Oracle® Health Sciences Information Gateway

Cross Community Access User Guide

Release 2.0.1

E37029-02

October 2013

This guide provides information on Oracle® Health Sciences Information Gateway (OHIG) Cross Community Access (XCA). It describes features and functionalities of Gateway, Integrating the Healthcare Enterprise (IHE) standards, and Web Services with their configuration options.

This document is intended for Oracle Health Information XCA Gateway users.

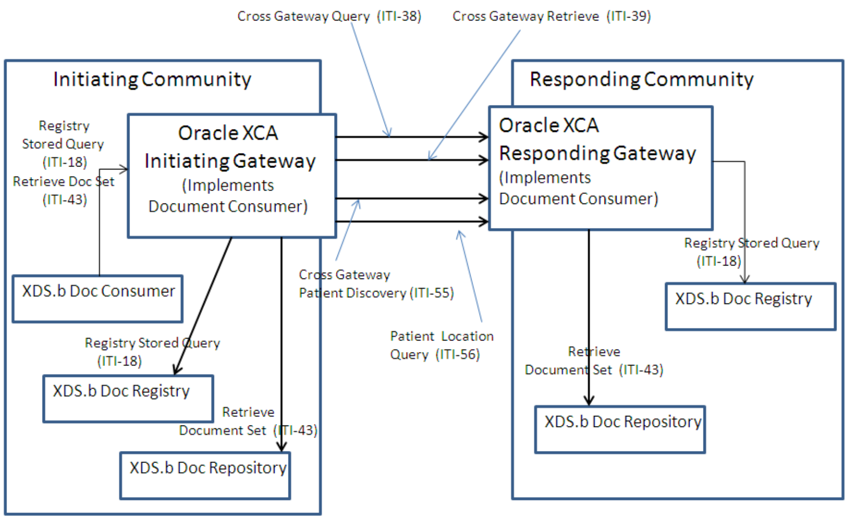
Overview

XCA keeps track of patients' documents by indexing them using the Document Metadata. XCA is IHE and Cross-Enterprise Document Sharing (XDS) standards compliant and implements the XDS Document Registry Actor.

Cross-Enterprise Document Sharing Actors and Transactions

Figure 1 shows the XDS actors and transactions among them.

Figure 1 XDS Actors and Transactions



Cross Gateway Query (ITI-38)
Cross Gateway Retrieve (ITI-39)
Initiating Community
Registry Stored Query (ITI-18)
Retrieve Doc Set (ITI-43)
Oracle XCA Initiating Gateway (Implements Document Consumer)
XDS.b Doc Consumer
Registry Stored Query (ITI-18)
XDS.b Doc Registry
Retrieve Document Set (ITI-43)
XDS.b Doc Repository
Cross Gateway Patient Discovery (ITI-55)
Patient Location Query (ITI-56)
Responding Community
Oracle XCA Responding Gateway (Implements Document Consumer)
Registry Stored Query (ITI-18)
XDS.b Doc Registry
Retrieve Document Set (ITI-43)
XDS.b Doc Repository

Actors and Transactions Supported by Gateway

XCA supports the following IHE profiles and transactions.

Table 1 Actors and Transactions Supported by XCA

Actors	Transactions
Initiating Gateway (IG)	Cross Gateway Query (ITI-38)
Initiating Gateway	Cross Gateway Retrieve (ITI-39)
Initiating Gateway	Registry Stored Query (ITI-18)
Initiating Gateway	Retrieve Document Set (ITI-43)
Responding Gateway (RG)	Cross Gateway Query (ITI-38)
Responding Gateway	Cross Gateway Retrieve (ITI-39)

Table 2 Actors and Transactions Supported by XCPD

Actors	Transactions
Initiating Gateway	Cross Gateway Patient Discovery (ITI-55)
Responding Gateway	Patient Location Query (ITI-56)
Initiating Gateway	Cross Gateway Patient Discovery (ITI-55)
Responding Gateway	Patient Location Query (ITI-56) Retrieve Document Set (ITI-43)

