Oracle® Health Sciences Information Manager

Health Record Locator Users Guide

4.0
F50768-01
January 2022

Oracle Health Sciences Information Manager Health Record Locator Users Guide, 4.0

F50768-01

Contents

Preface

This preface contains the following sections:

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.

Related resources

All documentation and other supporting materials are available on the Oracle Help Center.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Access to Oracle Support

Oracle customers that have purchased support 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.

1

Get started

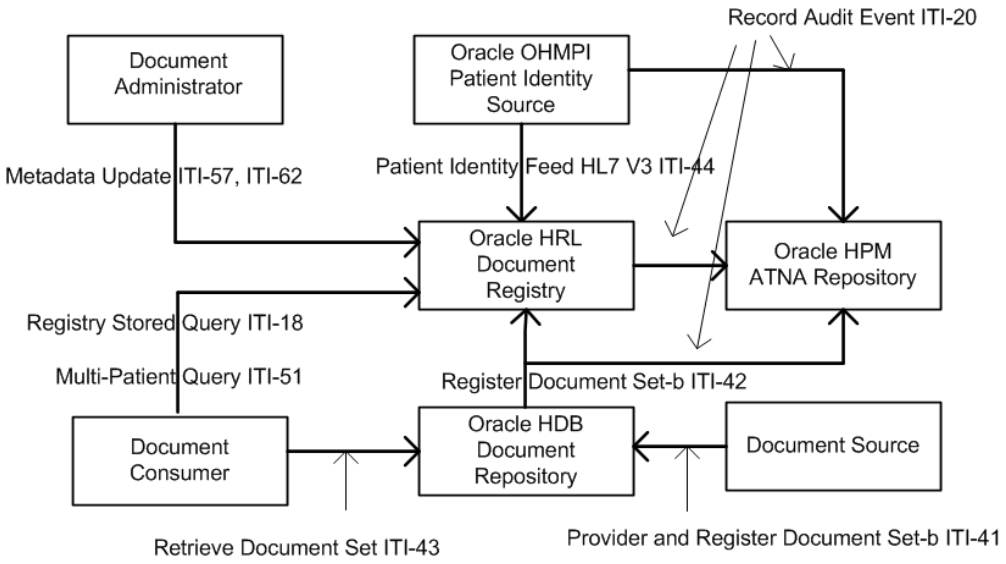
This section introduces the Health Sciences Information Manager Health Record Locator.

Health Record Locator keeps track of patient documents by indexing them using the Document Metadata (DSUB). Health Record Locator is IHE and Cross-Enterprise Document Sharing (XDS) standards compliant and implements the XDS Document Registry Actor.

- Cross-enterprise document sharing actors and transactions
  The diagram shows the XDS actors and transactions among them. It does not contain actors and transactions related to Document Metadata Subscription (DSUB).
- Supported IHE profiles
  This section lists the supported IHE profiles, the required version, and the URL of the profile definition.
- Services provided
  This section describes the IHE ITI transactions that Health Record Locator supports.
- Deployment environment
  This section describes the Health Record Locator components environment requirements.
- Hardware requirements
  This sections defines the minimum hardware requirements for Health Record Locator.
- Software requirements
  This section describes the minimum software requirements for the Health Record Locator.
- Related documents
  This section lists useful reference documents.
- Acronyms
  This section defines commonly used in this user guide.

Cross-enterprise document sharing actors and transactions

The diagram shows the XDS actors and transactions among them. It does not contain actors and transactions related to Document Metadata Subscription (DSUB).



- Actors and transactions supported by the Health Record Locator
  This section provides a reference of the IHE profiles and transactions supported by Health Record Locator

Actors and transactions supported by the Health Record Locator

This section provides a reference of the IHE profiles and transactions supported by Health Record Locator

Health Record Locator supports the following IHE profiles and transactions shown in the table below:

Table 1-1 Actors and Transactions Supported by HRL

Profile	Actor	Option	ITI TXN Number[1]
MPQ[2]	Document Registry	None	ITI-51
MPQ	Document Registry	Asynchronous Web Services Exchange	ITI-51
XDS.b[3]	Document Registry	Patient Identity Feed (HL7 V3)	ITI-44
XDS.b	Document Registry	None	ITI-18 ITI-42
XDS.b	Document Registry	Asynchronous Web Services Exchange	ITI-18 ITI-42
XDS.b	Document Registry	Document Metadata Update	ITI-57 ITI-62
XPID[4]	Document Registry	None	ITI-64
XDS.b	Document Registry	Patient Identity Feed (HL7 V2)	ITI-8

Table 1-1 (Cont.) Actors and Transactions Supported by HRL

Profile	Actor	Option	ITI TXN Number[1]
DSUB[5]	Document MetadataNotification BrokerDocument MetadataPublisher	None	ITI-52 ITI-53 ITI-54

Footnotes:

- **1** IT Infrastructure Transaction Number
- **2** MPQ - Multi-patient Queries
- **3** XDS.b - Cross-Enterprise Document Sharing
- **4** XPID - XAD-PID Change Management Profile
- **5** DSUB - Document Metadata Subscription

Supported IHE profiles

This section lists the supported IHE profiles, the required version, and the URL of the profile definition.

Table 1-2 Supported IHE profiles

Profile name	Version	Location
Cross-Enterprise Document Sharing (XDS)	Revision 17.0 July 20, 2020	https://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Rev17-0_Vol2a_FT_2020-07-20.pdf https://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Rev17-0_Vol2b_FT_2020-07-20.pdf https://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Rev17-0_Vol2x_FT_2020-07-20.pdf https://www.ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_TF_Rev17-0_Vol3_FT_2020-07-20.pdf

Table 1-2 (Cont.) Supported IHE profiles

Profile name	Version	Location
Document Metadata Subscription (DSUB)	Trial Implementation October 13, 2014	http://ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_Suppl_DSUB.pdf
XAD-PID Change Management (XPID)	Trial Implementation October 13, 2014	http://ihe.net/uploadedFiles/Documents/ITI/IHE_ITI_Suppl_XPID.pdf

Services provided

This section describes the IHE ITI transactions that Health Record Locator supports.

Most of the IHE ITI transactions that Health Record Locator supports are through SOAP 1.2 based Web Services. The following are the SOAP 1.2 Web Services that Health Record Locator supports:

- Core Registry Service (Patient Feed, Register Document Set, and Registry Stored Query) (see "Core Registry Service")
- Multi-patient Query Service (see "Multi-patient Query Service")
- Metadata Update Service (see "Metadata Update Service")
- Core Registry services
  This section lists the Health Record Locator Core Registry web service operations.
- Multi-patient Query Service
  The following Web Services operations and IHE transactions are supported for Multi-patient Query Service:
- Metadata Update Service
  This section describes web service operations supported for the Metadata Update Service.
- DSUB Service
  This section lists the Health Record Locator DSUB web service operations.
- HL7v2 services
  This section lists the Health Record Locator HL7v2 web service operations.

Core Registry services

This section lists the Health Record Locator Core Registry web service operations.

