contact, this update will be considered critical and the Registrant will be notified to confirm the changes

The result of an update request is returned in the STATUS parameter. It can have one of the following values:

```
OK:  Update request processed and accepted. A confirmation
     email will be sent to the Registrant
CRT: Update request processed. An email will be sent to
     the Registrant to invite him to confirm the new
     information. If the email address of the
     administrative contact was changed, the email will be
     sent to both the new and old email address.
ERR: Update request could not be processed.  Error codes
```

```
are returned.
```

Note: At least one phone number must be provided for the administrative contact: ctc_phone will be published in WHOIS while ctc_other_phone will remain confidential.

Socket-based web interface details

Registrars can submit their socket-based information update to:

https: //cira2.cira.ca/cira/registraires

XML format details

An update request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an information update request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
     Y   Response in XML format. Mandatory for
         socket-based web interface
     N   Other interface

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    UPD_CTC_INFO

rgpc_no                        NUMBER(16)      MANDATORY
 Registrant unique identifier.

ctc_id                         NUMBER(16) )    MANDATORY
  Contact unique identifier.

ctc_language                   VARCHAR2(3)    MANDATORY
 Contact  preferred language.
  Valid values:
    EN    English
    FR    French

ctc_last_name                  VARCHAR2(200)  MANDATORY
 Last name of contact.
```

```
ctc_title                      VARCHAR2(3)    OPTIONAL
 Name title.
  Valid values:
    MR     Mr
    MRS    Mrs
    MS     Ms
    DR     Dr

ctc_first_name                 VARCHAR2(50)   OPTIONAL
 First name of contact. This parameter is mandatory for
administrative contact.

ctc_midle_name                 VARCHAR2(50)   OPTIONAL
 Middle name of contact.

ctc_job_title                  VARCHAR2(200)  OPTIONAL
 Job title of contact.

ctc_nationality                VARCHAR2(3)    OPTIONAL
 Nationality of contact.
  Valid values:
    CND   Canadian
    OTH   Foreign citizenship
    RES   Canadian resident

ctc_company                    VARCHAR2(200)  OPTIONAL
  Company of contact.

ctc_phone                      VARCHAR2(40)   OPTIONAL
  Phone number of contact.

ctc_fax                        VARCHAR2(40)   OPTIONAL
  Fax number of contact.

ctc_mobile                     VARCHAR2(40)   OPTIONAL
  Mobile phone number of administrative contact.

ctc_email                      VARCHAR2(100)  MANDATORY
  Email address of contact.
ctc_second_email               VARCHAR2(100)  OPTIONAL
  Second email address of contact.

ctc_other_phone                VARCHAR2(40)   OPTIONAL
  Other phone number of contact. (Home, etc..)
```

```
ctc_access_key                 VARCHAR2(2048) OPTIONAL
  PGP access key for encrypted message from CIRA.

ctc_same_addr_admin        VARCHAR2(1)        OPTIONAL
  Indicates if the contact has the same address as the
administrative contact. This parameter is ignored if the
contact is an administrative contact. If Y, the following
addr_* parameters are ignored.
  Valid values:
    Y   Yes
    N   No

addr_address_number            VARCHAR2(200)  OPTIONAL
  Civic number of contact address.

addr_corporative               VARCHAR2(200)  OPTIONAL
  Additional delivery information. For example:  "Acme
Tower", "MIS Department"

addr_street_cat                VARCHAR2(3)    OPTIONAL
  Street type of contact address
  Valid values:
    AV    Avenue
    BL    Boulevard
    CT    Court
    CR    Crescent
    DR    Drive
    HW    Highway
    PL    Place
    PZ    Plaza
    RD    Road
    SQ    Square
    ST    Street
    WY    Way

addr_street_name               VARCHAR2(200)  OPTIONAL
  Street name of contact address.

addr_orientation               VARCHAR2(3)    OPTIONAL
  Street orientation of contact address
  Valid values:
    E     East
    N     North
    NE    North-East
    NW    North-West
    S     South
    SE    South-East
    SW    South-West
    W     West

addr_office                    VARCHAR2(20)   OPTIONAL
  Office number of contact address (Suite 340)

addr_city                      VARCHAR2(200)  MANDATORY
  City of contact address
addr_province_state            VARCHAR2(50)   OPTIONAL
  Province or territory of administrative contact address,
  MANDATORY if in Canada.
  Valid values for Canada, Not validated in other
countries:
    AB     Alberta
    BC     British Columbia
    MB     Manitoba
    NB     New Brunswick
    NF     Newfoundland
    NS     Nova Scotia
    NT     Northwest Territories
    NU     Nunavut
    ON     Ontario
    PE     Prince Edward Island
    QC     Quebec
    SK     Saskatchewan
    YK     Yukon
addr_postal_code               VARCHAR2(20)   OPTIONAL
  Postal code of contact address, MANDATORY if in Canada.

addr_country                   VARCHAR2(200)  MANDATORY
  Country of contact address. Valid values are ISO 3166
3-letter country codes. See annex B for the list of valid
values.
```

CHANGING DOMAIN NAME INFORMATION

This procedure is used to update information on a domain name registration. The result of an update request is returned in the STATUS parameter. It can have one of the following values:

```
OK: Update request processed.
```

```
ERR: Update request could not be processed.  Error codes
are returned.
```

Socket-based web interface details

Registrars can submit their socket-based information update to:

https: //cira2.cira.ca/cira/registraires

XML format details

An update request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for an information update request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
```

```
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    UPD_DOM_INFO

rgpc_no                        NUMBER(16)      MANDATORY
 Domain Name unique identifier.

rgpc_trade_mark_registered     VARCHAR2(1)     OPTIONAL
 Indicates if the domain name is a trademark.
  Valid values:
    Y    Yes
    N    No

nmsv_1_name                    VARCHAR2(200)  OPTIONAL
 Name of the first name server

nmsv_1_ip_address              VARCHAR2(100)  OPTIONAL
  IP address of the first name server

nmsv_1_ip_addr_type            VARCHAR2(4)    OPTIONAL
```

```
                                              Default IPV4
 Address type for the first name server IP address, only
IPV4 is currently supported.

nmsv_2_name                    VARCHAR2(200)  OPTIONAL
 Name of a second name server

nmsv_2_ip_address              VARCHAR2(100)  OPTIONAL
 IP address of the second name server

nmsv_2_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 Address type for the second name server IP address, only
IPV4 is currently supported.

nmsv_3_name                    VARCHAR2(200)  OPTIONAL
 Name of the third name server

nmsv_3_ip_address              VARCHAR2(100)  OPTIONAL
 IP address of the third name server
nmsv_3_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 Address type for the third name server IP address, only
IPV4 is currently supported.

nmsv_4_name                    VARCHAR2(200)  OPTIONAL
 Name of a fourth name server

nmsv_4_ip_address              VARCHAR2(100)  OPTIONAL
 IP address of the fourth name server

nmsv_4_ip_addr_type            VARCHAR2(4)    OPTIONAL
                                              Default IPV4
 Address type for the fourth name server IP address, only
IPV4 is currently supported.

nmsv_5_name                    VARCHAR2(200)  OPTIONAL
 Name of the fifth name server

nmsv_5_ip_address              VARCHAR2(100)  OPTIONAL
 IP address of the fifth name server

nmsv_5_ip_addr_type            VARCHAR2(4)    OPTIONAL
Default IPV4
```

```
 Address type for the fifth name server IP address, only
IPV4 is currently supported.

nmsv_6_name                  VARCHAR2(200)  OPTIONAL
 Name of a sixth name server

nmsv_6_ip_address            VARCHAR2(100)  OPTIONAL
 IP address of the sixth name server

nmsv_6_ip_addr_type          VARCHAR2(4)    OPTIONAL
                                            Default IPV4
 Address type for the sixth name server IP address, only
IPV4 is currently supported.
```

ADDING A NEW CONTACT

This procedure adds a new contact to a registrant. The result of an update request is returned in the STATUS parameter. It can have one of the following values:

```
OK:  New contact request processed and accepted.
ERR: New contact request could not be processed.
Error codes are returned.
```

Socket-based web interface details

Registrars can submit their socket-based new contact request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A new contact request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
```

```
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a new contact request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    ADD_CTC

rgpc_no                        NUMBER(16)      MANDATORY
 Registrant unique identifier.

crpc_type                      VARCHAR2(3)      MANDATORY
  Type of contact
  Valid values:
    TEC    Technical

exist_ctc_id                   NUMBER(16)      OPTIONAL
  Indicates the ID of an already registered contact that
will play the role defined in ctc_type.  If filled, the
following ctc_* and addr_* parameters are ignored.

ctc_language                   VARCHAR2(3)     MANDATORY
 Contact  preferred language.
  Valid values:
    EN             English
    FR             French
ctc_last_name                  VARCHAR2(200)   MANDATORY
 Last name of contact.