Services Provided

All of the IHE ITI transactions supported by XCA/XCPD are supported through SOAP 1.2 based Web Services. Following are the SOAP 1.2 Web Services supported by XCA/XCPD:

- Initiating Gateway Service (Registry Stored Query, Retrieve Document, Patient Discovery, Cross Gateway Query, Cross Gateway Retrieve)
- Responding Gateway Service (Patient Discovery, Cross Gateway Query, Cross Gateway Retrieve)

Web Services

Web Services are implemented using JAX-WS Web Services API and stack on both Oracle GlassFish and WebLogic servers. For more information on Web Service definitions and related IHE XDS transactions, see IHE IT Infrastructure XDS Profile specifications.

Core Gateway Service

The following Web Services operations and IHE transactions are supported for Core Gateway Services:

- ITI-18 Registry Stored Query

  The Registry Stored Query is classified into the following types:

  - FindDocuments
  - FindSubmissionSets
  - FindFolders

  - GetDocuments
  - GetFolders
  - GetAssociations
  - GetDocumentsAndAssociations
  - GetSubmissionSets
  - GetSubmissionSetAndContents
  - GetFolderAndContents
  - GetFoldersForDocument
  - GetRelatedDocuments
- ITI-43 Retrieve Document Set
- ITI-38 Cross Gateway Query
- ITI-39 Cross Gateway Retrieve
- ITI-55 XCPD Patient Discovery
- ITI-56 XCPD Patient Location Query

For details on these Web Services operations and IHE transactions, see `http://www.ihe.net/Technical_Framework`.

Deployment Environment

Core Gateway Services are implemented as Java Enterprise Edition (EE) components.

System Requirements

- Oracle Enterprise Linux 5.5 or higher, Microsoft Windows 2003, Microsoft Windows 2008, Microsoft Windows XP, and Microsoft Windows 7

Hardware Requirement

Following are the hardware requirements for installing XCA:

- 4 GB (4096 MB) of RAM for GlassFish
- 4 GB (4096 MB) of RAM for WebLogic
- 12 GB of disk space
- 16 GB of disk space for 64-bit

Software Requirement

Following are the software requirements for installing XCA:

- Java 1.6.0_23 or above
- Oracle Enterprise Linux 5.5 or higher
- Oracle Database 10+ (11g Release 1)
- Oracle GlassFish Server 2.1.1 Patch 17
- Oracle WebLogic Server 10.3.6.0 (11g Release 1)

Configuration

The configuration file is located under `config/xca/config` directory of the Application Server domain directory.

- GlassFish: `<GlassFish Home>/domains/<domain name>/config/xca/IG.properties<GlassFish Home>/domains/<domain name>/config/xca/RG.properties`
- WebLogic: `<Weblogic Middleware Home>/user_projects/domains/<domain name>/config/xca/IG.properties`

  `<Weblogic Middleware Home>/user_projects/domains/<domain name>/config/xca/RG.properties`

Configuring Initiating Gateway

Providing Local Home Community ID of the Initiating Gateway

Enter the following community IDs for configuring Initiating Gateway:

- `localHomeCommunityId_IG` =
- `localHomeCommunityId_XCPD =`

Configuring the Repository for Initiating Gateway

Prerequisite: Get the repository unique ID and repository URL for retrieving document transactions.

Update the configuration file as follows:

Syntax: `RepositoryUniqueId=RepositoryURL`

For example,

```
#1.3.6.1.4.1.21367.13.40.39=https://<hostname>:<port>/hd/services/xdsrepos
itoryb
```

Enabling the Grouping Option with Local Document Consumer

Prerequisite to enable grouping: Get the local community registry URL for Stored Query and Repository URL for retrieving document.

Set the following **INGWGroupedWithDocumentConsumer** property to **yes** to enable the grouping with local document consumer:

- `INGWLocalRegistry` - Takes the value of registry URL.
- `INGWLocalRepository` - Takes the value of repository URL.
- `INGWGroupedWithDocumentConsumer` = no

For example,

- `INGWLocalRegistry` =
- `INGWLocalRepository` =

Configuring Responding Gateway Using Home Community ID

Prerequisite: Responding Gateway Query and retrieve endpoints.