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

- ITI-44 Patient Identity Feed
- ITI-42 Register Document Set

- ITI-18 Registry Stored Query
  The Registry Stored Query is classified into the following types:
  - FindDocuments
  - FindDocumentsByReferenceId
  - FindSubmissionSets
  - FindFolders
  - GetDocuments
  - GetFolders
  - GetAssociations
  - GetDocumentsAndAssociations
  - GetSubmissionSets
  - GetSubmissionSetAndContents
  - GetFolderAndContents
  - GetFoldersForDocument
  - GetRelatedDocuments GetAll

Multi-patient Query Service

The following Web Services operations and IHE transactions are supported for Multi-patient Query Service:

The following Web Services operations and IHE transactions are supported for Multi-patient Query Service:

- ITI-51 Multi-patient Query. The Multi-patient Query is classified into the following types:
  - FindDocumentsForMultiplePatients
  - FindFoldersForMultiplePatients

Metadata Update Service

This section describes web service operations supported for the Metadata Update Service.

The following Web Services operations and IHE transactions are supported for Metadata Update Service:

- ITI-57 Update Document Set
- ITI-62 Delete Document Set

DSUB Service

This section lists the Health Record Locator DSUB web service operations.

The following Web Services operations and IHE transactions are supported for DSUB Service:

- ITI-54 Document Metadata Publish
- ITI-52 Document Metadata Subscribe or Document Metadata Unsubscribe

- ITI-53 Document Metadata Notify

HL7v2 services

This section lists the Health Record Locator HL7v2 web service operations.

The following IHE transactions are supported in HL7v2 Server:

- ITI-8 Patient Identity Feed
- ITI-64 Notify XAD-PID Link Change

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

Deployment environment

This section describes the Health Record Locator components environment requirements.

The Core Registry, Multi-patient Query, and Metadata Update Services are implemented as Java Enterprise Edition (EE) components.

HL7v2 Services are implemented as an optional application server component called XPID. You must deploy the XPID component on the same application server instance as that of the Core Registry Services component.

The DSUB Document Metadata Notification Broker service is implemented in an optional Java EE component. You may choose to deploy it on the same application server instance as that of Core Registry Services or on a separate instance.

DSUB Document Metadata Publisher is part of Core Registry Services component. It is enabled through the configuration parameter.

Hardware requirements

This sections defines the minimum hardware requirements for Health Record Locator.

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

Software requirements

This section describes the minimum software requirements for the Health Record Locator.

- Java 1.8 executable in path (for installer)
- Apache Ant 1.10.11 or later. The executable must be in the path `PATH=$PATH:<install_dir>/apache-ant-1.10.11/bin`
- Oracle JDK 1.8.0_311+ and WebLogic Server 12c (12.2.1.4)
- Oracle Database 12cR1 (12.1.0.2.0), Oracle Database 12cR2 (12.2.0.1), or Oracle Database 19c (19.3.0.0.0)

- Oracle Enterprise Linux 7.x or higher
- Microsoft Windows x64 (64-bit) 2012 R2
- Microsoft Windows x64 (64-bit) 2012
- Microsoft Windows x64 (64-bit) 2008 R2

Related documents

This section lists useful reference 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

Acronyms

This section defines commonly used in this user guide.

- **DSUB**: Document Metadata Subscription
- **HRL**: Health Record Locator
- **IHE**:ntegrating the Healthcare Enterprise
- **MPQ**: Multi-patient Queries
- **OHIM**:Oracle Health Sciences Information Manager
- **XDS**:Cross-Enterprise Document Sharing
- **XPID**: XAD-PID Change Management Profile

2

Configure Health Sciences Information Manager Health Record Locator

This chapter provides configuration information about the various components of Health Record Locator.

- Configuration file
  This section provides the location of the Health Record Locator configuration file.
- ATNA UDP or TLS message properties
  This section provides a reference to properties needed to enable ATNA UDP or TLS messages.
- Other HomeCommunity-level properties
  This section provides descriptions of additional HomeCommunity-level properties.
- Registry-level properties
  Registry-level properties for Health Record Locator
- DSUB properties
  This section provides configuration properties for DSUB.
- XPID properties
  This section provides XPID configuration properties in the codes file and the `xpid.properties` file.
- Codes file updates
  This section describes the location of the codes file.
- Transactions and web service URLs
  This reference provides tables of endpoints for Health Record Locator, DSUB, and XPID.
- Oracle database performance tuning
  Oracle recommends the generic Oracle database optimizations for Health Record Locator and on the specific database behavior.

Configuration file

This section provides the location of the Health Record Locator configuration file.

The Health Record Locator configuration file (`xconfig.xml`) is located under the `config/hrl/config` directory of the Application Server domain directory.

The full path for the WebLogic server is: `<Weblogic_Middleware_Home>/user_projects/domains/<domain_name>/config/hrl/config/xconfig.xml`.

Restart the application server to make `xconfig.xml` changes take effect. The following is the structure of the `xconfig.xml` file. Many Health Record Locator properties are set under the HomeCommunity and Registry elements.

```
<?xml version="1.0" encoding="utf-8"?>
<Config>
```

```
    <HomeCommunity name="home">
        <Property name="propName1">propVal1</Property>
        ...
    </HomeCommunity>
    <Registry name="localregistry">
        <Property name="propName2">propVal2</Property>
        ...
    </Registry>
</Config>
```

ATNA UDP or TLS message properties

This section provides a reference to properties needed to enable ATNA UDP or TLS messages.

To enable sending ATNA UDP or TLS messages, edit the value of the following properties under the HomeCommunity element and specify ATNA UDP or TLS server details.

Table 2-1 ATNA UDP or TLS messages TLS message properties

Property	Description
ATNAPerformAudit	Set to true to enable sending ATNA audit messages. By default, this value is set to false.
ATNAsyslogProtocol	Set to UDP or TLS (default value).
ATNAsyslogHost	Set to the ATNA UDP or TLS server host name or IP address.
ATNAsyslogPort	Set to the ATNA UDP or TLS server port number.
ATNAMessageType	Set to **RFC3881** or **DICOM** to indicate which audit message format should be generated by the system.

Ensure to configure the following properties when you use TLS for ATNAsyslogProtocol.

Table 2-2 TLS for ATNAsyslogProtocol properties

Property	Description
KeyStore	Set to the file path of the keystore. For example, `/home/common/cert/keystore.jks`.
KeyStoreType	Specify the type of the keystore. By default, the value is set to JKS.
TrustStore	Enter the file path of the truststore. For example, `/home/common/cert/keystore.jks`.
TrustStoreType	Specify the type of the truststore. By default, the value is set to JKS.
CredentialStore	Enter the directory where Oracle wallet is created. For example, `/home/common`.

Other HomeCommunity-level properties

This section provides descriptions of additional HomeCommunity-level properties.

Table 2-3 Other HomeCommunity-level properties

Property	Description
`ValidatePatientId`	Set this value to true (default value) to validate known patient IDs before registering DocumentEntry.
`XMLSchemaValidationEnabled`	Set this value to true (default value) to schema validate incoming messages.
`LogEnabled`	Set this value to true (default value) to enable logging registry request and response messages in Log schema tables. **NOTE:** This parameter is different from enabling ATNA audit log messages.
`ValidateAuthorRoleAndSpecialty`	Set this value to true (default value is false) to validate authorSpecialty and authorRole slot values are not empty.

