



Understanding the Sun GlassFish ESB PIX/PDQ Manager



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 821-0867-11
December 2009

Copyright 2009 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED “AS IS” AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Contents

Working With the PIX/PDQ Manager	5
PIX/PDQ Manager Overview	6
About the IHE Technical Framework	6
PIX/PDQ Manager Architecture	8
PIX/PDQ Processing Flow	10
Patient Identity Feed	11
Patient Demographics Query	12
PIX Query	14
PIX Update Notification	15
PIX/PDQ Management and Monitoring Consoles	17
PIX Console	17
Master Index Data Manager	20
About the PIX/PDQ Core Projects	20
PIXPDQAuditService	21
PIXPDQCommon	21
PIXPDQDomainLookupEJB	21
PIXPDQMasterIndex	21
PIXPDQMuralFacadeEJB	22
PIXPDQUtilities	22
PIXUpdateNotification	22
About the Audit Projects (ATNA)	22
About the HL7 V2 Projects	23
IHE_PIX_PDQ_V2_Direct	23
IHE_PIX_Update_V2_Direct	24
IHE_PIX_PDQ_V2_Direct_ca	24
IHE_PIX_Update_V2_Direct_ca	25
About the HL7 V3 Projects	26
PDQ_HL7V3_Direct	26

PIX_HL7V3_Direct	27
PIX_HL7V3_UpdateNotification_Direct	28
PIXPDQ_HL7V3_Direct_ca	28
HL7 V3 Messaging	29

Working With the PIX/PDQ Manager

This topic provides instructions to help you understand and use the PIX/PDQ Manager, which is installed as part of the Healthcare Pack. The Healthcare Pack includes the following components:

- Sun Master Index
- HL7 Binding Component
- PIX/PDQ Manager

To access all the GlassFish ESB documentation in PDF and HTML format, go to the GlassFish ESB page on docs.sun.com (<http://docs.sun.com/app/docs/prod/gf.entsvc.bus#hic>).

What's in This Document

This document includes the following topics:

- “PIX/PDQ Manager Overview” on page 6
- “PIX/PDQ Processing Flow” on page 10
- “PIX/PDQ Management and Monitoring Consoles” on page 17
- “About the PIX/PDQ Core Projects” on page 20
- “About the Audit Projects (ATNA)” on page 22
- “About the HL7 V2 Projects” on page 23
- “About the HL7 V3 Projects” on page 26

Additional Information

The following provide reference information along with information and instructions for using the HL7 Binding Component, HTTP Binding Component, Sun Master Index, the BPEL Service Engine, and the PIX/PDQ Manager. These are the key components of the PIX/PDQ system.

- *Working With the PIX/PDQ Manager*
- *BPEL Designer and Service Engine User's Guide*
- *Using the HL7 Binding Component*

- *Sun Master Index User's Guide*

The Sun Master Index User's Guide includes links to the complete documentation set for Sun Master Index, which includes information on configuring master index, working the match and standardization engines, and using the profiling, cleansing, and load tools provided with Sun Master Index.

PIX/PDQ Manager Overview

With the current focus on improving the exchange of electronic healthcare information, the ability to share information stored in various healthcare systems and healthcare organizations is becoming a requirement. Integrating healthcare systems makes data more easily accessible to providers and participants, ensures that the data is accurate and current, and provides the basis for the exchange of electronic health data. The Health Information Technology for Economic and Clinical Health (HITECH) Act encourages the adoption of electronic healthcare records by all providers and also defines penalties for breaches of patient health information.

Sun's PIX/PDQ Manager provides a flexible solution to healthcare integration needs in the form of a lightweight enterprise service bus (ESB) that can be easily customized and extended to meet your integration needs. The processing logic in the PIX/PDQ Manager is based on the guidelines and standards put forth by Integrating the Healthcare Enterprise (IHE). This assures compatibility with other vendors and healthcare organizations, making it easier to integrate now and in the future.

The PIX/PDQ Manager forms a specialized implementation of HL7 messaging that facilitates patient health information exchange, and supports both HL7 v2 and HL7 v3 messaging. It leverages the advanced standardization and matching algorithms of Sun Master Index to cross-reference and uniquely identify the patients in your healthcare organization. Master Index provides a single complete view of the participants in your healthcare system and is able to quickly reconcile which information is associated with which patient, allowing you to quickly create a complete medical history.

The PIX/PDQ Manager processes messages based on the IHE frameworks, which define how to process HL7 messages using existing standards when available. In compliance with these frameworks, the PIX/PDQ Manager generates and maintains an audit repository of all events processed by the manager, and also maintains a trace record of how each message was processed through the system and by which components. Information about the state of the PIX/PDQ Manager components is provided by a common logging, alerting, error handling, and reporting mechanism. The PIX/PDQ Manager provides a monitoring and management tool where you can monitor the audit repository, log messages, and message traces.