Following is the syntax for configuring the initiating gateway for a specific home community ID. You can configure multiple responding gateways.

Configuration for query transaction:

- Syntax: `CrossGatewayQuery_homecommunityid=RespondingGatewayQueryURL`
- Syntax: `CrossGatewayRetrieve_ homecommunityid=RespondingGatewayRetrieveURL`

Configuring Multiple Responding Gateways for Broadcasting Mode

Prerequisite: All the responding gateways query URLs and home community IDs that needs to be configured.

You can configure multiple responding gateways for the Cross Gateway Query queries by patient ID.

- `XCARespondingGateway_<count>` - This parameter takes the value of the Responding Gateway query URL.

  `<count>` is the variable which starts from one and can go to any number of Responding gateways that you would like to configure.

- `XCARespondingGateway_<count>_HomeCommunityId` - Takes the value of the Home community ID of the responding gateway.

For example, *when <count> value is 1,*

`XCARespondingGateway_1` =

`XCARespondingGateway_1_HomeCommunityId` =

When <count> value is 2,

`XCARespondingGateway_1` =

`XCARespondingGateway_1_HomeCommunityId` =

`XCARespondingGateway_2` =

`XCARespondingGateway_2_HomeCommunityId` =

As mentioned, `<count>` is the number of responding gateways that you plan to configure.

Configuring Local MPI to Initiating Gateway

Prerequisite: Local MPI PDQ Supplier endpoint.

`XCA_Local_PDQSupplier` takes the value of the PDQ supplier endpoint URL.

For example,

`XCA_Local_PDQSupplier =`

ATNA Audit Configuration

Prerequisites: Audit repository host name or IP and audit repository UDP or TLS port.

- `ApplicationName`, `sourceApplicationId`, and `sourceEnterpriseId` - Represent the system in the audit message. This can be the name of the clinic.
- `alternateUserId` - Takes the string value. Any user identifier is preferred. This parameter is used if the actual user ID is not found in the transactions.
- `ATNASyslogProtocol` - This value should be set to UDP or TLS.

To enable auditing, set **Audit** to `Yes`.

For example, Audit Configuration:

```
auditRepositoryServer =

auditRepositoryPort =

sourceApplicationId =

sourceEnterpriseId =

alternateUserId =

ApplicationName =

ATNASyslogProtocol =

Audit = no
```

> **Note:** For TLS auditing in WebLogic, ensure to start the WebLogic with the following JVM options for the keystore and truststore file:
>
> ```
> -Djavax.net.ssl.keyStore=<keystore file>
>
> -Djavax.net.ssl.keyStorePassword=<keystore pass>
>
> -Djavax.net.ssl.keyStoreType=<keystore type>
>
> -Djavax.net.ssl.trustStore=<truststore file>
>
> -Djavax.net.ssl.trustStorePassword=<truststore pass>
> ```

Configuring Number of Threads and Timeout for Initiating Gateway

You can configure one initiating gateway for multiple responding gateways. Multiple threads ensure better performance.

- `maximumThreadCount` - Takes the value of max number of threads that you want to create.

  For example, number of threads required to send the Cross Gateway requests:

  `maximumThreadCount =`

Time out configurations for the requests:

- `default_timeout_sync` - Takes the value of the time out for synchronous transactions.
- `default_timeout_async` - Takes the value of the time out for asynchronous transactions.

For example,

- `default_timeout_sync =`
- `default_timeout_async =`

Configuring XCPD Initiating Gateway

Configuring XCPD Responding Gateway

Prerequisite: XCPD URL and Homecommunity ID of the responding gateway.

- `XCPD_RespondingGW_<TargetHomeCommunityID>` - Takes the value of the responding gateway XCPD URL.