Registry-level properties

Registry-level properties for Health Record Locator

Table 2-4 Registry-level properties

Property	Description
`ReceiverDeviceId`	Set this value to construct response messages in HL7v2 services.
`ReceiverDeviceName`	Set this value to construct response messages in HL7v2 services. By default, this value is set to `ORACLE_HIA_RLS_XDSbRegistry`.
`AcceptPIDOnlyFrom`	Set this property to let the Registry accept patient feed only from the specified domain. Comment or delete this property to let the Registry accept patient feed from all domains.
`MaxLeafObjectsAllowedFromQuery`	Set an integer value that determines the maximum number of document entries returned with Registry Stored Query response messages. By default, this value is set to 25. **Note**: This property is applicable only when the query request contains return type value LeafClass.
`TrimLogQueueMessages`	Set this value to true (default value) to trim the messages logged in log schema tables.
`MaxLeafObjectsPerLogQueueMsg`	Set the maximum number of Leaf objects to log per message.
`MaxObjectRefsPerLogQueueMsg`	Set the maximum value of Object references to log per message.

Table 2-4 (Cont.) Registry-level properties

Property	Description
ValidateCodeDisplayName	Set this value to true to let Health Record Locator validate the code display names in the registry metadata of the request against the codes.xml file. By default, this value is set to false.
ValidateDocEntryURISlotValue	Set this value to true to let Health Record Locator validate the URI slot value in the document metadata of the request. By default, this value is set to false.
AffinityDomainPIDAssigningAuthorityID	Set this property to let registry append the Assigning Authority ID to the received Patient ID, which does not have this value. After updating the configured Assigning Authority Id value, the registry persists the Patient ID in the database. By default, this property is commented. To let Health Record Locator employ the above specified behavior, uncomment this property. The format of this property value is: &OIDOfAssigningAuthorityID&ISO For example: <Property name="AffinityDomainPIDAssigningAuthorityID"> &1.3.6.1.4.1.21367.2005.3.7&ISO </Property>
AllowDuplicateUniqueIdInSubmission	Set this value to true to let Health Record Locator duplicate unique IDs in a submission. By default, this value is set to false.
ValidateCodes	Set this value to true to let Health Record Locator validate all CodeType code values in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateClassCode	Set this value to true to let Health Record Locator validate Class code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateContentTypeCode	Set this value to true to let Health Record Locator validate Content Type code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateFormatCode	Set this value to true to let Health Record Locator validate Format code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateTypeCode	Set this value to true to let Health Record Locator validate Type code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateAssociationDocumentation	Set this value to true to let Health Record Locator validate Association Documentation code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.

Table 2-4 (Cont.) Registry-level properties

Property	Description
ValidateFolderCodeList	Set this value to true to let Health Record Locator validate Folder CodeList code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidatePracticeSettingCode	Set this value to true to let Health Record Locator validate Practice Setting code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateEventCodeList	Set this value to true to let Health Record Locator validate Event CodeList code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateHealthcareFacilityTypeCode	Set this value to true to let Health Record Locator validate Healthcare Facility Type code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true.
ValidateConfidentialityCode	Set this value to true to let Health Record Locator validate Confidentiality code value in the registry metadata of the request against the codes.xml file. By default, this value is set to true. 2.9.4 DSUB

DSUB properties

This section provides configuration properties for DSUB.

Table 2-5 DSUB properties

Property	Description
NotificationEnabled	Set this value to true to enable publishing registry events to the DSUB Notification Broker. By default, this value is set to false.
PublishEndPoint	Set the publish endpoint URL of the DSUB Notification Broker.
DsubValidateCodeAndCodingScheme	Set this value to true (default value) to validate code and coding scheme containing DSUB subscription message against codes file of the registry.
DefaultDaysBeforeExpiryOfSubscription	Set an integer value that indicates the number of days after which the subscription will expire. By default, this value is set to 30.
NotificationBrokerSubscribeEndPoint	Set the Subscribe endpoint URL of the DSUB Notification Broker.
DeleteExpiredSubscriptionsIntervalDuration	Set the number of milliseconds between successive invocations of the batch job to delete expired subscriptions (if any) from the database. Enter an integer value specifying milliseconds for the interval. The default value is 86400000 (1 day).

Table 2-5 (Cont.) DSUB properties

Property	Description
DeleteExpiredSubscriptionsTimerStartInterval	Set the duration in milliseconds after the receipt of the first subscribe request to the Notification Broker. The batch job to delete the expired records becomes active at this point. Enter an integer value specifying the milliseconds interval. The default value is 86400000 (1 day).

XPID properties

This section provides XPID configuration properties in the codes file and the `xpid.properties` file.

Table 2-6 XPID configuration properties

Property	File	Description
xpid.classification.scheme	Codes file	Content type classification coding scheme.
xpid.classification.code	Codes file	Content type code for the coding scheme.

Codes file updates

This section describes the location of the codes file.

The codes file is located under `config/hrl/codes` directory of the application server domain directory.

WebLogic: `<Weblogic_Middleware_Home>/user_projects/domains/<domain_name>/config/hrl/codes/codes.xml`

You can update these files with new codes as applicable. Restarting the application server is not required for new codes to take effect.

Transactions and web service URLs

This reference provides tables of endpoints for Health Record Locator, DSUB, and XPID.

You can find the Web Service WSDL by suffixing endpoint Uniform Resource Locator (URL) with `?wsdl`.

Table 2-7 Record Locator transaction and web service URLs

Transaction	Sync	Async	Endpoint URL
Register Document Set-b [ITI-42]	Yes	Yes	http(s)://<HRL_HOST>:<PORT>/hrl/regsvc

Table 2-7 (Cont.) Record Locator transaction and web service URLs

Transaction	Sync	Async	Endpoint URL
Registry Stored Query [ITI-18]	Yes	Yes	`http(s)://<HRL_HOST>:<PORT>/hrl/regsvc`
Patient Identity Feed(HL7 V3) [ITI-44]	Yes	Yes	`http(s)://<HRL_HOST>:<PORT>/hrl/regsvc`
Multi Patient Query [ITI-51]	Yes	Yes	`http(s)://<HRL_HOST>:<PORT>/hrl/regmpqsvc`
Metadata Update - Update [ITI-57]	Yes	Yes	`http(s)://<HRL_HOST>:<PORT>/hrl/regupdsvc`
Metadata Update - Delete [ITI-62]	Yes	Yes	`http(s)://<HRL_HOST>:<PORT>/hrl/regupdsvc`

Use the custom endpoints below to support

- Minimal Document Metadata Update: `http(s)://<HRL_HOST>:<PORT>/hrl/regupdsvc`
- Web Services Atomic Transaction for Registry: `http(s)://<HRL_HOST>:<PORT>/hrl/regatsvc`

: :

Use the endpoints in the following table to configure your Document Metadata publisher and subscribers as needed.