About the IHE Technical Framework

Integrating the Healthcare Enterprise (IHE) is an organization whose aim is to improve how electronic patient information is shared among healthcare systems and, by doing so, to make

sure that current and accurate data is readily available to both patients and healthcare professionals. IHE has developed technical frameworks that define how to process healthcare events, how data is shared, how security is handled, how audit records are generated, and how components interact with one another. The frameworks are made up of integration profiles that provide specifications of how each type of event is processed and how the audit message should be generated for each type of event. The profiles also define standards for security, communication, and time synchronization. These profiles are designed to ensure that data is transmitted securely and accurately among systems, and that data handling is coordinated according to communication and security standards. Having this common framework gives the various participants in a healthcare system a common base for integrating disparate systems.

Sun's PIX/PDQ Manager focuses on the following IT Infrastructure profiles of the IHE framework:

- **Audit Trail and Node Authentication (ATNA):** This profile describes authenticating systems and transmitting audit events to a repository. The PIX/PDQ Manager provides a complete audit log of events processed through the PIX/PDQ system in the prescribed ATNA format. You can view a complete log of audit events on the PIX Console. The format varies according to the type of event being processed.
- **Patient Identifier Cross Referencing (PIX):** This profile describes cross-referencing patient identifiers between the various systems that register and store healthcare information. This allows domains to correlate information about a patient from multiple sources that use different identifiers. Sun Master Index maintains a complete cross-reference of patient identifiers using probabilistic matching to identify records. When updates are made to key information in the master index database, Master Index generates PIX outbound notifications, which can then be broadcast to interested domains. The PIX/PDQ Manager supports PIX queries for patient identifiers associated with a given identifier.
- **Patient Demographics Query (PDQ):** This profile describes system queries against a central repository of patient information to obtain a patient's demographic information, allowing multiple domains to query a central source of patient information to retrieve demographic data for matching patients. Sun Master Index maintains this central repository for the PIX/PDQ Manager, and processes patient demographic queries.

Each healthcare participant registers identifiers for patients in their own computer system, also known as a *domain* in the IHE framework. Participants maintain control over their own domain's index, but sometimes need access to information in a different domain. They can access the central repository of the PIX/PDQ Manager to find additional local identifiers for their patients. Domains can also be automatically notified when other systems update patient information.

As per IHE standards, Sun's PIX/PDQ Manager cross-references patient IDs (PIX) and supports patient demographic queries (PDQ). It also supports PIX queries for local IDs that are associated with a given local ID from a domain and patient identity feeds, such as when a record

is added or updated in a domain. The PIX/PDQ Manager can broadcast notifications to interested domains when certain updates occur in the master patient index.

The Audit Trail and Node Authentication (ATNA) profile defines how audit messages are generated and formatted. The audit repository helps ensure that patient information remains confidential, data integrity is maintained, and users are accountable for accessing data. The Sun PIX/PDQ Manager supports the ATNA profile by maintaining an audit repository that stores information about each event processed through the system, including the source type and source ID, event type, event ID, event action, and event outcome. It also stores the date and time the event occurred and the date and time it was received by the PIX/PDQ system.

PIX/PDQ Manager Architecture

Sun's PIX/PDQ Manager uses various components of GlassFish ESB such as HTTP, JMS, and HL7 Binding Components; the BPEL Service Engine; Composite Applications; Java EE EJBs; and Sun Master Index. The solution includes the following GlassFish ESB components:

- **Sun Master Index**
Maintains a central repository of patient information from all domains that share data in the PIX/PDQ system, and also provides the cross-referencing, query, standardization, and matching functions to support patient identity resolution for PIX and PDQ requests. The master index can detect when two records from the same or different domains are referring to the same person, and can reconcile the two records to determine the best information to use.
- **PIX Console**
Provides a view into the ATNA audit repository and PIX/PDQ processing log, and traces how a message is processed through the PIX/PDQ Manager. The console also provides tools to manage and maintain domains, subscriptions to outbound notifications, and application configurations and variables.
- **BPEL Service Engine**
Defines the processing flow and business logic of messages through the PIX/PDQ system, including routing and transforming data, mapping data fields, and calling the appropriate functions from other PIX/PDQ components. The BPEL processes also write the required data to the ATNA audit repository, and are responsible for generating HL7 ACK/NACK responses. Each BPEL process is specific to either HL7 v2 or HL7 v3.
- **HL7 Binding Component**
Supports the HL7 messaging structure up to version 2.6 and defines the communication protocols for connecting to the HL7 messaging systems. The HL7 Binding Components is configured to use the Minimal Lower Layer Protocol (MLLP) version 1 to connect to external systems.

- **HTTP Binding Component**

Supports the HL7 v3 messaging structure and defines the communication protocols for connecting to the HL7 messaging systems. The HTTP Binding Components uses the SOAP protocol to connect with external systems. The PIX/PDQ Manager provides for and supports both SOAP versions 1.1 and 1.2.