  `<TargetHomeCommunityID>` should be replaced with the homecommunity ID of the responding gateway.

```
XCPD_RespondingGW_TargetHomeCommunityID = XCPD Responding Gateway URL
```

For example, `XCPD_RespondingGW_1.0 = http://localhost:8080/ RespondingGateway_Service/XCPDRespondingGateway`

Configuring Sender and Receiver OIDs

The following properties take sender and receiver OID values appropriately.

- `XCPD_IG_SenderOID` =
- `XCPD_IG_RecieverOID` =

Patient ID Mapping Workflow

The property `PatientID_Mapping_Workflow` takes two values.

- `xca` - When the value is xca, initiating gateway does not send any XCPD request to find patient ID in remote community. IG uses the same patient id that is sent by the document consumer.
- `xcpd`: When the value is xcpd, the initiating gateway will send XCPD request to each configured responding gateway, fetch the patient ID, and uses that patient ID for the respective Cross Gateway Query Transactions.

  For example,

  `PatientID_Mapping_Workflow` =

Configuring Responding Gateway

Configuring Responding Gateway Local Home Community

Enter the following IDs for configuring responding gateway local home community:

- `localHomeCommunityId_RG` =
- `localHomeCommunityId_XCPD` =

Configuring Responding Gateway's Local Registry Repository

Prerequisite: Responding Gateway's local registry, repository URLs with repository unique ID.

- `RespondingGatewayRegistryURL =`
- `RespondingGatewayRepositoryID` =

Prerequisite: Get repository unique and repository URL for retrieving document transactions.

Update the configuration file as follows:

Syntax: `RepositoryUniqueId=RepositoryURL`

For example,

```
1.3.6.1.4.1.21367.13.40.39=http://<hostname>:<port>/services/xdsrepository
b
```

ATNA Audit Configuration

Prerequisites: Audit repository host name or IP and audit repository UDP or TLS port.

- `ApplicationName`, `sourceApplicationId`, and `sourceEnterpriseId` - Represent the system in the audit message. This can be the name of the clinic.
- `alternateUserId` - Takes the string value. Any user identifier is preferred. This value is used if the actual user ID is not found in the transactions.
- `ATNASyslogProtocol` - This value should be set to UDP or TLS.

To enable auditing, set **Audit** to `Yes`.

For example, Audit Configuration:

- `auditRepositoryServer` =
- `auditRepositoryPort` =
- `sourceApplicationId` =
- `sourceEnterpriseId` =
- `alternateUserId` =
- `ApplicationName` =
- `ATNASyslogProtocol` =
- `Audit` = no

> **Note:** For TLS auditing in WebLogic, ensure to start the WebLogic with the following JVM options for the keystore and truststore file:
>
> ```
> -Djavax.net.ssl.keyStore=<keystore file>
>
> -Djavax.net.ssl.keyStorePassword=<keystore pass>
>
> -Djavax.net.ssl.keyStoreType=<keystore type>
>
> -Djavax.net.ssl.trustStore=<truststore file>
>
> -Djavax.net.ssl.trustStorePassword=<truststore pass>
> ```

Timeout Configurations for the Requests

- `default_timeout_sync` - Takes the value of the time out for synchronous transactions.
- `default_timeout_async` - Takes the value of the time out for asynchronous transactions.

For example,

- `default_timeout_sync` =
- `default_timeout_async` =

Configuring Local MPI to Responding Gateway

Prerequisite: Local MPI PDQ Supplier endpoint.

- `XCPD_RG_PDQSupplier<count>` - Takes the value of the PDQ endpoint of the MPI.
- `XCPD_RG_PDQSupplier<count>_domainID` - Takes the value of the domain ID.

For example, IHERED, IHEBLUE, and so on.

XCPD Responding Gateway settings:

You can have multiple PDQ Suppliers to talk with.

- `XCPD_RG_PDQSupplier<count> =`
- `XCPD_RG_PDQSupplier<count>_domainID =`

`<count>` can be replaced with any number of PDQ suppliers that are planned to configure. Responding gateway can look through multiple MPI systems to search for a patient.