Table 2-8 DSUB transaction and endpoint URLs

Transaction	Endpoint URL
Document Metadata Publish [ITI-54]	`http(s)://<DSUB_HOST>:<PORT>/WS-BrokeredNotificationPubl ish_Service/Publish`
Document Metadata Subscribe [ITI-52]	`http(s)://<DSUB_HOST>:<PORT>/WS-BrokeredNotificationSubscribe_Service/Subscribe`
Document Metadata Unsubscribe [ITI-52]	`http(s)://<DSUB_HOST>:<PORT>/WS-BrokeredNotificationUnSubscribe_Service/Unsubscribe?subscriptionId=<UUID>` `<UUID>` should be a valid value as assigned by DSUB Notification Broker.

Use the endpoints in the table below to configure PIX Manager as needed.

Table 2-9 XPID transaction and endpoint URLs

Transaction	Endpoint URL

Notify XAD-PID Link Change [ITI-64]	`hl7://<XPID_HOST>:<PORT>`
	
	The port number is specified in
	
	`<domain_name>/config/hrl/xpid.properties`

Oracle database performance tuning

Oracle recommends the generic Oracle database optimizations for Health Record Locator and on the specific database behavior.

Table 2-10 Oracle database performance parameters

Parameter	Value
db_cache_size	1 GB
memory_target	8 GB
memory_max_target	8 GB
log_buffer	3 MB
LARGE_POOL_SIZE	100 MB
PGA_AGGREGATE_TARGET	2 GB
SGA_MAX_SIZE	4 GB
SGA_TARGET	4 GB
SHARED_POOL_SIZE	1 GB
processes	1200
session	1350
open_cursors	1200
java_pool_size	100 MB

3
Usage notes

The section arranges a number of important use cases for your review and use as needed.

- Latest deprecated version of DocumentEntry
  This section describes how to use the FindDocument query.
- AuthorPerson query
  This section describes how to use the AuthorPerson query.
- Large numbers of DocumentEntry objects
  This section describes how to use queries to limit the number of DocumentEntry objects returned.
- ITI-62 transaction simulation
  This section describes how Health Record Locator simulates ITI-62 transactions.
- Custom query parameter $orcl.order.by

Latest deprecated version of DocumentEntry

This section describes how to use the FindDocument query.

When you execute the FindDocument Registry Stored Query (ITI-18) with status parameter value of deprecated (`urn:oasis:names:tc:ebxml-regrep:StatusType:Deprecated`), the registry returns all document versions with the status deprecated as per the IHE specification.

To query only the latest deprecated version of DocumentEntry, a new status parameter value deleted (`urn:orcl.reg:names:StatusType:Deleted`) is added to Health Record R Locator. When this status parameter value deleted is specified in the query, registry returns the latest version of the DocumentEntry for a patient where all versions have deprecated status.

> **Note:**
>
> This parameter value is only applicable to the FindDocument query type of Registry Stored Query (ITI-18).

AuthorPerson query

This section describes how to use the AuthorPerson query.

The `$XDSDocumentEntryAuthorPerson` query parameter value is used in a case-sensitive manner in Registry Stored Query (ITI-18) to retrieve matching document entries. To query for document entries with AuthorPerson value in a case-insensitive manner, a new parameter `$orcl.caseInsensitive.DocumentEntryAuthorPerson` is added

Large numbers of DocumentEntry objects

This section describes how to use queries to limit the number of DocumentEntry objects returned.

Registry queries executed without any filters retrieve all document entries of a patient.

You must ensure to design client document consumer actor to query with filter conditions or parameters. However, using a filter may still result in large document entries and hence, Oracle recommends that you use the following queries instead of one large query retrieving all DocumentEntry metadata.

Execute large query with `returnType="ObjectRef"`, which returns all document entries Object References or Universally Unique Identifier (UUID)s. This query executes faster compared to one with `returnType="LeafClass"`, which returns all metadata (XML structure).

Subsequent queries can use limited number of UUIDs to query document entries metadata depending on the page size.

For example:

- Executing the FindDocuments query type with returnType="ObjectRef" returns a large number of ObjectRefs (UUIDs).
- Executing subsequent GetDocuments query type with returnType="LeafClass" with a limited number of UUIDs from the list returns document entries metadata (XML structure) to be processed and displayed on one page.

ITI-62 transaction simulation

This section describes how Health Record Locator simulates ITI-62 transactions.

Health Record Locator simulates an ITI-62 transaction without committing the changes or deleting the metadata entry. To simulate this, add the SOAP header SimulateTransaction to the ITI-62 request. For example:

```
<custom:SimulateTransaction
xmlns:custom="urn:oracle:hsgbu:him:hrl:customheader">yes</
custom:Simula teTransaction>
```

Custom query parameter $orcl.order.by

Health Record Locator supports an optional custom parameter, `$orcl.order.by`, for ITI-18 Find Documents query, which sorts the query result on a specific attribute. The parameter should have single value in the `<sortfieldname>:<a|d>` format, where **a** is for ascending order and **d** is for descending order. `<sortfieldname>` can be one of the following:

- `creationTime`
- `languageCode`
- `mimeType`

- `serviceStartTime`
- `serviceStopTime`
- `size`

For example:

```
<Slot name="$orcl.order.by">
    <ValueList>
        <Value>'creationTime:a'</Value>
    </ValueList>
</Slot>
```

4

Security configuration issues

This section describes security configuration issues you must consider when implementing Health Record Locator.

- General security principles
  This section describes fundamental security principles for using any application securely.
- Configure strong database passwords
  This section discusses configuring strong passwords on the database.
- Follow the principle of least privilege
  This section describes the principle of least privilege.
- Disable Telnet service
  This section describes why you should not use the Telnet service.
- Disable other services
  This section explains why you should disable other unused services.
- Design multiple layers of protection
  This section describes the need for multiple layers of protection.
- Use SSL
  This section describes why SSL is a good choice for Health Record Locator.

General security principles

This section describes fundamental security principles for using any application securely.

General security principals include:

- **Keep software up-to-date**: Keep all software versions and patches up-to-date.
- **Keep up-to-date on the latest security information and critical patches:** 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.
- **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.

Configure strong database passwords

This section discusses configuring 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 ADT, HRLCORE, and LOG.
- No password for the database listener. Oracle recommends that you do not configure a password for the database listener as this enables 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

This section describes 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, use the following default file permissions in a Unix environment:

- 740 for executable files
- 640 for regular files

Disable Telnet service

This section describes why you should not use the Telnet service.

The Health Record Locator 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, disable Telnet in favor of Secure Shell (SSH). Disabling Telnet protects your system security.

Disable other services

This section explains why you should disable other unused services.

In addition to not using Telnet, the Health Record Locator 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 Health Record Locator standard configuration. If you are not using these services for other applications, 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.

Design multiple layers of protection

This section describes the need for 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.

Use SSL

This section describes why SSL is a good choice for Health Record Locator.

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

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

A

DocumentEntry status change

This appendix provides information about DocumentEntry status changes.