ctc_title                      VARCHAR2(3)     OPTIONAL
 Name title.
  Valid values:
    MR    Mr
    MRS   Mrs
    MS    Ms
    DR    Dr

ctc_first_name                 VARCHAR2(50)    OPTIONAL
 First name of contact.

ctc_midle_name                 VARCHAR2(50)    OPTIONAL
 Middle name of contact.

ctc_job_title                  VARCHAR2(200)   OPTIONAL
 Job title of contact.
```

```
ctc_nationality                VARCHAR2(3)     OPTIONAL
 Nationality of contact.
  Valid values:
    CND    Canadian
    OTH    Foreign citizenship
    RES    Canadian resident

ctc_company                    VARCHAR2(200)   OPTIONAL
  Company of contact.

ctc_phone                      VARCHAR2(40)    OPTIONAL
  Phone number of contact.

ctc_fax                        VARCHAR2(40)    OPTIONAL
  Fax number of contact.

ctc_mobile                     VARCHAR2(40)    OPTIONAL
  Mobile phone number of contact.

ctc_email                      VARCHAR2(100)   MANDATORY
  Email address of contact.
ctc_second_email               VARCHAR2(100)   OPTIONAL
  Second email address of contact.

ctc_other_phone                VARCHAR2(40)    OPTIONAL
  Other phone number of contact. (Home, etc..)

ctc_access_key                 VARCHAR2(2048)  OPTIONAL
  PGP access key for encrypted message from CIRA.

addr_address_number            VARCHAR2(200)   OPTIONAL
  Civic number of contact address.

ctc_same_addr_admin        VARCHAR2(1)        OPTIONAL
  Indicates if the contact has the same address as the
administrative contact. If Y, the following addr_*
parameters are ignored.

addr_corporative               VARCHAR2(200)   OPTIONAL
  Additional delivery information. For example:  "Acme
Tower", "MIS Department"

addr_street_cat                VARCHAR2(3)     OPTIONAL
  Street type of contact address
  Valid values:
```

AV	Avenue
BL	Boulevard
CT	Court
CR	Crescent
DR	Drive
HW	Highway
PL	Place
PZ	Plaza
RD	Road
SQ	Square
ST	Street
WY	Way

addr_street_name VARCHAR2(200) OPTIONAL
Street name of contact address.

addr_orientation VARCHAR2(3) OPTIONAL
Street orientation of contact address
Valid values:

E	East
N	North
NE	North-East
NW	North-West
S	South
SE	South-East
SW	South-West
W	West

addr_office VARCHAR2(20) OPTIONAL
Office number of contact address (Suite 340)

addr_city VARCHAR2(200) MANDATORY
City of contact address.

addr_province_state VARCHAR2(50) OPTIONAL
Province or territory of administrative contact address, MANDATORY if in Canada.
Valid values for Canada, not validated in other countries:

AB	Alberta
BC	British Columbia
MB	Manitoba
NB	New Brunswick
NF	Newfoundland
NS	Nova Scotia

NT	Northwest Territories
NU	Nunavut
ON	Ontario
PE	Prince Edward Island
QC	Quebec
SK	Saskatchewan
YK	Yukon

```
addr_postal_code               VARCHAR2(20)    OPTIONAL
  Postal code of contact address, MANDATORY if in Canada.

addr_country                   VARCHAR2(200)   MANDATORY
  Country of contact address. Valid values are ISO 3166
3-letter country codes. See Annex B for the list of valid
values.
```

Request response additional parameters:

```
ctc_id                         NUMBER(16)
  Contact unique identifier.
```

DELETING A CONTACT

This procedure deletes a contact. If the contact has more than one role, only the specified role for the specified Registrant will be deleted. Administrative contacts cannot be deleted. The result of an update request is returned in the STATUS parameter. It can have one of the following values:

```
OK:  Contact deletion request processed and accepted.
ERR: Contact deletion request could not be processed.
     Error codes are returned.
```

Socket-based web interface details

Registrars can submit their socket-based contact deletion request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A deletion request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If any problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a contact deletion request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                                NUMBER(16)       MANDATORY
  Registrar user id

pwd                                 VARCHAR2(30)     MANDATORY
  Registrar password

XML                                 VARCHAR2(10)     MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface

transac_type                        VARCHAR2(10)     MANDATORY
 Transaction type
  Valid values
    DEL_CTC

rgpc_no                             NUMBER(16)       MANDATORY
 Registrant unique identifier.

ctc_id                              NUMBER(16)       MANDATORY
  Contact unique identifier.

crpc_type                            VARCHAR2(3)      MANDATORY
 Contact role
  Valid values:
    TEC    Technical
```

CANCEL APPLICANT REGISTRANT

The process of Applicant Registrant can be cancelled by the Registrar so long as the admin contact has not yet approved the registration agreement.

Socket-based web interface details

Registrars can submit their socket-based approval request to:

https: //cira2.cira.ca/cira/registraires

XML format details

An approval request will generate the following return codes in XML:

```
<result>
   <status>OK</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed approval request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned.

Classic email interface details

The body of the response email for an approval request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted some additional parameters will also be included.

Request parameters:

```
user                       NUMBER(16)     MANDATORY
Registrar user id

pwd                            VARCHAR2(30)   MANDATORY
Registrar password

XML                          VARCHAR2(10)   MANDATORY
  Valid values:
    Y  Response in XML format. Mandatory for socket-based
       web interface
    N  Other interface

transac_type               VARCHAR2(10)   MANDATORY
  Transaction type
  Valid values:
    ANNUL_T

pris_id                       NUMBER(10)    MANDATORY
 ART Process instance identification number to cancel.
```

Request response additional parameters:

```
decisionVARCHAR2(25)
 Status of request to cancel process of ART
```

Chapter 8 Mergers and Transfers

REGISTRANT MERGER

This procedure will initiate the process to merge duplicate Registrants. Registrars using this procedure are required to be the Registrar of record of at least one of the domain names to be merged.

This procedure will MERGE the second registrant with the first. All domain names and technical contacts of the second (or merged) Registrant will be transferred to the first Registrant. However, the administrative contact of the second Registrant will be deleted.

The function will fail if either Registrant has active processes such as critical information changes and domain name transfers. In cases where the request fails, the list_inv parameter will return a list of all error conditions.

Once the MERGE process is started, both Registrants will receive an email inviting them to the CIRA web site to approve the process.

The result of the function is returned in the STATUS parameter. It can have one of the following values:

```
OK : Merger request initiated.
ERR: Merger request failed.
```

The Registrar can ask the system to verify the result of a merger request by setting parameter transac_type to VERIF_INIT_MERGE_RANT. Since this transaction will report on the request without starting a merger process, the response parameter 'pris_id' will be empty.

Socket-based web interface details

Registrars can submit their socket-based Registrant merger requests to: https://cira2.cira.ca/cira/registraires

XML format details

A merger request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If any problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface
```

```
transac_id                     VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    INIT_MERGE_RANT          Merge Registrants
    VERIF_INIT_MERGE_RANT    Check a Registrants Merger

rgpc_1_no                      NUMBER(16)     MANDATORY
  Identification number of Registrant that will remain.

rgpc_2_no                      NUMBER(16)     MANDATORY
  Identification number of Registrant that will be merged
into rgpc_1_no.

reason                        VARCHAR2(200)  OPTIONAL
 Reason to merge Registrants.
```

Request response additional parameters:

```
pris_id                        NUMBER(10)
 CIRA’s transfer process instance identification number.

nb_dom                         NUMBER(10)
 Number of domains transferred.

list_dom                       VARCHAR2(32000)
 List of domains transferred. The list format is the
following:
 <rgpc_no><tab><domain name><cr>.

nb_inv                         NUMBER(10)
 Number of registrant or domain conditions that prevented
the merger

list_inv                       VARCHAR2(32000)
 Returns a list of registrant or domain conditions that
prevented the merger. The list will contain only the first
250 conditions. The list format is the following:

 <dom/rant number><tab><dom_name or space><tab><RANT or
DOM>- <reason><tab><details><cr>.
```

```
where:
RANT indicates the condition is related to a Registrant;
DOM indicates the condition is related to a domain name.

The reason field has the following format:
   <STS or ADR or process type>

  - STS indicates a conflicting status.  For example:
    a domain name that needs manual validation by CIRA.
  - ADR indicates the domain is the subject of
    a conflict and cannot be transferred or changed.
  - In other cases, there will be a 3-letter process
    type code.

..... SPD   Suspend Domain
..... CNC   Cancel Domain
..... EDA   Enable Domain
..... CFD   Confirm Domain
..... CRA   Change registrar
..... CRT   Change registrant
..... TRF   Initiate transfer of registrant
..... AAD   Align anniversary date
..... DTD   Domain term decrease
..... CAD   Change critical information
..... MRG   Registrant Merger


The details field has the following format:
<status or pris_id or space>

If the reason field has an STS code, the details
field contains the conflicting status.
If the reason field has a process type code, the
details field contains the process status.
```

TRANSFER TO AN ALTERNATE REGISTRAR

This procedure initiates the process that will transfer to an alternate Registrar :

a) a specified domain name or Registrant

b) a Registrant and its domain names

c) the domain names of a Registrant (including the Registrant if necessary) currently managed by a specified Registrar.