- **JMS Binding Component:** Supports outbound notifications from the master patient index, which are published to JMS topics and then distributed to all subscribed domains.

The following figure illustrate the interaction of the above components in the PIX/PDQ Manager.

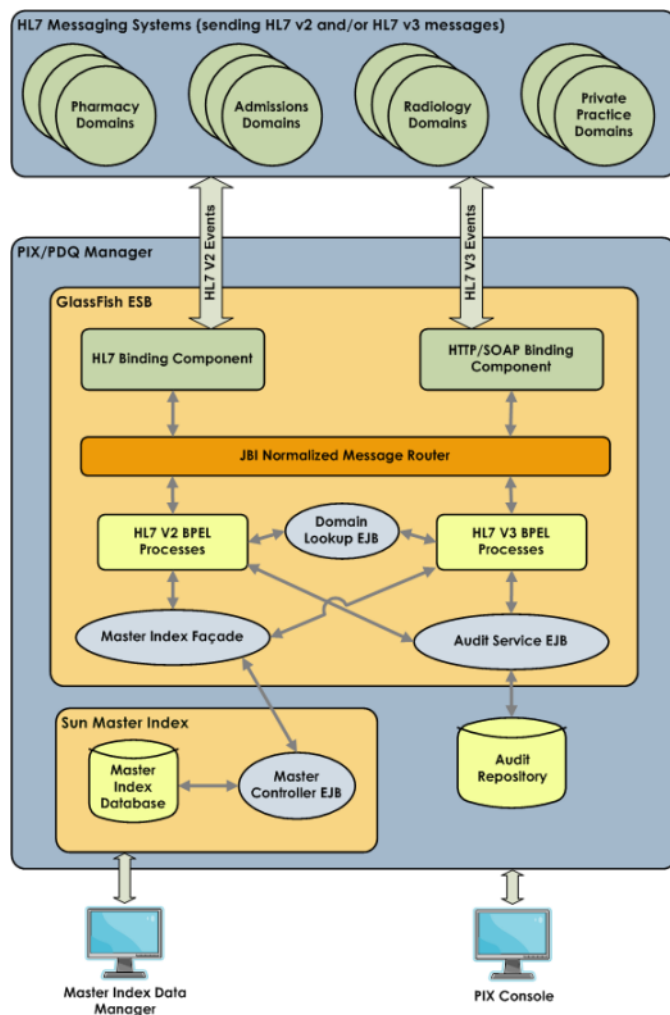


FIGURE 1 PIX/PDQ Manager Architecture

PIX/PDQ Processing Flow

Data enters the PIX/PDQ system through either the HL7 Binding Component for v2 messages or the HTTP Binding Component for HL7 v3 messages. The data is then mapped, transformed, and routed through the appropriate BPEL process so it can be understood by the master index. Depending on the type of HL7 event, the BPEL process calls the domain lookup interface and then the Master Index facade interface which interacts with the master index itself. The message is processed through the master index system as a query, a new record, an update to an existing

record, or a merge transaction. When the results are returned to the BPEL process, the BPEL process constructs an HL7 acknowledgment (ACK) and sends a reply through either the HL7 or HTTP Binding Component.

You can follow the path that each message takes through the PIX/PDQ Manager by viewing the message trace on the PIX Console. The trace reports various checkpoints throughout the transaction, and displays the order in which it was processed by each component.

The Sun PIX/PDQ Manager supports the following IHE event types. The processing flow is different for each type.

- “Patient Identity Feed” on page 11
- “Patient Demographics Query” on page 12
- “PIX Query” on page 14
- “PIX Update Notification” on page 15

Patient Identity Feed

A patient identity feed event is triggered when a record is added, updated, or merged in a source system. The new information is then fed to the PIX/PDQ Manager and specifically to the Master Index for cross-referencing and identification. Once master index processing is complete, a message is sent out to the domains that subscribe to master index updates. An event is also entered in the ATNA audit log.

Patient Identity Feed Processing Flow

Patient identity feeds are processed as follows. This can be seen in the figure [Figure 1](#).

1. An event occurs in an HL7 messaging domain that creates a new record or updates or merges existing records.
2. The message is received by the HL7 Binding Component if it is HL7 v2, or by the HTTP Binding Component if it is HL7 v3.
3. The message is processed through the Normalized Message Router to one of the following BPEL processes, depending on which version of HL7 is being processed: IHEPixPdqv2BpelProcess (v2) or bpelPIXManager (v3).
4. The BPEL process maps the data and routes it to the MDM Facade EJB, which then sends it to the Master Index for processing.
5. The Master Index compares the data against its existing records and updates the database accordingly.

The action taken by the Master Index is based on whether the new information constitutes a new patient record, updates an existing patient record, or merges two existing patient records. This is determined by the results of the matching algorithm. If a new record is added, the master index might mark the record as a potential duplicate of existing records if the matching probabilities fall within a defined range.