- DocumentEntry association type diagrams
  This section provides diagrams of supported association types.

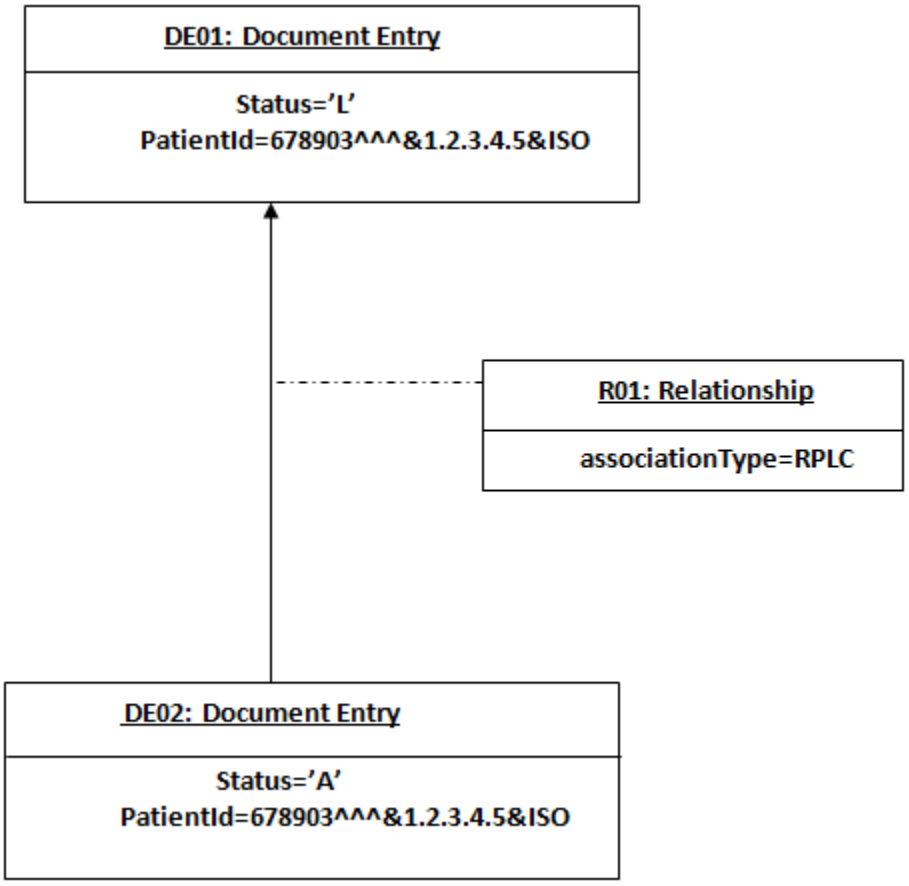
DocumentEntry association type diagrams

This section provides diagrams of supported association types.

The following diagrams illustrate the status values for an existing DocumentEntry object and new DocumentEntry object after a document relationship is applied to an existing DocumentEntry.

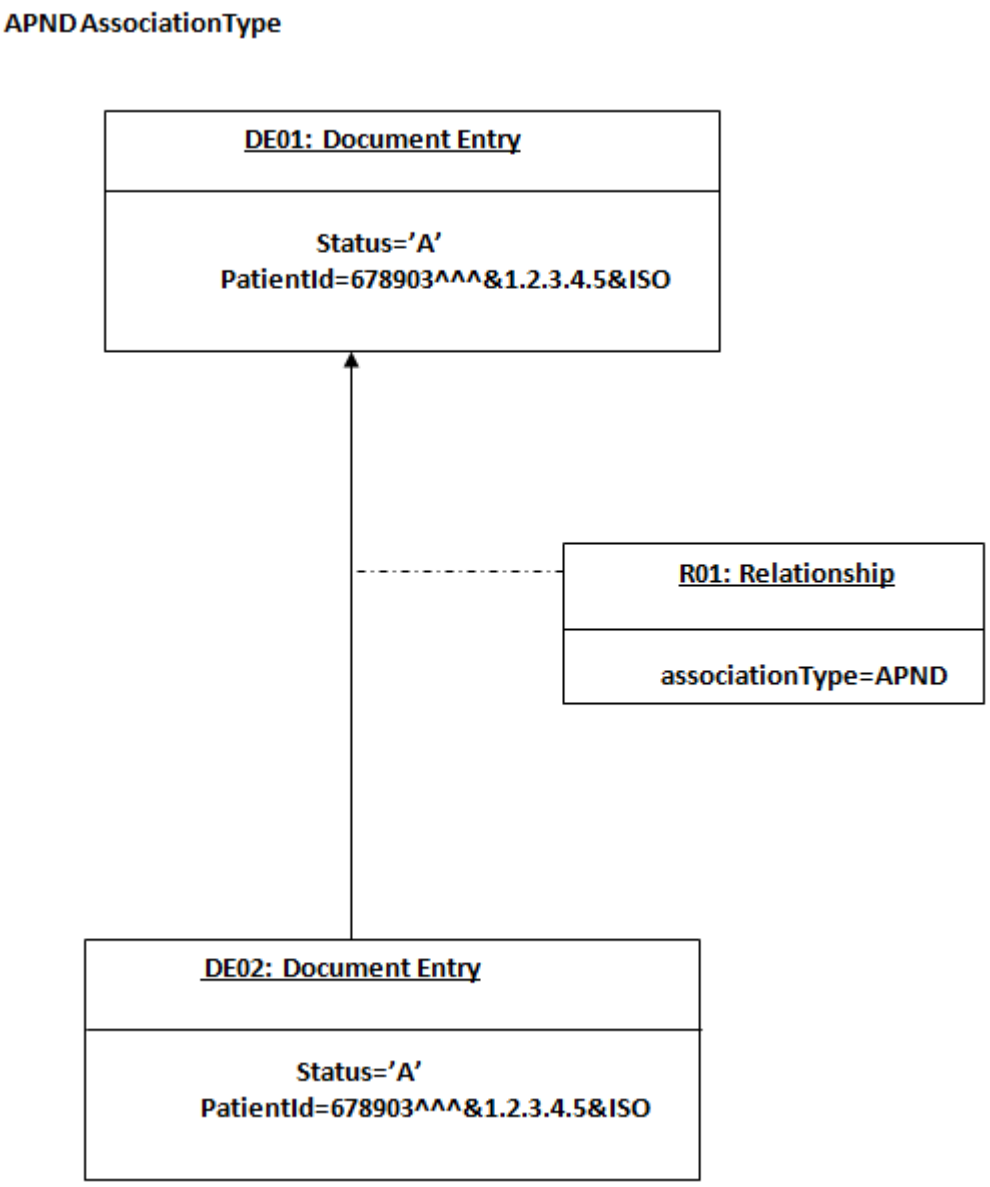
> **Note:**
>
> The Status column in the Doc_Entry database table indicates the status of that particular DocumentEntry object.