The process is to be initiated by the future Registrar. The result of the function is returned in the STATUS parameter. It can have one of the following values:

```
OK : Transfer request initiated.
ERR: Transfer request is incomplete and was not processed.
```

The Registrar can ask the system to verify the result of a transfer by setting parameter transac_type to VERIF_INIT_CHG_ RAR. The procedure will only report on the request without starting the transfer process.

> ***Note:*** The response parameter 'pris_id' will be empty since no transfer takes place.

Socket-based web interface details

Registrars can submit their socket-based transfer of Registrar request to: https://cira2.cira.ca/cira/registraires

XML format details

A transfer request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Request parameters:

```
user                            NUMBER(16)      MANDATORY
  Registrar user id

pwd                             VARCHAR2(30)    MANDATORY
  Registrar password

XML                             VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface
transac_id                      VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                    VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    INIT_CHG_RAR          Transfer of Registrar
    VERIF_INIT_CHG_RAR    Check a transfer of Registrar

rgpc_alt_rar                     NUMBER(16)     MANDATORY
  Identification number of alternate Registrar, the one
who will receive the Registrant or the Domain names.

rgpc_no_totrf                    NUMBER(16)     MANDATORY
Identification number of the domain name or Registrant
that is being transferred.  If the identification number
represents a domain name, only this domain name will be
transferred. To transfer more than one domain name, it
should specify the Registrant identification number.
Warning: Sometimes a domain id can refer to the registrant
id.  Make sure to use all_dom=N when only the specified
domain is to be transferred.

all_dom                         VARCHAR2(1)    OPTIONAL
  Valid values:
       Y  All domains of the Registrant
       N  Only the Domain Name or Registrant specified
          in rgpc_no_totrf
If the rgpc_no_totrf refers to the Registrant, specify
whether or not all its domain names must be transferred.
This allows transferring the Registrant or the initial
domain name of the Registrant (frequently the same id) to
another Registrar without transferring the other domain
names.
```

```
rgar_no_from                    NUMBER(16)     OPTIONAL
  If specified, indicates the identification number of the
Registrar who currently manages the domain names to be
transferred.  This parameter is used only if rgpc_no_totrf
refers to the Registrant. If rgpc_no_totrf refers to the
Registrant and all_dom = Y and rgar_no_from is null, the
Registrant and all his Domain Names will be transferred to
the alternate Registrar. If rgar_no_from is specified,
only the domains managed by the specified registrar for
that registrant will be transferred.

reason                         VARCHAR2(200)  OPTIONAL
 Reason to change Registrar.
```

Request response additional parameters:

These parameters are present only if the request status is OK.

```
nb_dom                         NUMBER(10)
 Number of domains transferred.

nb_trf                         NUMBER(10)
 Number of domains and registrant transferred.

list_trf                       VARCHAR2 (32000)
 List of domains and registrant transferred. The list
format is the following:
<rgpc_no><tab><domain or registrant name><cr>.

cost                           NUMBER(10,2)
 Cost of the transaction, including taxes, there is no
cost to transfer a registrant without domain.

pris_id                        NUMBER(10)
 CIRA's transfer process instance identification number.
```

TRANSFERRING OWNERSHIP OF DOMAIN NAMES

This procedure is used to initiate the process that will move one or several domain names from one Registrant to another one. The Registrar managing the Registrant who is the original owner of the domain names must initiate the process.

The result of the function is returned in the STATUS parameter. It can have one of the following values:

```
OK : Transfer request initiated.
ERR: Transfer request is incomplete and was not processed.
```

The Registrar can ask the system to verify the result of a transfer by setting the parameter transac_type to VERIF_INIT_CHG_RANT. This function will report on the request without starting the transfer process.

> ***Note:*** The response parameters 'pris_id' will be empty since no transfer takes place.

Socket-based web interface details

Registrars can submit their socket-based transfer of Registrant requests to: https://cira2.cira.ca/cira/registraires

XML format details

A transfer request via web sockets will generate the following return codes via XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface

transac_id                     VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.
transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    INIT_CHG_RANT          Transfer of Registrant
    VERIF_INIT_CHG_RANT    Check a transfer of Registrant

rgpc_alt_rant                   NUMBER(16)     MANDATORY
  Identification number of Registrant that will receive
the Domain Names.

rgpc_dom_totrf                  NUMBER(16)     MANDATORY
  Identification number of Domain Name or Registrant that
is being transferred.  If the identification number is
representing a Domain Name, only this Domain Name will be
transferred. To transfer more than one Domain Name, it
should specify the Registrant identification number.

rgpc_alt_rar                   NUMBER(16)      OPTIONAL
  Identification number of the Registrar who will manage
the domain and who will be charged for the transfer. If
not specified, will be the registrar of the future
registrant.

all_dom                         VARCHAR2(1)    OPTIONAL
  Valid values:
       Y  All domains of the Registrant
       N  Only the domain specified in rgpc_dom_totrf
```

```
If the rgpc_dom_totrf refers to the Registrant, specify
whether or not all his Domain Names must be transferred.
This allows transferring the initial Domain Name of the
Registrant (frequently the same id) to another Registrar
without transferring the other Domain Names.

reason                       VARCHAR2(200)  OPTIONAL
 Reason to change Registrant.
```

Request response additional parameters:

These parameters are present only if the request status is OK.

```
nb_dom                        NUMBER(10)
 Number of domains transferred.

list_trf                      VARCHAR2(32000)
 List of domains transferred. The list format is the
following:

 <rgpc_no><tab><domain name><cr>.

cost                          NUMBER(10,2)
 Cost of the transaction, including taxes.
pris_id                       NUMBER(10)
 CIRA's transfer process instance identification number.
```

CHAPTER 9 RENEWAL, PROLONGATION, CANCELLATION

RENEWAL OF DOMAIN NAME REGISTRATIONS

This procedure is used to renew or prolong an active domain name registration.

Renewal notices are sent to the Registrar and Registrant by the CIRA system 60 days, 30 days, and 7days prior to the expiration date of the registration.

A prolongation is a request to extend the length of the domain name registration. The result of the renewal request is returned in the STATUS parameter. It can have one of the following values:

```
OK : Renewal request processed and accepted.
ERR: Renewal request is incomplete and was not processed.
```

Socket-based web interface details

Registrars can submit their socket-based renewal request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A renewal request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a renewal request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y       Response in XML format. Mandatory for
            socket-based web interface
    N       Other interface

transac_id                     VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    RENEW_DOM          Registration renewal transaction

dom_no                         NUMBER(16)      MANDATORY
  Domain name registration identification number.

length_of_term                 NUMBER         OPTIONAL
 Number of extra years the domain will be registered. The
total length of the registration cannot be higher than 10
years.
  Valid values: integer between 1 and 10, default 1.
```

Request response additional parameters:

These parameters are present only if request status is OK.

```
pris_id                        NUMBER(10)
 CIRA's renewal process instance identification number.

effect_cost                    NUMBER(3.2)
  Cost to Registrar, including applicable taxes.
```

NON-RENEWAL OF DOMAIN NAME REGISTRATIONS

This procedure is used when a domain name registration will not be renewed at the end of its term.

The system will send an email to the Registrant requesting that he confirm the non-renewal request in the next 30 days. If confirmed, no more notice will be sent to the Registrant or its Registrar regarding domain name registration renewal. The result of the request is returned in the STATUS parameter. It can have one of the following values:

```
OK : Request processed.
ERR: Request is incomplete and was not processed.
```

Socket-based web interface details

Registrars can submit their socket-based non-renewal request to:

https: //cira2.cira.ca/cira/registraires

XML format details

A non-renewal request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If any problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a non-renewal request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                          NUMBER(16)      MANDATORY
  Registrar user id

pwd                           VARCHAR2(30)    MANDATORY
  Registrar password

XML                           VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface

transac_id                    VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                  VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
```

```
    DO_NOT_RENEW_DOM          Registration renewal
transaction

dom_no                         NUMBER(16)      MANDATORY
  Domain name registration identification number.

reason                         VARCHAR2(200)   OPTIONAL
 Reason to not renew the domain name registration.
```

Request response additional parameters:

These parameters are present only if request status is OK.