6. The BPEL process then generates an ACK or NACK response, which is sent back out to the originating domain.
7. The BPEL process generates the audit record of the transaction, which is stored in the ATNA repository.

Patient Identity Feed HL7 Message Types

In the HL7 v2 IHE framework, this type of event is known as an ITI-8 transaction. It is triggered by the following HL7 v2 events:

- **A01:** Inpatient admission
- **A04:** Outpatient registration
- **A05:** Inpatient preadmission
- **A08:** Update to an existing patient record
- **A40:** Patient record merge

In the HL7 v3 IHE framework, this type of event is known as an ITI-44. It is triggered by the following HL7 v3 events:

- **PRPA_TE201301UV02:** Patient registry record added (triggers a PRPA_IN201301UV02 message)
- **PRPA_TE201302UV02:** Patient registry record revised (triggers a PRPA_IN201302UV02 message)
- **PRPA_TE201304UV02:** Patient registry duplicates resolved (triggers a PRPA_IN201304UV02 message)

For audit purposes, these events are all identified as “Patient Record” events, with an event ID of 110110.

Patient Demographics Query

A patient demographics query event is triggered when a consumer needs to select a patient based on demographic data. The consumer requests information about patients that match the data from the master patient index. The query message provides a set of demographic data, which the master patient index compares and matches against existing records in the database. The master patient index generates a response with the demographic data for matching patients.

Patient Demographics Query Processing Flow

Patient demographics queries are processed as follows. This can be seen in the figure [Figure 1](#).

1. A request is sent by an HL7 messaging domain for patients whose demographic information closely matches the demographic data sent in.
2. The message is received by the HL7 Binding Component if it is HL7 v2, or by the HTTP Binding Component if it is HL7 v3.
3. The message is processed through the Normalized Message Router (NMR) to one of the following BPEL processes, depending on which version of HL7 is being processed: IHEPixPdqv2BpelProcess (v2) or bpelPDQManager (v3).
4. The BPEL process maps the data and routes it to the MDM Facade EJB, which then sends it to the Master Index for processing.
5. The Master Index compares the input data against existing records in the database. Any matches are returned to the BPEL process.
6. The BPEL process then sends the patient information back through the NMR and Binding Components to the originating domain, and generates an ACK or NACK response, which is also sent back out to the originating domain.
7. The BPEL process generates the audit record of the transaction, which is stored in the ATNA repository.

Patient Demographics Query HL7 Message Types

In the HL7 v2 IHE framework, this type of event is also called an ITI-21 transaction. It is triggered by the following HL7 v2 event. For audit purposes, it is identified as a “Query” event with an event ID of 110112.

- **Q22:** Find candidates

The master patient index response to the above request is the following HL7 v2 event:

- **K22:** Find candidates response

In the HL7 v3 IHE framework, this type of event is also called an ITI-47 transaction. It is triggered by the following HL7 v3 event. For audit purposes, it is identified as a “Query” event with an event ID of 110112.

- **PRPA_TE201305UV02:** Find candidates query (triggers a PRPA_IN201305UV02 message)

The master patient index response to the above request is the following HL7 v3 event:

- **PRPA_IN201306UV02:** Find candidates query (triggers a PRPA_IN201306UV02 message)

PIX Query

A PIX query is triggered when one of the domains in the PIX/PDQ system requests a list of patient identifiers that are associated with a specific patient identifier. This request is processed by the Master Index, which responds with a list of associated patient identifiers if any exist.

PIX Query Processing Flow

PIX queries are processed as follows. This can be seen in the figure [Figure 1](#).

1. A request is sent by an HL7 messaging domain for any identifiers associated with the given patient identifiers.
2. The message is received by the HL7 Binding Component if it is HL7 v2, or by the HTTP Binding Component if it is HL7 v3.
3. The message processed through the Normalized Message Router (NMR) to one of the following BPEL processes, depending on which version of HL7 is being processed: IHEPixPdqv2BpelProcess (v2) or bpelPIXManager (v3).
4. The BPEL process maps the data and routes it to the MDM Facade EJB, which then sends it to the Master Index for processing.
5. The Master Index cross references the given patient identifier against existing information in the master index database, and either returns a list of identifiers or null, if none were found.
6. The BPEL process then sends the list of identifiers back through the NMR and Binding Components to the originating domain, and generates an ACK or NACK response, which is also sent back out to the originating domain.
7. The BPEL process generates the audit record of the transaction, which is stored in the ATNA repository.

PIX Query HL7 Message Types

In the HL7 v2 IHE framework, this type of event is known as an ITI-9 transaction. It is triggered by the following HL7 v2 event, and is identified as a “Query” event with an event ID of 110112:

- **Q23:** Get corresponding identifiers

The master patient index response to the Q23 request is the following HL7 v2 event:

- **K23:** Corresponding patient identifiers

In the HL7 v3 IHE framework, this type of event is known as an ITI-45 transaction. It is triggered by the following HL7 v3 event. For audit purposes, it is identified as a “Query” event with an event ID of 110112.