Figure A-1 RPLC Association Type



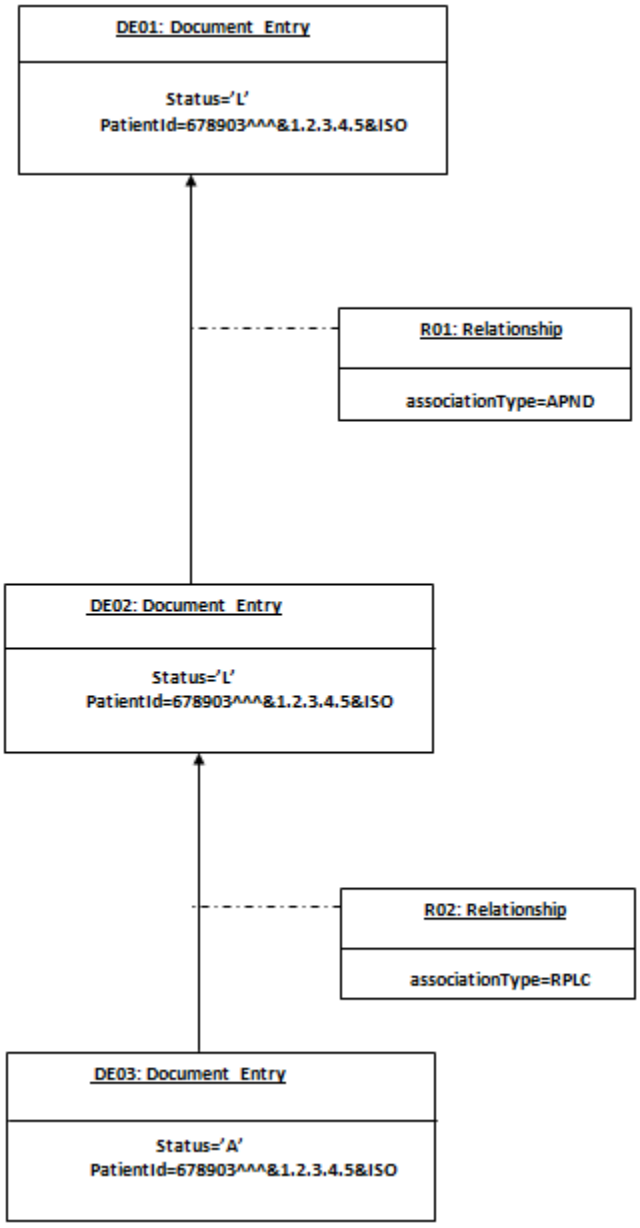
Where DE01 is an existing document entry.

Figure A-2 APND Association Type



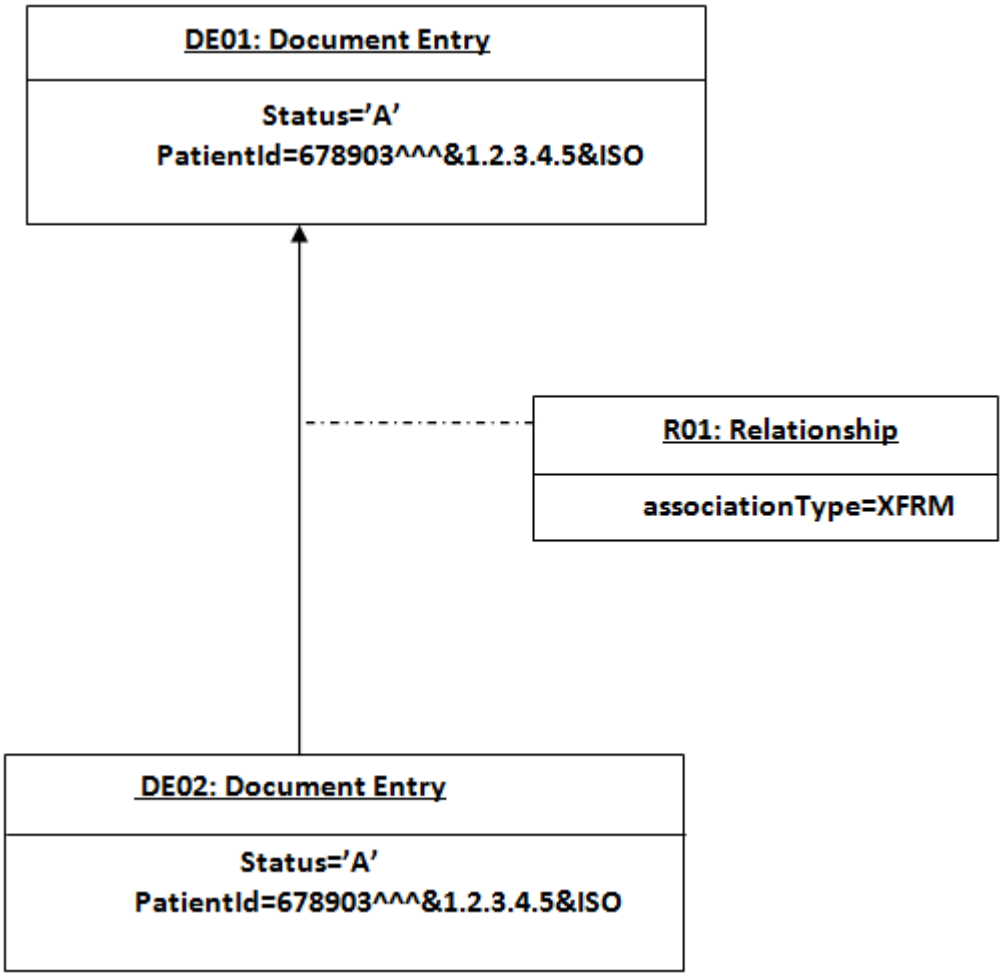
Where DE01 is an existing document entry.