For example, when `<count>` is 1,

```
XCPD_RG_PDQSupplier1 =
```

```
XCPD_RG_PDQSupplier1_domainID =
```

When `<count>` is 2,

```
XCPD_RG_PDQSupplier1 =
```

```
XCPD_RG_PDQSupplier1_domainID =
```

```
XCPD_RG_PDQSupplier2 =
```

```
XCPD_RG_PDQSupplier2_domainID =
```

Configuring Health Data Locator

To enable Health Data Locator, set the value of `SupportsHealthDataLocatorBelow` property in the RG.properties file to `yes`.

If the value is set to `yes`, RG responds to the XCPD request indicating that it supports patient location query.

If the value is set to `no`, RG does not support Health Data Locator.

Transactions and Web Service Uniform Resource Locator

Table 3 lists the Web Services supported by XCA. You can find the Web Service WSDL by suffixing endpoint Uniform Resource Locator (URL) with `?wsdl`.

Table 3 Transactions and Web Service URL

Transaction	Synch	Asynch	Endpoint URL
Cross Patient Discovery (ITI-55)	Yes	Yes	http(s)://<XCA_HOST>:<PORT>/RespondingGateway_Service/XCPDRespondingGateway
Registry Stored Query (ITI-18)	Yes	No	http(s)://<XCA_HOST>:<PORT>/InitiatingGatewayQuery_Service/XCAInitiatingGatewayQuery
Retrieve Document Set (ITI-43)	Yes	No	http(s)://<XCA_HOST>:<PORT>/InitiatingGatewayRetrieve_Service/XCAInitiatingGatewayRetrieve
Cross Document Query (ITI-38)	Yes	Yes	http(s)://<XCA_HOST>:<PORT>/RespondingGatewayQuery_Service/XCARespondingGatewayQuery

Table 3 (Cont.) Transactions and Web Service URL

Transaction	Synch	Asynch	Endpoint URL
Cross Document Retrieve (ITI-39)	Yes	Yes	http(s)://<XCA_HOST>:<PORT>/RespondingGatewayRetrieve_Service/XCARespondingGatewayRetrieve
Patient Location Query (ITI-56)	Yes	Yes	http(s)://<XCA_HOST>:<PORT>/RespondingGateway_Service/XCPDRespondingGateway
Asynchronous Registry Stored Query	No	Yes	http(s)://<XCA_HOST>:<PORT>/IGAsyncServices/XCAInitiatingGatewayQuery
Asynchronous Retrieve Document Set	No	Yes	http(s)://<XCA_HOST>:<PORT>/IGAsyncServices/XCAInitiatingGatewayRetrieve

Oracle Extensions

As per the IHE specification, XCA and XCPD are two different profiles. The Oracle implementation is merged together. The Initiating Gateway will first send the XCPD request, find out the patient identifiers in the remote community, and then use this patient identifier for the next cross gateway query call to the responding gateway.

Security Configuration Issues

This section describes security configuration issues you must consider when implementing XCA.

General Security Principles

The following are fundamental principles for using any application securely:

Keep software up-to-date

Keep all software versions and patches up-to-date.

Keep up-to-date on latest security information critical patch

Oracle continually improves its software and documentation. Critical patch updates are the primary means of releasing security fixes for Oracle products to customers with valid support contracts. Oracle recommends you to apply these patches as soon as they are released.

Configure strong passwords on the database

Repeat the following basic rule of security management.

Ensure all passwords are strong. You can strengthen passwords by creating and using password policies for your organization. For guidelines on securing passwords and for additional ways to protect passwords, refer to the *Oracle® Database Security Guide* specific to the database release you are using.

You should modify the following passwords to use your policy-compliant strings:

- Passwords for the database default accounts, such as SYS and SYSTEM.
- Passwords for the database application-specific schema accounts, such as Gateway.

- Password for the database listener.