```
pris_id                        NUMBER(10)
 CIRA's non-renewal request process instance
identification number.
```

BATCH RENEWAL

This procedure is used to renew or prolong more than one active domain name registration for the same registrant.

The use of this function requires:

- all domain names must be active;
- all domain names must belong to the same registrant;
- all domain names must be managed by the same registrar.

The registrar managing the domains must make the request. If one of the domains does not meet these requirements, the function will abort and no renewals will take place. The error descriptions will include the reason for failure and the list of invalid domain names.

While this is a batch renewal process, each domain name is individually renewed, so there is a process instance and a transaction for each domain name registration. However, only a single email will be sent to the registrant. The number of domains being renewed at the same time cannot exceed 500.

The result of the renewal request is returned in the STATUS parameter. It can have one of the following values:

```
OK : Renewal request processed and accepted.
ERR: Renewal request is incomplete and was not processed.
```

Socket-based web interface details

Registrars can submit their socket-based renewal requests to:

https: //cira2.cira.ca/cira/registraires

XML format details

A renewal request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a renewal request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
```

```
(output parameter-name)='(output parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)      MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)    MANDATORY
  Registrar password

XML                            VARCHAR2(10)    MANDATORY
  Valid value
    Y       Response in XML format. Mandatory for
            socket-based web interface
    N       Other interface

transac_id                     VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(10)    MANDATORY
 Transaction type
  Valid values
    RENEW_DOM_LIST       Bulk registration renewal
transaction

dom_list                    VARCHAR2(10000)  MANDATORY
  List of domain name registration identification numbers
to renew. All domain names must belong to the same
registrant and be managed by the requesting registrar.
The format of the list is:
  <dom_no>,<dom_no>,<dom_no>
```

```
For example:
   5894,59876,54876

length_of_term                 NUMBER         OPTIONAL
 Number of extra years the domain will be registered. The
total length of the registration cannot be higher than 10
years.
  Valid values: integer between 1 and 10, default 1.
```

Request response additional parameters:

These parameters are present only if the request status is OK.

```
pris_id                    NUMBER(10)
 Batch renewal process instance identification number.

pris_id_list                VARCHAR2(20000)
 List of individual renewal process instance
identification number. The format of the list is:
  <dom_no>:<pris_id>,<dom_no>:<pris_id>,<dom_no><pris_id>

For example:
   5894:15648,59876:15649,54876:15650

effect_cost                 NUMBER(3.2)
  Cost to Registrar, including applicable taxes.
```

BATCH RENEWAL WITH POSSIBLE EXCLUSIONS

This procedure is used to renew or prolong multiple active domain name registrations for the same registrant. The registrar managing the domains must make the request. By default, this function will renew all domains managed by the Registrar.

The difference between this procedure and the Batch renewal procedure is that the registrar can choose to exclude some of the registrants’ domain name registrations from the renewal request. Domain names that are flagged as not being renewed (status = ANR) won’t be included.

If one domain cannot be renewed, the function will abort and no renewal will take place. The error descriptions will include the reason for the failure and the list of invalid domain names.

While this is a batch renewal process, each domain name is individually renewed, so there is a process instance and a transaction for each domain name registration.

However, only a single email will be sent to the registrant. The number of domains being renewed at the same time cannot exceed 500.

The result of the renewal request is returned in the STATUS parameter. It can have one of the following values:

```
OK : Renewal request processed and accepted.
ERR: Renewal request is incomplete and was not processed.
```

Socket-based web interface details

Registrars can submit their socket-based renewal request to:
https://cira2.cira.ca/cira/registraires

XML format details

A renewal request will generate the following return codes in XML:

```
<result>
   <status>(status code)</status>
   <input_parameter_name>(VALUE)</input_parameter_name>
     ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
     ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed update request will generate a response in the following format:

```
<result>
   <status>ERR</status>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a renewal request will be in the following format:

```
status:(status code)
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

The body of the response email for a rejected request will be in the following format:

```
status:ERR
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will indicate a rejected request. In every case, all the parameters transmitted in the originating request will be returned. If the request is accepted, some additional parameters will also be included.

Request parameters:

```
user                           NUMBER(16)       MANDATORY
  Registrar user id

pwd                            VARCHAR2(30)     MANDATORY
  Registrar password

XML                            VARCHAR2(10)     MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for socket-based
        web interface
    N   Other interface

transac_id                     VARCHAR2(50)     OPTIONAL
  Registrar private transaction identifier.

transac_type                   VARCHAR2(10)     MANDATORY
 Transaction type
  Valid values
    RENEW_DOM_RANT      registrant registration renewal
```

```
                        transaction

rant_no                        NUMBER(16)      MANDATORY
  Registrant identification number.

not_renew_dom_list             VARCHAR2(10000)  OPTIONAL
  List of domain name registration identification number
to NOT renew. All domain names must belong to the
registrant.  The format of the list is:
  <dom_no>,<dom_no>,<dom_no>
For example
   5834,59576,54376

length_of_term                 NUMBER         OPTIONAL
 Number of extra years the domain will be registered. The
total length of the registration cannot be higher than 10
years.
  Valid values: integers between 1 and 10, default 1.
```

Request response additional parameters:

These parameters are present only if the request status is OK.

```
pris_id                   NUMBER(10)
 Batch renewal process instance identification number.

pris_id_list               VARCHAR2(20000)
 List of individual renewal process instance
identification number. The format of the list is:
  <dom_no>:<pris_id>,<dom_no>:<pris_id>,<dom_no><pris_id>
For example
   5894:15648,59876:15649,54876:15650

effect_cost                NUMBER(3.2)
  Cost to Registrar, including applicable taxes.
```

CHAPTER 10 REGISTRAR REPORTS

This chapter deals with Registrar reports. All reports are now available through all three interfaces. However, email-based reports can only be in XML format.

Socket-based web interface details

Registrars can submit their socket-based requests to:
https://cira2.cira.ca/cira/registraires

XML format details

A request will generate the following return codes in XML:

```
<result>
   <input_parameter_name>(VALUE)</input_parameter_name>
    ......
   <input_parameter_name>(VALUE)</input_parameter_name>
   <output_parameter_name>(VALUE)</output_parameter_name>
    ......
   <output_parameter_name>(VALUE)</output_parameter_name>
</result>
```

If any problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed request will generate a response in the following format:

```
<result>
   <error>
      <code>(ERROR CODE)</code>
      <description>(ERROR DESCRIPTION)</description>
   </error>
```

```
 <input_parameter_name>(VALUE)</input_parameter_name>
    ......
 <input_parameter_name>(VALUE)</input_parameter_name>
</result>
```

Classic email interface details

The body of the response email for a successful request will be in the following format:

```
 (input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
(output parameter-name)='(output parameter-value)'
....
(output parameter-name)='(output parameter-value)'
```

If a problem is encountered while completing the request, one or several error codes and descriptions will be returned. A failed request will generate a response in the following format:

```
errors:
(return code number): (return code text, optional)
end of error
(input parameter-name)='(input parameter-value)'
....
(input parameter-name)='(input parameter-value)'
```

One of several error codes and descriptions will explain why the request failed. In every case, all the parameters transmitted in the originating request will be returned.

PROCESS ACTIVITY

All the transactions done on a domain or a registrant are tracked in the system via processes. The following report gives information on these processes. If the process id is specified, the other parameters (date, registrant, etc.) will not be used. If no parameters or not enough parameters are specified, the request will be rejected in order to avoid time consuming reports.

The report format looks as follows:

```
Summary line:
-process id
-process type description
-process status: ACT Active
                 APP Approved
                 CAN Cancelled
                 WRT Waiting for registrant
                 WAR Waiting for alternate registrar
                 acknowledgment
                 WP1 Waiting for actual registrant
                 initial acknowledgment

                 WP2 Waiting for future registrant
                 acknowledgment
                 WP3 Waiting for actual registrant
                 final acknowledgment
                 FIN Finished
-user who initiated the request, registrant or registrar
 ID (rgpc_no)
-starting date
-ending date
-expiration date

Detail line :
involved parties i.e. registrars, registrants and domains
involved.
Emails and accessed web pages i.e. all the steps/approvals
that have been completed.
```