- **PRPA_TE201309UV02:** Patient registry get identifiers query (triggers a PRPA_IN201309UV02 message)

The master patient index response to the above request is the following HL7 v3 event:

- **PRPA_IN201310UV02:** Patient registry get identifiers query response

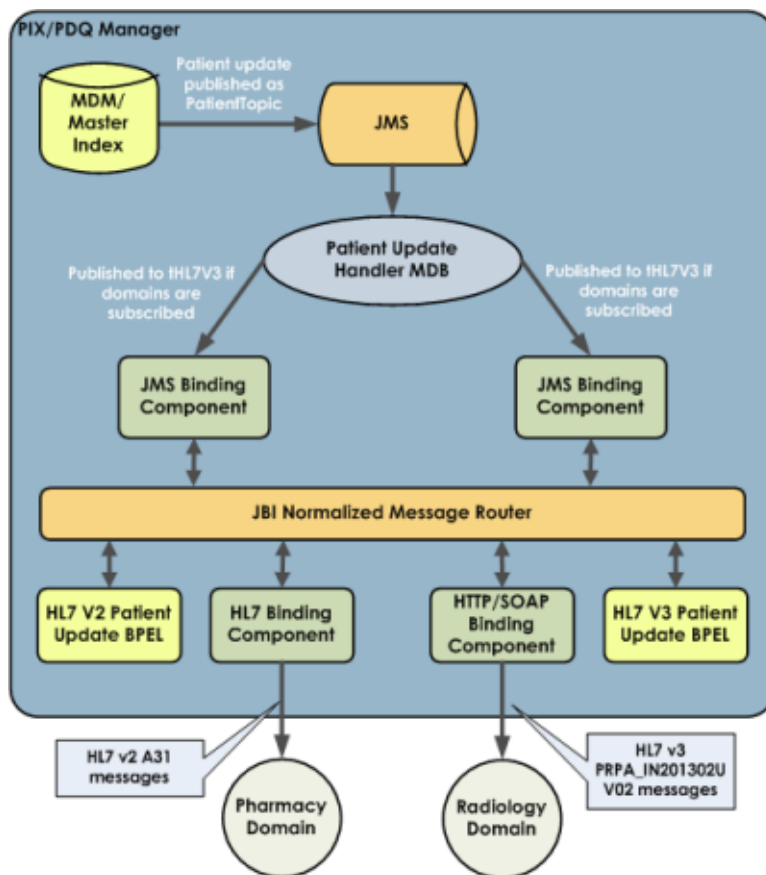
PIX Update Notification

A PIX update notification event occurs when key data in a patient record is updated by a domain in the master index, which then broadcasts the update to a JMS topic to make it available to all interested domains. This type of message is unlike the others in that the event originates in the master index to be distributed to the HL7 domains, and it uses JMS topics to distribute the messages.

PIX Update Notification Processing Flow

PIX update notifications are processed as follows:

1. Sun Master Index receives an update from a domain, and publishes the update to a JMS topic, PatientTopic.
2. The PIX Update Notification MDB receives the update from PatientTopic and determines if any domains are subscribed to updates from the originating domain, and if so, if they require HL7 v2 or v3 messages. This information is found in the PIX/PDQ database tables.
3. For subscribing HL7 v2 domains, the MDB writes the outbound message to the tHL7V2 topic; for subscribing HL7 v3 domains, the MDB writes the outbound message to the tHL7V3 topic.
4. The update notification BPEL process subscribes to tHL7V2 in the HL7 v2 projects, and to tHL7v3 in the HL7 v3 projects. Messages written to these topics are processed by the appropriate BPEL process and mapped to the required HL7 format.
5. The audit record of the outbound notification is generated and stored in the ATNA repository.
6. HL7 v2 updates are sent out through the HL7 Binding Component to the subscribing HL7 v2 domains. HL7 v3 updates are sent out through the HTTP Binding Component to the subscribing HL7 v3 domains.



PIX Update Notification HL7 Message Types

In the HL7 v2 IHE framework, a PIX outbound notification is also called an ITI-10 transaction. An update to the patient cross-reference triggers the following HL7 v2 event. For audit purposes, it is identified as a “Patient Record” event with an event ID of 110110.

- **A31:** Update person information

In the HL7 v3 IHE framework, this type of event is also called an ITI-46 transaction. An update to the patient cross-reference triggers the following HL7 v3 event. For audit purposes, it is identified as a “Patient Record” event with an event ID of 110110.

- **PRPA_IN201302UV02:** Patient registry record revised

PIX/PDQ Management and Monitoring Consoles

The PIX/PDQ system provides two web-based management and monitoring consoles that allow you to monitor system processes, audit records, and log messages, as well as manage and maintain the data that is stored in the master index database. The following topics provide additional information about each console.