Figure A-3 Document Replacing an Addendum (APND) of an Existing DocumentEntry



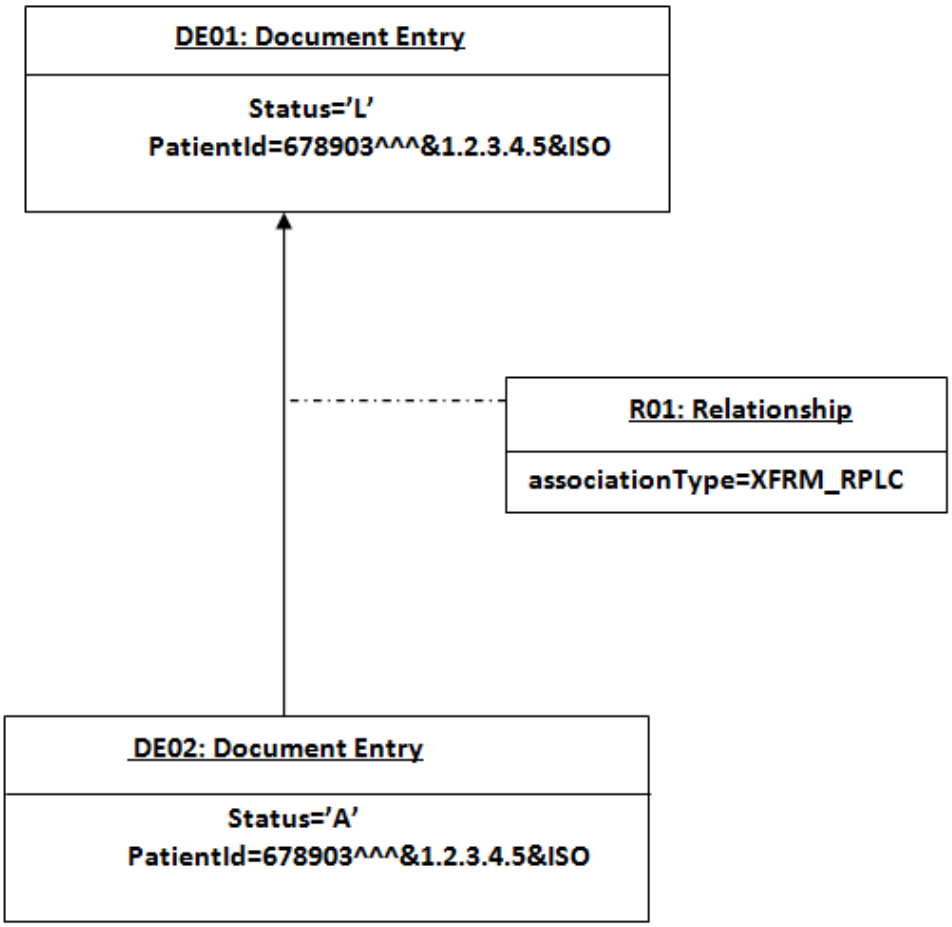
Where DE01 and DE02 are existing document entries.

Figure A-4 XFRM Association Type



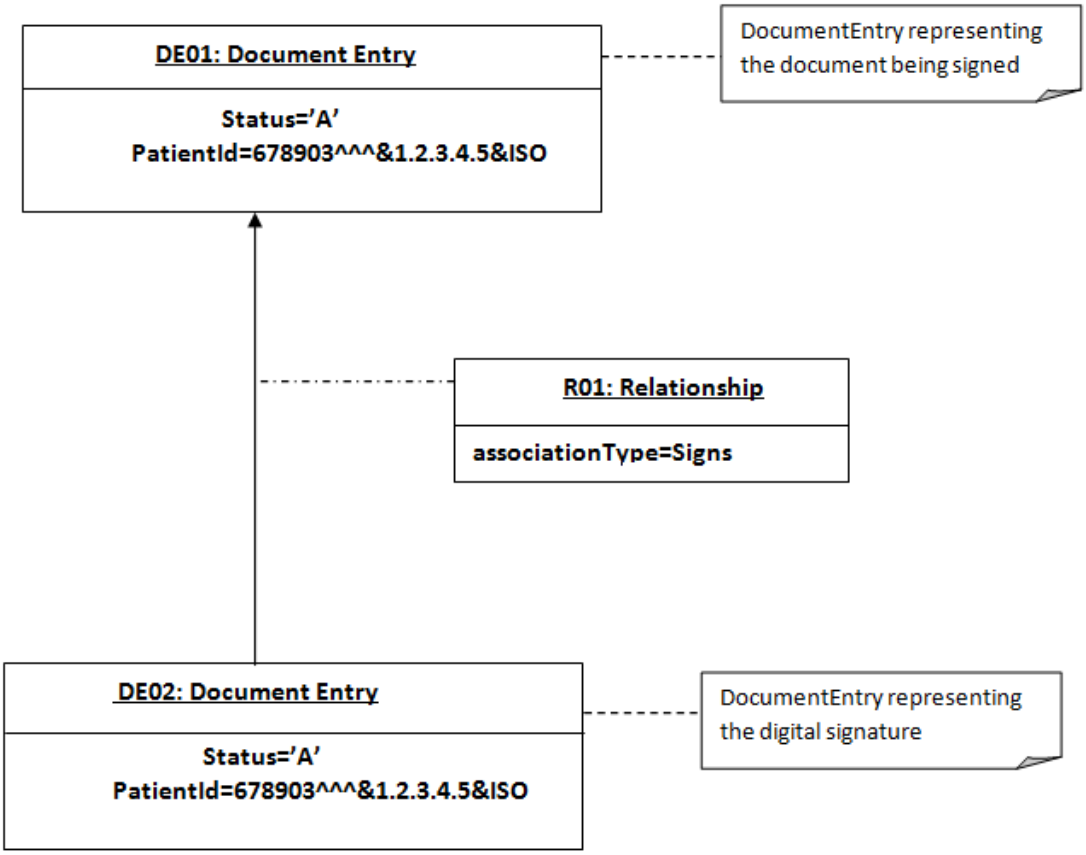
Where, DE01 is an existing document entry.

Figure A-5 XFRM_RPLC Association Type



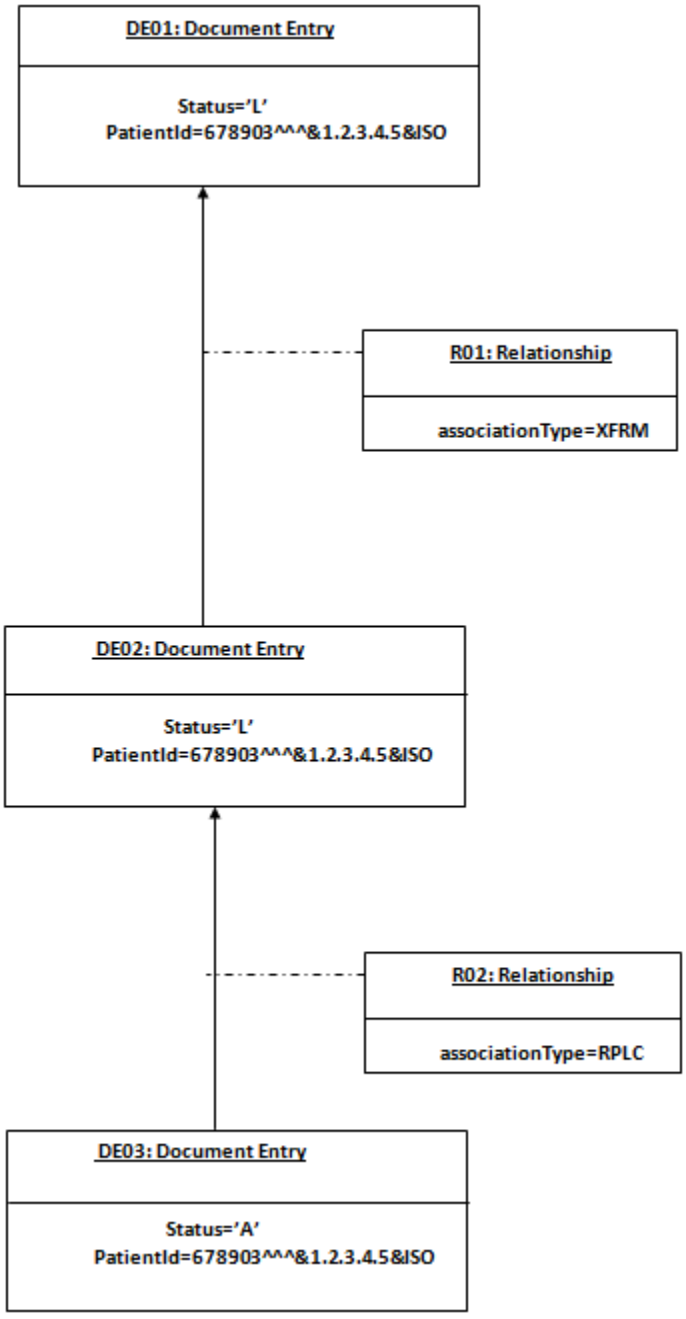
Where, DE01 is an existing document entry.