Request parameters:

```
user              NUMBER(16)      MANDATORY
  Registrar user id

pwd               VARCHAR2(30)    MANDATORY
  Registrar password

XML                VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for
        socket-based web interface
    N   Other interface

transac_id          VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type        VARCHAR2(20)    MANDATORY
 Transaction type
  Valid values
    RAR_REPORT_ACT             Process Activity
```

```
rgpc_no              NUMBER(16)     OPTIONAL
  Identification number of the Registrar involved in
the processes to be listed.

date_from            DATE           OPTIONAL
Initial date in the format YYYY/MM/DD of the interval
in between the process was active.

date_to              DATE           OPTIONAL
  Latest date in the format YYYY/MM/DD of the
interval in between the process was active.
rgpc_disp            NUMBER(16)     OPTIONAL
  Identification number of the Registrant or the
domain to be queried.

proc_type            VARCHAR2(3)    OPTIONAL
  Process type
  Valid values:

     ..... ART   Registration (domain or Registrant)
     ..... SPD   Suspend Domain
     ..... CNC   Cancel Domain
     ..... EDA   Enable domain
     ..... CFD   Confirm Domain
     ..... CRA   Change registrar
     ..... CRT   Change registrant
     ..... RND   Domain registration renewal
     ..... BRN   Bulk domain registration renewal
     ..... NRN   Request to not renew a domain
     ..... TRF   Initiate transfer of registrant
     ..... RND   Renew Domain
     ..... DTD   Domain term decrease
     ..... CAD   Change critical information
     ..... MRG   Registrant Merger
     ..... POL   Poll
     ..... PWD   Change user password
     ..... CFM   Change request confirmation
     ..... LOW   Registrar’s account balance is low
     ..... N07   7-day renewal notification
     ..... N30   30-day renewal notification
     ..... N60   60-day renewal notification


pris_id                        NUMBER(10)   OPTIONAL
```

```
 CIRA’s process instance identification number,
referred to in the email as the reference number.

summary                        VARCHAR2(1)  OPTIONAL
  Valid values:
       Y  only the first line is required
       N  the detail must be displayed

lang_rep.                      VARCHAR2(2)  OPTIONAL
 Language in which the report will be printed
  Valid values:
    EN    English
    FR    French
```

Request response additional parameters:

These parameters are present only if the request status is OK.

```
data                           long character string
 Content of the report
```

GL TRANSACTIONS

The system keeps the accounting transactions at detail level (process, fee type, GL account). The GL transaction report allows the Registrar to track the fees and the amounts debited or credited during the processing of his registrations. The report format looks as follows:

- process id (if related to a process – deposits are not)

- description (process type and domain or registrant) (for more information on a process, run the activity report)

- GL account description

- date of the transaction

- amount

If no particular process is specified, a date range must be specified.

Request parameters:

```
user                 NUMBER(16)      MANDATORY
  Registrar user id

pwd                  VARCHAR2(30)    MANDATORY
  Registrar password
```

```
XML                   VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for
        socket-based web interface
    N   Other interface

transac_id             VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type            VARCHAR2(20)    MANDATORY
 Transaction type
  Valid values
    GL_RAR                         GL Transactions

rgpc_no                NUMBER(16)     OPTIONAL
  Identification number of the Registrar involved in the
processes to be listed.

date_from             DATE           OPTIONAL
  Initial date in the format YYYY/MM/DD of the interval in
between the process was active.

date_to               DATE           OPTIONAL
  Latest date in the format YYYY/MM/DD of the interval in
between the process was active.

pris_id          NUMBER(10)     OPTIONAL
 CIRA’s process instance identification number, referred
in the email as the reference number.

lang_rep.         VARCHAR2(2)  OPTIONAL
 Language in which the report will be printed
  Valid values:
    EN    English
    FR    French
```

Request response additional parameters:

These parameters are present only if request status is OK.

```
data                           long character string
 Contents of the report
```

DOMAIN NAMES LIST

This is a summary of the domain names list. If registrars own registrants without domain names, they will be listed at the end. The report format looks as follows:

```
- rgpc_no (i.e. domain name number or registrant number
  in case of registrant without domain)
- domain name
- domain name status (registrant status in case of
  registrant without domain name)
- registrant number
- registrant name
- user/password (for applicant only)
- date of renewal
```

Request parameters:

```
user               NUMBER(16)      MANDATORY
  Registrar user id

pwd                VARCHAR2(30)    MANDATORY
  Registrar password

XML                VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for
        socket-based web interface
    N   Other interface

transac_id         VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type        VARCHAR2(20)    MANDATORY
 Transaction type
  Valid values
    RAR_REPORT_DOM             List of domains

rgpc_no             NUMBER(16)     OPTIONAL
  Identification number of the Registrar involved in the
processes to be listed.

lang_rep.            VARCHAR2(2)  OPTIONAL
 Language in which the report will be printed
  Valid values:
    EN    English
    FR    French
```

Request response additional parameters:

These parameters are present only if request status is OK.

```
data                           long character string
 Contents of the report
```

DOMAIN NAME REGISTRATIONS TO BE RENEWED

This report lists domain name registrations that need to be renewed. The report format looks as follow:

```
 -  domain name number
 -  domain name
 -  registrant number
 -  registrant name
 -  date of renewal
 -  number of days to date of renewal
```

The function will return a list of registrations to be renewed for all of the Registrar's Registrants. However, the Registrar can request a list for a single Registrant by specifying its Registrant number in parameter rant_no.

Parameter 'date_range' lets the Registrar define which registrations to include in the report based on the renewal date. Another method is to define the selection period by using date_from and date_to parameters. Please note that a selection period cannot be more than 30 days.

Request parameters:

```
user              NUMBER(16)      MANDATORY
  Registrar user id

pwd                  VARCHAR2(30)    MANDATORY
  Registrar password

XML               VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for
        socket-based web interface
    N   Other interface
transac_id         VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.

transac_type        VARCHAR2(20)    MANDATORY
```

Transaction type

Valid values

RAR_REPORT_DOM_RENEWAL	List of domains to be renewed

rar_no NUMBER(16) OPTIONAL

Identification number of the Registrar managing the domains to be listed.

lang_rep. VARCHAR2(2) OPTIONAL

Language in which the report will be printed

Valid values:

EN	English
FR	French

rant_no NUMBER(16) OPTIONAL

Identification number of Registrant; if used, the list will be limited to registrations belonging to this registrant. To get a list for all registrants, ignore this parameter.

date_range VARCHAR(20) OPTIONAL

Indicates registrations to be included in the report based on the renewal date.

Valid values

7DAY	Renewal date in 7 days
30DAYS	Renewal date in 30 days
60DAYS	Renewal date in 60 days
90DAYS	Renewal date in 90 days
7DAYSORLESS	Renewal date in 7 days or less
30DAYSORLESS	Renewal date in 30 days or less
30TO60DAYS	Renewal date between 30 to 60 days
60TO90DAYS	Renewal date between 60 to 90 days
90TO120DAYS	Renewal date between 90 to 120 days

date_from DATE OPTIONAL

Beginning date of selection time interval. Registrations having their renewal date in the selection period will be included in the report. Date is in YYYY/MM/DD format. This parameter is ignored if parameter date_range is filled.

date_to DATE OPTIONAL

```
  End date of selection time interval.  Registrations
having their renewal date in the selection period will be
included in the report. Date is in YYYY/MM/DD format.
This parameter is ignored if parameter date_range is
filled.
```

Request response additional parameters:

These parameters are present only if request status is OK.

```
data                           long character string
 Contents of the report
```

REGISTRAR ACCOUNT BALANCE VERIFICATION

This transaction will allow a Registrar to verify the balance of his CIRA account.

Socket-based Web forms interface details

Registrars can submit their socket-based requests to:

https://cira2.cira.ca/cira/registraires

Example:

https://cira2.cira.ca/cira/registraires?transac_type=BALANCE&user=22&pwd=demoen532&XML=Y

XML format details

A request will generate the following return codes in XML:

```
<nom_registraire>(VALUE)</nom_registraire>
<rgpc_balance>(VALUE)</rgpc_balance>
<transac_type>BALANCE</transac_type>
```

A failed request will generate a response in the following format:

```
<result>
<error>
<code>01003</code>
```

```
<description>Non valid account/Compte
invalide</description>
</error>
</result>
```

Request Parameters:

```
user                          NUMBER(16)      MANDATORY
  Registrar user id

pwd                           VARCHAR2(30)    MANDATORY
  Registrar password

XML                           VARCHAR2(10)    MANDATORY
  Valid value
    Y       Response in XML format only. Mandatory.

transac_type                  VARCHAR2(10)    MANDATORY
  Transaction type

Valid values
     BALANCE         Registrar’s Cira account balance
```

REGISTRAR DEPOSIT ACCOUNT STATEMENT

This report provides a statement of financial activities affecting the registrar deposit account. The statement is available for selected time periods giving a beginning and closing balance for that period.