- [“PIX Console” on page 17](#)
- [“Master Index Data Manager” on page 20](#)

PIX Console

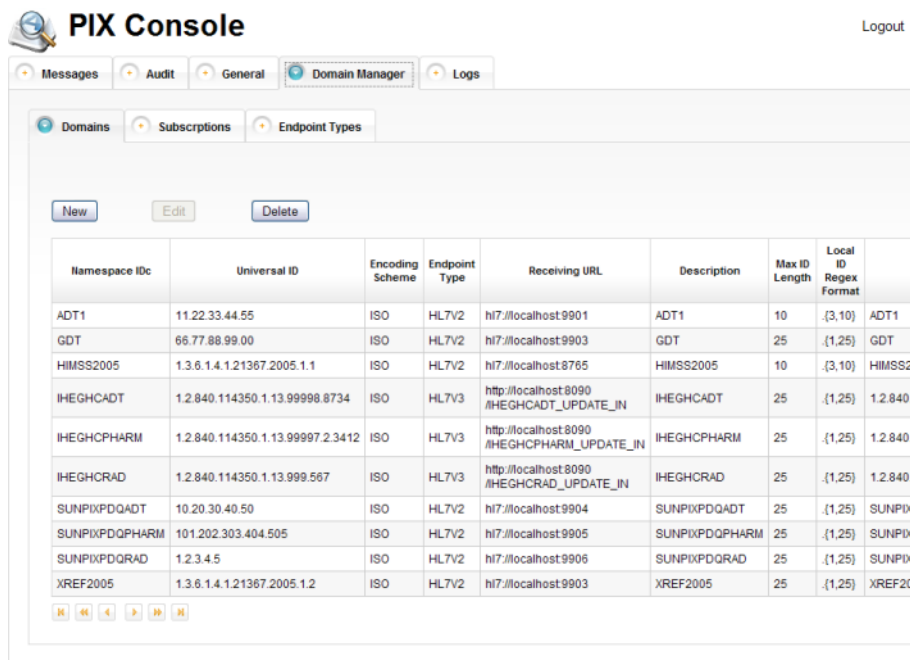
The PIX Console provides you with a variety of useful information and tools that allow you to monitor end-to-end processing of the messages that are processed through the PIX/PDQ Manager. The PIX Console helps fulfill IHE requirements by providing a view into the ATNA audit repository and the server log messages that pertain to the PIX/PDQ Manager. You can also view information about a message at any point in its processing, along with the components that processed it.

The console provides these primary features:

- A central view of request and response messages
- A message tracking center
- A PIX/PDQ ATNA audit log viewer with search and filter capabilities
- The ability to update application variables and configurations
- A Domain Manager for viewing and maintaining information about the external systems that share information in the PIX/PDQ system
- A central view of server log messages specific to the PIX/PDQ system

The PIX console includes the following components:

- [“Message Viewer” on page 18](#)
- [“Audit Viewer” on page 18](#)
- [“Application Variable and Configuration Manager” on page 19](#)
- [“Domain Manager” on page 19](#)
- [“Log Message Viewer” on page 19](#)



The screenshot shows the PIX Console interface. At the top, there's a navigation bar with tabs: Messages, Audit, General, Domain Manager (selected), and Logs. Below this, there's a sub-navigation bar with tabs: Domains (selected), Subscriptions, and Endpoint Types. In the Domains tab, there are buttons for New, Edit, and Delete. Below these buttons is a table listing various domains.

Namespace IDc	Universal ID	Encoding Scheme	Endpoint Type	Receiving URL	Description	Max ID Length	Local ID Regex Format	
ADT1	11.22.33.44.55	ISO	HL7V2	hi7://localhost9901	ADT1	10	{3,10}	ADT1
GDT	66.77.88.99.00	ISO	HL7V2	hi7://localhost9903	GDT	25	{1,25}	GDT
HIMSS2005	1.3.6.1.4.1.21367.2005.1.1	ISO	HL7V2	hi7://localhost8765	HIMSS2005	10	{3,10}	HIMSS2
IHEGHCADT	1.2.840.114350.1.13.99998.8734	ISO	HL7V3	http://localhost8090/IHEGHCADT_UPDATE_IN	IHEGHCADT	25	{1,25}	1.2.840.
IHEGHCPHARM	1.2.840.114350.1.13.99997.2.3412	ISO	HL7V3	http://localhost8090/IHEGHCPHARM_UPDATE_IN	IHEGHCPHARM	25	{1,25}	1.2.840.
IHEGHCRAD	1.2.840.114350.1.13.999.567	ISO	HL7V3	http://localhost8090/IHEGHCRAD_UPDATE_IN	IHEGHCRAD	25	{1,25}	1.2.840.
SUNPIXPDQADT	10.20.30.40.50	ISO	HL7V2	hi7://localhost9904	SUNPIXPDQADT	25	{1,25}	SUNPIX
SUNPIXPDQPHARM	101.202.303.404.505	ISO	HL7V2	hi7://localhost9905	SUNPIXPDQPHARM	25	{1,25}	SUNPIX
SUNPIXPDQGRAD	1.2.3.4.5	ISO	HL7V2	hi7://localhost9906	SUNPIXPDQGRAD	25	{1,25}	SUNPIX
XREF2005	1.3.6.1.4.1.21367.2005.1.2	ISO	HL7V2	hi7://localhost9903	XREF2005	25	{1,25}	XREF20