  Oracle recommends that you do not configure a password for the database listener as it will enable remote administration. For more information, refer to the section *Removing the Listener Password* of *Oracle® Database Net Services Reference 11g Release 2 (11.2)*.

Follow the principle of least privilege

The principle of least privilege states that users should be given the least amount of privilege to perform their jobs. Overly ambitious granting of responsibilities, roles, grants - especially early on in an organization's life cycle when people are few and work needs to be done quickly - often leaves a system wide open for abuse. User privileges should be reviewed periodically to determine relevance to current job responsibilities.

To restrict access, it is recommended to have the following default file permissions in Unix environment.

- 740 for executable
- 640 for regular files

Managing default user accounts

Lock and expire default user accounts.

Closing all open ports when not in use

Keep only the minimum number of ports open. You should close all ports when not in use.

Disabling the Telnet service

Oracle XCA standard configuration does not use the Telnet service. By default, Telnet listens on port 23. Telnet, which sends clear-text passwords and user names through a log in, is a security risk to your servers. If the Telnet service is available on any system, it is recommended to disable Telnet in favor of Secure Shell (SSH). Disabling Telnet protects your system security.

Disabling Other Unused Services

In addition to not using Telnet, the Oracle XCA Gateway standard configuration does not use the following services or information for any functionality:

- Simple Mail Transfer Protocol (SMTP) - This protocol is an Internet standard for e-mail transmission across Internet Protocol (IP) networks.
- Identification Protocol (identd) - This protocol is generally used to identify the owner of a TCP connection on UNIX.
- Simple Network Management Protocol (SNMP) - This protocol is a method for managing and reporting information about different systems.

Restricting these services or information does not affect the use of Oracle XCA standard configuration. If you are not using these services for other applications, it is recommended to disable these services to minimize your security exposure. If you need SMTP, identd, or SNMP for other applications, ensure to upgrade to the latest version of the protocol to provide the most up-to-date security for your system.

Designing multiple layers of protection

When designing a secure deployment, design multiple layers of protection. If a hacker gains access to one layer, such as Application server, that should not automatically give them easy access to other layers, such as the database server.

Providing multiple layers of protection may include:

- Enabling only those ports required for communication between different tiers. For example, only allow communication to the database tier on the port used for SQL*NET communications (by default, 1521).
- Placing firewalls between servers so that only expected traffic can move between servers.

Utilizing SSL

Consider utilizing Application Server SSL service for the XCA application. The XCA application is a standard Java EE application and can utilize an industry standard security infrastructure and framework. There is no configuration required on the XCA application. The application Server (WebLogic or GlassFish) provides SSL service. For more information about configuring SSL to achieve SSL security for XCA, see the Application Server's documentation.

When SSL or TLS is configured, it is recommended to use TLS_RSA_WITH_AES_128_CBC_SHA cipher instead of SSL_RSA_WITH_3DES_EDE_CBC_SHA for TLS authentication.

Performance Tuning

GlassFish

Oracle recommends the following parameter if XCA is running on the Oracle GlassFish server:

- Minimum Java heap size 512 MB
- Maximum Java heap size 2048 MB

Related Documents

Refer to the following links for standard definitions of:

- Integrating the Healthcare Enterprise (IHE) Actors: http://wiki.ihe.net/index.php?title=Actors
- IHE Profiles and Standards: http://www.ihe.net/profiles/index.cfm
- IT Infrastructure Domain: http://wiki.ihe.net/index.php?title=IT_Infrastructure
- Cross-Enterprise Document Sharing (XDS): http://wiki.ihe.net/index.php?title=Cross-Enterprise_Document_Sharing

Appendix A: Acronyms

This section provides a list of commonly used acronyms.

- IG - Initiating Gateway

- IHE - Integrating the Healthcare Enterprise
- RG - Responding Gateway
- XCA - Cross Community Access
- XCPD - Cross-Community Patient Discovery
- XDS - Cross-Enterprise Document Sharing

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Oracle® Health Sciences Information Gateway Cross Community Access User Guide, Release 2.0.1
E37029-02