Request parameters:

```
user              NUMBER(16)      MANDATORY
  Registrar user id

pwd                  VARCHAR2(30)    MANDATORY
  Registrar password

XML               VARCHAR2(10)    MANDATORY
  Valid value
    Y   Response in XML format. Mandatory for
        socket-based web interface
    N   Other interface
transac_id         VARCHAR2(50)    OPTIONAL
  Registrar private transaction identifier.
```

```
transac_type         VARCHAR2(20)    MANDATORY
 Transaction type
  Valid values
    RAR_DEP_ACC_STMT       Deposit Account Statement

rgpc_disp                 NUMBER(16)      MANDATORY
  Identification number of the Registrar.

lang_rep.            VARCHAR2(2)     OPTIONAL
 Language in which the report will be printed
  Valid values:
    EN   English
    FR   French

type           VARCHAR(3)     MANDATORY
 Time interval type for report
 Valid values:
    WKY          Weekly
    MTH          Monthly
    YRY          Yearly

sunday         DATE           OPTIONAL
 End date of selection time interval for p_type=WKY
 Entry must be in YYYY/MM/DD and correspond to any given
 Sunday

month          VARCHAR2(3)    OPTIONAL
 Valid months of the year for p_type=MTH:
     JAN          January
     FEB          February
     MAR          March
     APR          April
     MAY          May
     JUN          June
     JUL          July
     AUG          August
     SEP          September
     OCT          October
     NOV          November
     DEC          December


year            VARCHAR(4)    OPTIONAL
 Valid year in YYYY format for p_type=YRY
```

Request response additional parameters:

These parameters are present only if request status is OK.

```
data                          long character string
 Contents of the report
```

ANNEX A TELEPHONE NUMBER VERIFICATION

Telephone number verification is done differently for North American telephone numbers and other international telephone numbers. North American telephone numbers have a 3-digit area code and 7-digit number.   The system defines a North American telelphone number as a number that requires a caller to dial 1 before the 10-digit number when called from Canada. This includes telephone numbers from countries not located in North America. The following is a list of those countries:

```
CAN - Canada
AIA - Anguilla
ATA - Antarctica
ATG - Antigua And Barbuda
BHS - Bahamas
BRB - Barbados
BMU - Bermuda
CYM - Cayman Islands
DMA - Dominica
DOM - Dominican Republic
GRD - Grenada
GUM - Guam
JAM - Jamaica
MSR - Montserrat
MNP - Northern Mariana Islands
PRI - Puerto Rico
KNA - St Kitts and Nevis
LCA - St Lucia
VCT - St Vincent and the Grenadines
TTO - Trinidad and Tobago
TCA - Turks and Caicos
USA – United States of America
```

```
    VGB - Virgin Islands (British)
    VIR - Virgin Islands (US)
    KNA - Saint Kitts and Nevis
```

The validity of the area code is not checked by CIRA's system. However, area codes and 7-digit numbers that begin by 0 or 1 are invalid and will be rejected.

Other international numbers must begin by the international prefix number 011 + the country code. This is needed to differentiate them from North American telephone numbers. Since there is no standard in the length of the country code, area code and number for these telephone numbers, no validation is done.

Telephone numbers containing letters before the 10 first digits for North American numbers or before the first 6 digits for other international numbers are also rejected. However, parentheses, comas, periods, brackets, and the plus sign are accepted. This means the following numbers are accepted:

```
 +011 (33) 54.22.65.22
 (514) 453-3454
 568 584 3546 ext. 788
+(011) 65 78.23.32432 #7
```

The following numbers are also valid since the prefix 1 is not required for North American phone numbers:

```
 +1 375.344.2453
 (375) 344-2453
```

ANNEX B COUNTRY CODES

Country codes are ISO 3166 3-letter country codes. They are as follows:

```
CAN - Canada
AFG - Afghanistan
ALB - Albania
DZA - Algeria
ASM - American Samoa
AND - Andorra
AGO - Angola
AIA - Anguilla
ATA - Antarctica
ATG - Antigua And Barbuda
ARG - Argentina
ARM - Armenia
ABW - Aruba
AUS - Australia
AUT - Austria
AZE - Azerbaijan
BHS - Bahamas
BHR - Bahrain
BGD - Bangladesh
BRB - Barbados
BLR - Belarus
BEL - Belgium
BLZ - Belize
BEN - Benin
BMU - Bermuda
BTN - Bhutan
BOL - Bolivia
BIH - Bosnia And Herzegowina
```

BWA - Botswana
BVT - Bouvet Island
BRA - Brazil
IOT - British Indian Ocean Territory
BRN - Brunei Darussalam
BGR - Bulgaria
BFA - Burkina Faso
BDI - Burundi
KHM - Cambodia
CMR - Cameroon
CPV - Cape Verde
CYM - Cayman Islands
CAF - Central African Republic
TCD - Chad
CHL - Chile
CHN - China
CXR - Christmas Island
CCK - Cocos Islands
COL - Columbia
COM - Comoros
COG - Congo
COD - Congo, Democratic Republic of
COK - Cook Islands
CRI - Costa Rica
CIV - Cote D'Ivoire
HRV - Croatia
CUB - Cuba
CYP - Cyprus
CZE - Czech Republic
DNK - Denmark
DJI - Djibouti
DMA - Dominica
DOM - Dominican Republic
TMP - East Timor
EQU - Ecuador
EGY - Egypt
SLV - El Salvador
GNQ - Equatorial Guinea
ERI - Eritrea
EST - Estonia
ETH - Ethiopia
FLK - Falkland Islands
FRO - Faroe Islands
FJI - Fiji
FIN - Finland

FRA - France
FXX - France Metropolitan
GUF - French Guiana
PYF - French Polynesia
ATF - French Southern Territories
GAB - Gabon
GMB - Gambia
GEO - Georgia
DEU - Germany
GHA - Ghana
GIB - Gibraltar
GRC - Greece
GRL - Greenland
GRD - Grenada
GLP - Guadeloupe
GUM - Guam
GTM - Guatemala
GIN - Guinea
GNB - Guinea-Bissau
GUY - Guyana
HTI - Haiti
HMD - Heard and McDonald Islands
HND - Honduras
HKG - Hong Kong
HUN - Hungary
ISL - Iceland
IND - India
IDN - Indonesia
IRN - Iran
IRQ - Iraq
IRL - Ireland
ISR - Israel
ITA - Italy
JAM - Jamaica
JAP - Japan
JOR - Jordan
KAZ - Kazakhstan
KEN - Kenya
KIR - Kiribati
PRK - Korea, Democratic People's Republic
KOR - Korea, Republic of
KWT - Kuwait
KGZ - Kyrgyzstan
LAO - Laos
LVA - Latvia

LBN	Lebanon
LSO	Lesotho
LBR	Liberia
LBY	Libya
LIE	Liechtenstein
LTU	Lithuania
LUX	Luxembourg
MAC	Macao
MKD	Macedonia
MDG	Madagascar
MWI	Malawi
MYS	Malaysia
MDV	Maldives
MLI	Mali
MLT	Malta
MHL	Marshall Islands
MTQ	Martinique
MRT	Mauritania
MUS	Mauritius
MYT	Mayotte
MEX	Mexico
FSM	Micronesia
MDA	Moldova
MCO	Monaco
MNG	Mongolia
MSR	Montserrat
MAR	Morocco
MOZ	Mozambique
MMR	Myanmar
NAM	Namibia
NRU	Nauru
NPL	Nepal
NLD	Netherlands
ANT	Netherlands Antilles
NZL	New Zealand
NCL	New Caledonia
NIC	Nicaragua
NER	Niger
NGA	Nigeria
NIU	Niue
NFK	Norfolk island
MNP	Northern Mariana Islands
NOR	Norway
OMN	Oman
PAK	Pakistan

PLW - Palau
PSE - Palestinian Territory
PAN - Panama
PNG - Papua New Guinea
PRY - Paraguay
PER - Peru
PHL - Philippines
PCN - Pitcairn
POL - Poland
PRT - Portugal
PRI - Puerto Rico
QAT - Qatar
REU - Reunion
ROM - Romania
RUS - Russian Federation
RWA - Rwanda
KNA - Saint Kitts and Nevis
LCA - Saint Lucia
VCT - Saint Vincent and the Grenadines
WSM - Samoa
SMR - San Marino
STP - Sao Tome and Principe
SAU - Saudi Arabia
SEN - Senegal
SYC - Seychelles
SLE - Sierra Leone
SGP - Singapore
SVK - Slovakia
SVN - Slovenia
SLB - Solomon Islands
SOM - Somalia
ZAF - South Africa
SGS - South Georgia And The South Sand
ESP - Spain
LKA - Sri Lanka
SHN - St. Helena
SPM - St. Pierre and Miquelon
SDN - Sudan
SUR - Suriname
SJM - Svalbard and Jan Mayen Islands
SWZ - Swaziland
SWE - Sweden
CHE - Switzerland
SYR - Syria
TWN - Taiwan

TJK - Tajikistan
TZA - Tanzania
THA - Thailand
TGO - Togo
TKL - Tokelau
TON - Tonga
TTO - Trinidad and Tobago
TUN - Tunisia
TUR - Turkey
TKM - Turkmenistan
TCA - Turks and Caicos islands
TUV - Tuvalu
UGA - Uganda
UKR - Ukrenia
ARE - United Arab Emirates
GBR - United Kingdom
USA - United States of America
URY - Uruguay
UZB - Uzbekistan
VUT - Vanuatu
VAT - Vatican City State
VEN - Venezuela
VNM - Viet Nam
VGB - Virgin Islands (British)
VIR - Virgin Islands (U.S.)
WLF - Wallis and Futuna Islands
ESH - Western Sahara
YEM - Yemen
YUG - Yugoslavia
ZMB - Zambia
ZWE - Zimbabwe

ANNEX C ERROR CODES

The following table is a list of error codes and descriptions that the CIRA system will generate to alert the Registrar of problems or errors in transactions.