Message Viewer

The Message Viewer displays a list of transactions processed through the PIX/PDQ Manager. You can select a transaction to view trace information about the path it took through the PIX/PDQ Manager, including which components processed the message and the order in which it was processed. The Message Viewer also displays the content, or payload, of the message in XML format, and displays information about which component processed the message at each checkpoint. Use the Message Viewer to monitor the status of the PIX/PDQ system.

Audit Viewer

The Audit View displays an audit log of transactions processed by the PIX/PDQ Manager based on the Audit Trail and Node Authentication (ATNA) guidelines. The information collected for auditing allows you to monitor transactions for inappropriate activity. The central repository for storing audit information is the arrdb database, and the Audit page of the PIX Console provides a graphical interface to view audit information.

The Audit Viewer allows you to filter the list of transactions displayed on the Audit page. You can narrow down transactions by source ID, source type, event ID, event type, event action, event outcome, and date range.

Application Variable and Configuration Manager

The Application Variable and Configuration Manager allows you to view and edit the predefined application variables and configurations that are part of the PIX/PDQ Manager. The variables define connectivity information for the binding components, and define information specific to HL7 messaging for the BPEL Service Engine.

Domain Manager

The Domain Manager displays information about each external system that shares data with the PIX/PDQ Manager. In IHE terminology, these external systems are called domains. In adherence with IHE requirements, the following information is stored for each domain:

- The domain name, called the namespace ID. This can be identical to the system code or name used by the master index.
- The domain's universal ID.
- The domain's encoding scheme (ISO).
- The endpoint type and receiving URL for PIX update notifications.
- A description of the domain.
- The length and format of the local identifiers assigned by the domain.
- The application and facility for HL7 v2 messaging domains, or the device and organization for HL7 v3 messaging domains.

You can make updates to existing domain information, add new domains, and delete domains using the Domain Manager. Once you save any updates you make, the Domain manager propagates the information to the `sbyn_systems` table in the master index database and to the `pix_pdq_systems` table, which supports the domain lookup interface.

Note – If you modify domain information, you need to restart the GlassFish server in order for the changes to take effect in the PIX/PDQ Manager.

The Domain Manager also handles subscriptions to outbound notifications of updates to patient information, letting you add, delete, and modify these subscriptions. You can also view the available endpoint types for subscriptions.

Log Message Viewer

The Log Message Viewer displays messages from the GlassFish server log that are relevant to the PIX/PDQ Manager. This allows you to quickly see the information that applies only to PIX/PDQ without having to look through general application server message or messages for other applications.

Master Index Data Manager

The Master Index Data Manager (MIDM) is a web-based interface that allows you to access, monitor, and maintain the data stored by the master index applications you create using Sun Master Index. The MIDM provides the ability to search for, add, update, deactivate, merge, unmerge, and compare records in the master index database. You can also view and correct potential duplicate profiles, view and reverse assumed matches, view transaction histories and audit logs, and print reports.

The screenshot shows the MIDM web interface. At the top, there is a navigation bar with the following tabs: Dashboard, Duplicate Records, Record Details, Assumed Matches, Transactions, Reports, Source Record, and Audit Log. The 'Dashboard' tab is currently selected. The main content area is divided into four sections:

- Summary:** Contains two yellow boxes. The first box displays 'Potential Duplicates (Last 24 Hours): 0'. The second box displays 'Assumed Matches (Last 24 Hours): 3'.
- Quick Search:** Features a 'Search EUID:' text input field, an 'Advanced Search' link, and a 'Search' button.
- Reports:** Contains a yellow box with four links: 'Merged Records', 'Deactivated EUIDs', 'Unmerged Records', and 'Audit Log'.
- Compare EUIDs:** Features four input fields labeled 'EUID 1:', 'EUID 2:', 'EUID 3:', and 'EUID 4:'. Below these fields are 'Compare' and 'Clear' buttons.

The Sun logo is visible in the top right corner of the interface.

The MIDM is your primary tool to view and maintain the data stored in the master index database and cross-referenced by a master index application. For complete information and instructions for using the MIDM, see the [Master Index Data Manager User's Guide](#)

About the PIX/PDQ Core Projects

The PIX/PDQ core projects define the core functionality of the PIX/PDQ system, including the master index, the domain lookup interface, the master index facade interface, and the audit service interface. They also include two projects that support the PIX/PDQ projects by holding common files or defining utilities.