Figure A-6 Signs Association Type



Where, DE01 is an existing document entry.

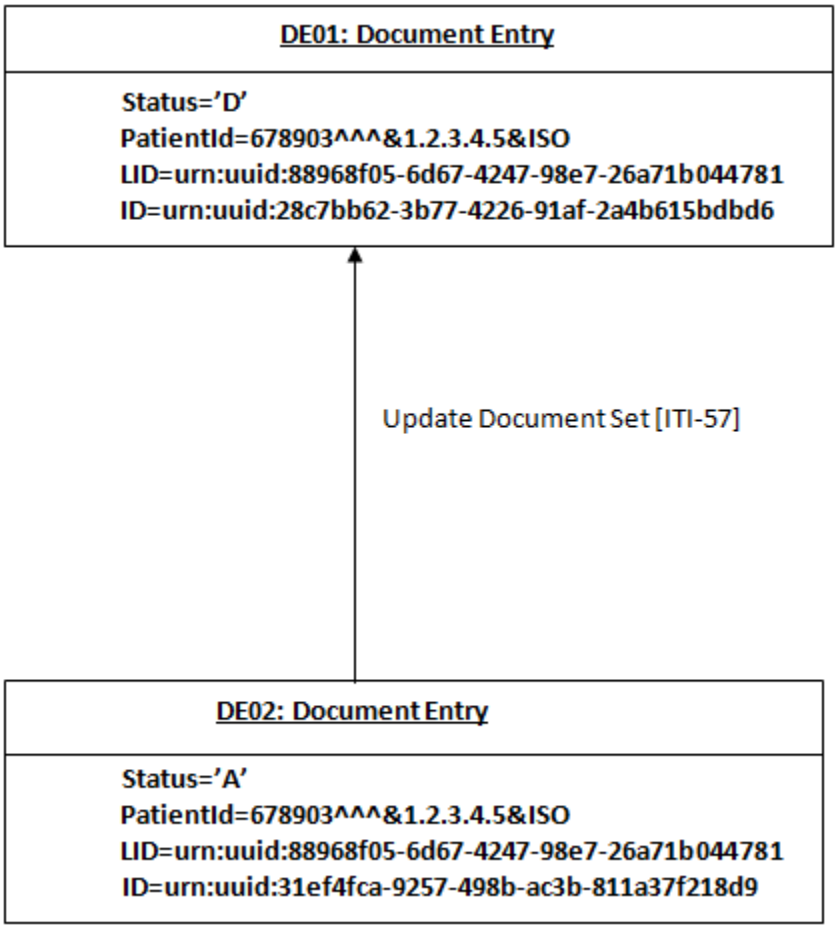
Figure A-7 Document Replacing a Transformation (XFRM) of an Existing DocumentEntry



Where, DE01 and DE02 are existing document entries.

The following diagram depicts the status of both the previous version of DocumentEntry and the updated DocumentEntry when an Update Document Set [ITI-57] transaction is executed against Health Record Locator.

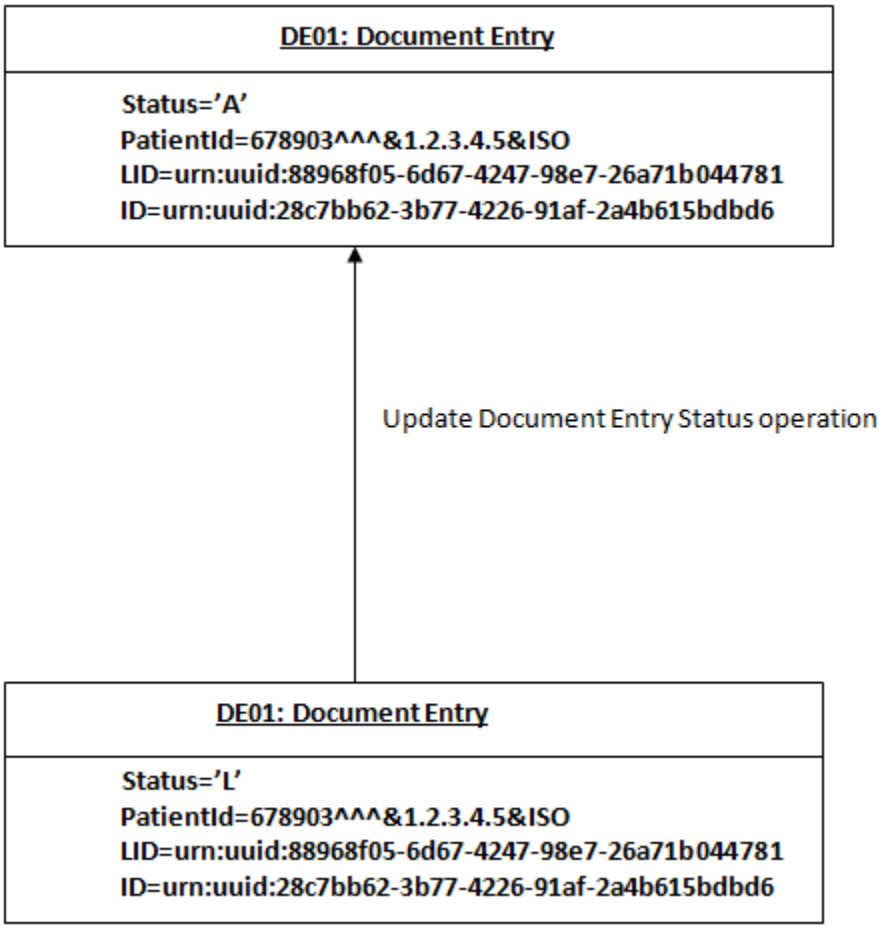
Figure A-8 DocumentEntry



Where, DE01 is an existing document entry version and DE02 is a new version created.

The following diagram depicts the status of a DocumentEntry when the Update DocumentEntry Status operation is executed against Health Record Locator.

Figure A-9 Status of a Document Entry



Where, DE01 is an existing document entry version. A new version is not created.

- Whenever a Register Document Set-b[ITI-42] transaction is executed against Health Record Locator, the newly created DocumentEntry or Folder is set with the status 'A'.
- Whenever an Update Document Set [ITI-57] transaction is executed against Health Record Locator, the previous version of DocumentEntry or Folder is set with the status 'D' while the new version of DocumentEntry or Folder is set with status 'A'.