Error Code	Description
01000	This user account does not exist.
01001	This user account is inactive.
01002	Access denied.
01003	Invalid user session context.
01004	Access expired, please reconnect.
01005	This contact already has a user account.
01006	When (re)connecting, user must commit.
01007	Fee type does not exist
01008	No record found.
01009	List does not exist.
01010	Form type does not exist
01011	Specify at least the name and one value.
01012	Job code does not exist.
01013	Process type does not exist.
01014	Specified Process id does not exist.
01015	Value does not exist.
01016	Interventionist type does not exist for the process.
01017	Cannot update process/communications once instantiated.
01018	Interventionist type does not exist for the communication.
01019	Specified communication does not exist.
01020	Specified standard form does not exist.
01021	Cannot update standard form once instantiated.
01022	Contact type is invalid:

Error Code	Description
01023	Address type is invalid:
01024	Street category is invalid:
01025	Street orientation is invalid:
01026	Province or state code is invalid:
01027	Country code is invalid:
01028	Language code is invalid:
01029	Contact title code is invalid:
01030	Nationality code is invalid:
01031	User has no right to modify the information.
01032	Contact does not exist
01033	Domain name does not exist.
01034	Registry status code invalid.
01035	Registrant status code invalid:
01036	Registrar status code invalid:
01037	Registration status code invalid:
01038	Legal status code is invalid:
01039	No standard form defined or current for this process/communication.
01040	Domain already exists.
01041	Administrative contact not found.
01042	Domain name is invalid.
01043	Domain name is restricted.
01044	Job already run.
01045	Initial Registrar account balance too low. Balance should be enough to cover annual certification fee and initial commitment.
01046	Invalid amount.
01047	Invalid date (format yyyy/mm/dd).
01048	GL account does not exist.
01049	Tax does not exist.
01050	You must specify DNS name.
01051	You must specify a registered person/company.
01052	You must specify the person/company name.
01053	You must specify contact's last name.
01054	You must specify contact's language.
01055	You must specify a contact.
01056	You must specify the address type.
01057	You must specify the identifier type and value.
01058	You must specify the contact's type.
01059	You must specify a valid transaction date.

Error Code	Description
01060	Interventionist type does not match with the one required by process definition.
01061	Interventionist needs to be the registry.
01062	Interventionist needs to be an applicant registrar.
01063	Interventionist needs to be an applicant registrant.
01064	Interventionist needs to be an active registrar.
01065	Interventionist needs to be an active registrant.
01066	Process instantiation is not found or is not the right type.
01067	Initiator of the process instance cannot be found.
01068	Invalid process status code.
01069	Invalid communication status code.
01070	The contact does not have a postal address of this type.
01071	The contact does not have a postal address.
01072	The process instance cannot be found.
01073	The process instance was cancelled or is finished.
01074	The process instance is still active.
01075	This operation can only be done to an inactive registrar.
01076	This registrar cannot be suspended.
01077	This registrar cannot be revoked.
01078	This operation can only be done to a registrar.
01079	Operation cancelled because of registrar lack of funds.
01080	Second Interventionist is mandatory for this process.
01081	Communication instance does not exist.
01082	Registered person/company does not exist.
01083	Registered person/company is already a registrar.
01084	Administrative contact of an active registrar cannot be deleted.
01085	Administrative contact of an active registrant or an active registration cannot be deleted.
01086	The contact already has this relationship with the registered person/company.
01087	Address of an administrative contact must be provided.
01088	Address of the contact must be provided since there is no administrative contact.
01089	This operation can only be done to registrar.
01090	This contact has no relationship with this registered person/company.
01091	Address does not exist.
01092	New administrator must first provide office civic address.
01093	Office civic address of an administrator cannot be deleted.

Error Code	Description
01094	Specified Interventionist does not exist.
01095	IP Address type code is invalid.
01096	Identifier type code is invalid:
01097	You must specify a process ID and a transaction type
01098	Transaction does not exist.
01099	Domain name registration must be active.
01100	Registrar not specified.
01101	Not a registry clerk.
01102	Registration is in alternate dispute resolution process.
01103	Registration status does not allow change.
01104	Registration currently transferred to another registrant.
01105	Registrant currently transferred to another registrar.
01106	Registration must be active for this process.
01107	Registration is already with this registrant.
01108	Registrant is already with this registrar.
01109	The alternate registrant refers to the current Registrant.
01110	The alternate registrant changes of registrant or registrar.
01111	Sub domain must be registered first at UBC.
01112	Sub domain info (admin name or email / company) does not match UBC info.
01113	2nd level must be a province or territory code (ab, bc, qc, on, etc).
01114	Domain name is reserved: names of other level exist.
01115	3rd level must be a city.
01116	Too many levels for a domain name.
01117	Registrant must be active for this process.
01118	Technical contact is missing.
01119	Administrative contact is missing.
01120	Two Name servers are required.
01121	Registrar account is too low.
01122	Registrant is under another registrant. Please contact CIRA.
01123	Domain is already active.
01124	Process already submitted.
01125	Domain name cannot be modified.
01126	Domain must be specified.
01127	User must be admin contact of the domain.
01128	Delay not allowed for this process.
01129	Specify reference or reason for the fee.
01130	Invalid variable format in the standard form.
01131	System parameter not defined.

Error Code	Description
01132	The form does not specify if it is the INItiator, 1ST or 2ND Interventionist.
01133	Table or column non-processed in formatting program.
01134	Parameters do not refer to communication instance.
01135	Invalid Parameters.
01136	PGP access key must be missing.
01137	Referral URL must be specified.
01138	Registrar type is invalid:
01139	No amount specified for the fee.
01140	GL deposit account must be defined for the registrar.
01141	Transaction list is invalid.
01142	Registrar already connected, wait until the connection is released to reconnect.
01143	User account cannot be created when at this status
01144	Invalid PGP key.
01145	Transaction type invalid or non-existent.
01146	Financial transaction not allowed in the context
01147	Transactions up to this date are already transferred.
01148	Email format invalid.
01149	Domain does not exist or is not active.
01150	Not enough money for Certification and Application fees.
01151	Seems to be a duplicate deposit.
01152	Invalid communication for STANDARD_FORMS variable.
01153	Operation not allowed for the user.
01154	Invalid question seq for STANDARD_FORMS special variable.
01155	Change of registrant proposed by current registrant
01156	Change of registrant rejected by current registrant
01157	Change of registrant requested by the registrar of the alternate registrant
01158	Change of registrant rejected by the registrar of the alternate registrant
01159	Change of registrant accepted by the alternate registrant
01160	Change of registrant rejected by the alternate registrant
01161	Change of registrant acknowledged by current registrant
01162	Final rejection of the change of registrant by current registrant
01163	Wrong Interventionist for the current step
01164	The step does not occur in the right order.
01165	This process cannot start based on the status of the domain/registrant or due to other active processes.

Error Code	Description
01166	Missing parameters.
01167	This contact does not have a user account.
01168	This function is not implemented yet.
01200	Registration must be suspended or lapsed for this process.
01201	Suspended or rejected registration cannot be suspended again.
01202	You can't edit an admin contact information under this relationship.
01204	Applicant registration cannot be suspended.
01205	Address should be in Canada.
01206	Postal code is invalid.
01207	Canadian province or territory is missing or invalid.
01208	This operation can only be done to rejected registrar.
01209	Corporate information not found.
01210	Suspended or rejected registration doesn’t need to be re-confirmed.
01211	Registration is already in re-confirmation process.
01212	Only one administrative contact is allowed.
01213	Country code is missing.
01214	Province/Territory code is missing.
01215	Postal code is missing.
01216	Registrar must be active.
01217	Registrar needs at least one technical contact.
01218	Registrant and domain name information cannot be changed before OTD.
01219	CIRA cannot be suspended or revoked.
01220	Email address is missing (Mandatory for administrative and technical contact).
01221	No new domain can be registered now.
01222	New domain name needs a registrant.
01223	Agreement acceptance must be made in the right order.
01224	Unknown agreement.
01225	Reserved.
01226	All parts of the agreement must be approved before the registration can be accepted.
01227	Invalid Transaction Report Frequency code:
01228	Invalid service type code:
01229	Domain name is already a second level domain.
01230	Invalid domain upgrade options.
01231	Reserved.

Error Code	Description
01232	This letter will be released at
01233	No more than 1 transaction every 5 seconds.
01234	The system will be available in
01235	This TBR domain name is already registered.
01236	Backend systems are temporarily unavailable.
01237	Oracle error.
01238	Oracle internal error.
01239	There is no such function in this class.
01240	Our systems are presently unavailable due to routine scheduled maintenance.
01241	Reserved.
01242	Reserved.
01243	Reserved.
01244	Reserved.
01245	Reserved.
01246	Upgraded domain name is not available.
01247	Invalid user code or password.
01248	You have no processes to confirm.
01249	This domain was not upgraded
01250	This upgraded domain does not belong to registrant
01251	There is no active registration process for this domain.
01252	Domain name is reserved.
01253	There is no active upgrade process for this domain.
01254	Upgraded domain does not exist
01255	RGPC_TRADE_MARK_REGISTERED parameter can only have the values 'Y' or 'N'.
01256	Reserved.
01257	Registrant does not exist
01258	This is not a registrant.
01259	Process starting date is missing.
01260	Not a valid registrant.
01261	It is too early to submit a transaction.
01262	Domain name is not available: names of other level exist.
01263	There is no unconfirmed modified information for this Registrant or Registration.
01264	Registrant does not exist.
01265	Information cannot be updated, Registrant is confirming the latest changes.
01266	Invalid transaction code.
01267	This update process is not for this Registrant.

Error Code	Description
01268	Invalid confirmation decision.
01269	Invalid IP address.
01270	Invalid DNS name.
01271	IP address contains superfluous 0.
01272	A request for this domain name is presently being processed. You may try again.
01273	First and second name server must exist if other name server exists
01274	Third name server must exist if fourth, fifth or sixth name server exists
01275	Fourth name server must exist if fifth or sixth name server exists
01276	Fifth name server must exist if sixth name server exists
01277	IP address mandatory for DNS server with the same name as the registered domain.
01278	Domain name is reserved while in the process of being approved.
01279	Domain name renewal date cannot be more than 10 years away.
01280	New domain name requests are not accepted for the moment.
01281	Invalid email address:
01282	Domain name is restricted to municipal government.
01283	Registration process for another domain name of this registrant does not exist.
01284	Domain does not exist.
01285	A domain name that is being checked by CIRA cannot be approved.
01286	This function can only cancel domain name that have their registration process completed
01287	It is too late to upgrade a domain previously registered at UBC.
01288	Domain name registration period cannot be decreased.
01289	You can't decrease the registration period to less than 6 months
01290	Registration period cannot be decreased by the requested number of years. The maximum possible number of years is :
01291	Registration period cannot be decreased by the requested number of years:
01292	Domain name registration cannot be cancelled with full refund.
01293	Cannot cancel with full refund a domain name that was registered or renewed more than 3 months ago.

Error Code	Description
01294	Member parameter value is invalid:
01295	Registrant name is invalid.
01296	Registrant having a status of Canadian Citizen or Permanent Resident must also be his/her own administrative contact.
01297	New domain name can't be the name of the Registrant
01298	Registrant already exists. Its Unique Registrant Identifier Code is :
01299	Invalid phone number :
01300	Invalid fax number :
01301	This is not a Master Registrant
01302	A Registrant with an active domain name registration cannot be cancelled.
01303	The domain name cannot be exactly the same as the Registrant’s name.
01304	To change the name of an individual Registrant, you must change the name of the administrative contact.
01305	Values for parameter all_dom can only be ‘Y’ or ‘N’.
01306	The administrative contact of an individual Registrant cannot be changed.
01307	Merge of Registrants has been accepted by the first Registrant.
01308	Merge of Registrants has been accepted by the annexed Registrant.
01309	Merge of Registrants has been rejected by the first Registrant.
01310	Merge of Registrants has been rejected by the annexed Registrant.
01311	The Registrant is already a master Registrant:
01312	Extension or renewal period must be at least one year.
01313	The Registrant is in the process of being merged. Domain name registrations cannot be added while the process is active.
01314	Invalid mobile phone number.
01315	You must provide at least one phone number.
01316	You must provide a fax number.
01317	You must confirm if you can comply with the minimum qualifications for CIRA Certified Registrars.
01318	You must provide your corporate number and jurisdiction.
01319	You must indicate how many years the Organization has been in existence.
01320	You must indicate the principal business of the

Error Code	Description
	Organization.
01321	You must indicate if the Organization’s gross revenue is more than $1,000,000. per year.
01322	You must provide the name of your bank.
01323	Request already submitted.
01324	Request already accepted. The domain name registration will not be renewed.
01325	End date of renewal period is missing.
01326	The system does not recognize the Registrant’s name as the name of a corporation.
01327	The system recognizes the Registrant name as the name of a corporation. Corporations must use “Canadian Corporation” as the Registrant legal type.
01328	According to the legal type selected, this Registrant must be an individual. The system does not recognize the Registrant’s name as the name of an individual.
01329	The administrative contact must be an individual. The system does not recognize this administrative contact’s name as the name of an individual.
01330	Registration period must be at least one year.
01331	The email address of an administrative contact cannot be changed. Contact CIRA for more information.
01332	You must specify a list of domain names.
01333	You must specify a registrant.
01334	The format of the list of domain names is invalid.
01335	The domain names in the list belong to more than one Registrant. These registrants are:
01336	The user does not have the right to modify the following domain name registration:
01337	The following domain name registration cannot be renewed because it is not active:
01338	This function cannot renew more than 500 domain name registrations at once.
01339	There is no domain name registration to renew.
01340	Registration is suspended.
01341	The registration cannot be renewed at this time.
01342	The domain name is not yet available.
01345	The domain of the DNS server name is not registered with CIRA.
02001	This domain name is restricted to municipal government.
02002	Her Majesty the Queen’s requests are handled by CIRA's

Error Code	Description
	staff.
02003	Administrative contact address is outside Canada.
05001	The following error(s) are related to the administrative contact:
05002	End of error(s) related to administrative contact.
05003	The following error(s) are related to the technical contact:
05004	End of error(s) related to technical contact.
05005	The following error(s) are related to first name server:
05006	End of error(s) related to first name server.
05007	The following error(s) are related to second name server:
05008	End of error(s) related to second name server.
05009	The following error(s) are related to third name server:
05010	End of error(s) related to third name server.
05011	The following error(s) are related to fourth name server:
05012	End of error(s) related to fourth name server.
05013	The following error(s) are related to fifth name server:
05014	End of error(s) related to fifth name server.
05015	The following error(s) are related to sixth name server:
05016	End of error(s) related to sixth name server.
05017	The following error(s) are related to corporate information:
05018	End of error(s) related to corporate information.
05019	The following error(s) are related to the banking contact:
05020	End of error(s) related to the banking contact.
05021	The following error(s) are related to the Registrar’s home address:
05022	End of error(s) related to the Registrar’s home address.
05023	Access denied.

CODING ERRORS	
Code	**Description**
10001	P_RGPC_NO and P_CTC_ID fields are mandatory
10002	Initiator and first Interventionist are mandatory.
10003	Unknown action.
10004	Parameter P_CTC_SAME_ADDR_ADMIN should not be null.
10005	Function unavailable.

ANNEX D PROCESS NAME ABBREVIATIONS

ARA	-	Applicant Registrar
ART	-	Applicant Registrant
BRN	-	Bulk Domain Registration Renewal
CAD	-	Change critical information
CFD	-	Confirm Domain
CFM	-	Change Request Confirmation
CNC	-	Cancel Domain
CRA	-	Change registrar
CRT	-	Change registrant
DTD	-	Domain term decrease
EDA	-	Enable domain
ERA	-	Reactivate Registrar
LOW	-	Registrar's low account
MRG	-	Merger of registrant
N07	-	7 days renewal notification
N30	-	30 days renewal notification
N60	-	60 days renewal notification
NRN	-	Request to not renew a domain
PWD	-	Change user password
RND	-	Domain Registration Renewal
RRA	-	Revoke Registrar
RRC	-	Registrar Re-Certification
RTF	-	Transfer business from one registrar to the other
SPD	-	Suspend Domain
SRA	-	Suspend Registrar
UPG	-	Upgrade domain

ANNEX E PROCESS STATUS ABBREVIATIONS

ACT - Active
APP - Approved
CAN - Cancelled
FIN - Finished
RFP - Registrar re-certification form approved - processing underway
RFS - Registrar re-certification form submitted
WAR - Waiting for alternate registrar acknowledgment
WP1 - Waiting for actual registrant initial acknowledgment
WP2 - Waiting for future registrant acknowledgment
WP3 - Waiting for actual registrant final acknowledgment
WRF - Waiting for registrar re-certification form
WRT - Waiting for registrant

Technical Specifications for Registrars
User Guide

Publication: RPPG 00012
Document Version: 1.4
Document Date: November 1, 2002
Publication Date: December 3, 2002
Publication Time: 09:00 Ottawa Time

Canadian Internet
Registration Authority

Autorité canadienne pour
les enregistrements Internet