The PIX/PDQ core projects include the following:

- “PIXPDQAuditService” on page 21
- “PIXPDQCommon” on page 21
- “PIXPDQDomainLookupEJB” on page 21
- “PIXPDQMasterIndex” on page 21
- “PIXPDQMuralFacadeEJB” on page 22
- “PIXPDQUtilities” on page 22
- “PIXUpdateNotification” on page 22

PIXPDQAuditService

The PIXPDQAuditService EJB project defines the interface to the ATNA audit log. This project builds on the functions defined in the ATNA audit log projects and provides an interface to the audit repository. The JAR file from this project is deployed to the application server.

PIXPDQCommon

The PIXPDQCommon project holds all of the processing files that are common to multiple projects in the PIX/PDQ system. These files include the `CustomConfig.jar` file, the audit service WSDL file and its corresponding schema, and the HL7 V3 messaging schemas. For a complete list of the HL7 V3 artifacts provided in the common project, see [“HL7 V3 Messaging” on page 29](#). No files from this project are deployed.

PIXPDQDomainLookupEJB

The PIXPDQDomainLookupEJB project defines an EJB to resolve the domain identification differences between the HL7 messages and the system codes used by the Sun Master Index. The Sun Master Index typically uses the namespace ID field of the domain identifier to locate a system, but the domain identifier includes two additional fields, universal ID and ID type. This EJB module resolves the identification information (namespace ID, universal ID, and ID type) for the domains in the incoming data and then resolves the information again for outbound data. The JAR file from this project is deployed to the application server.

PIXPDQMasterIndex

The PIXPDQMasterIndex project defines the enterprise-wide master patient index that provides a cross-reference of each domain's local IDs and also maintains a centralized repository of each patient's demographic information. This project contains all of the master index processing logic for data standardization, matching, queries, updates, assumed matches, potential duplicate flagging, and single best record (SBR) survivorship. It also defines the data structure for the master patient index. This project is deployed to the application server.

This project, or files from this project, are referenced from several of the projects included with the PIX/PDQ Manager. You should rebuild and redeploy the required projects if you modify the PIXPDQMasterIndex project in such a way that it needs to be rebuilt and redeployed. The PIX/PDQ Manager provides build scripts that automate the process of rebuilding and redeploying all projects. For more information, see [“Building and Deploying the PIX/PDQ Manager Projects”](#) in *Working With the PIX/PDQ Manager*.

PIXPDQMuralFacadeEJB

The PIXPDQMuralFacadeEJB project provides an interface to the functions of the master patient index primarily to facilitate the paging of query results. The JAR file from this project is deployed to the application server. The Libraries folder in this project include the `index-core.jar` and `master-index-client.jar` files from the master index project (PIXPDQMasterIndex), so the project should be rebuilt and the JAR file redeployed when the PIXPDQMasterIndex project is updated.

PIXPDQUtilities

The PIXPDQUtilities project defines utility functions that support the remaining PIX/PDQ core projects, including exceptions, internationalization and location functions, and EJB client stubs. No files from this project are deployed to the application server.

PIXUpdateNotification

The PIXUpdateNotification project defines a message driven bean (MDB) to handle PIX update notification events. These events occur when changes are made application server. This MDB waits for the JMS topic to receive messages. Messages in the topic trigger the PIX update notification bean, which then checks the message to determine which domain caused the update and looks up the domains that are subscribed to updates from that domain. It generates all the necessary JMS messages and queues them on a V2 or V3 topic.

About the Audit Projects (ATNA)

The ATNA Audit projects provide the underlying code that defines how the ATNA requirements are handled for the PIX/PDQ Manager. This includes how messages are generated and written to the arrdb database. These projects support the main Audit service interface, [“PIXPDQAuditService”](#) on page 21.

The ATNA Audit projects include the following projects:

- `AuditClientHelper`
- `AuditRecRepository`
- `AuditRecRepoSyslogClient`

The `AuditRecRepository` project includes the deployed lifecycle module, `com.sun.openesb.ihe.arr.sysloglistener.AuditRecRepoSyslogListenerLoader.java`. It also contains the Java class that defines how audit records are persisted in the `arrdb` database, and the XSL template file that defines the format of the audit messages you can view on the PIX Console.

About the HL7 V2 Projects

The HL7 v2 projects provide the business logic to support HL7 v2 message processing in accordance with the IHE framework. These projects use the HL7 Binding Component along with several BPEL processes to process HL7 v2 messages into and out of the PIX/PDQ Manager. The HL7 v2 projects include two BPEL projects and two corresponding Composite Application projects. The first BPEL project (`IHE_PIX_PDQ_V2_Direct`) handles patient identity feeds, demographic queries, and PIX queries for patient identifiers. The second BPEL project (`IHE_PIX_Update_V2_Direct`) handles outbound patient updates from the master index.

These projects are configured to use the Minimal Lower Layer Protocol (MLLP) version 1.

The HL7 V2 projects include the following:

- [“IHE_PIX_PDQ_V2_Direct” on page 23](#)
- [“IHE_PIX_Update_V2_Direct” on page 24](#)
- [“IHE_PIX_PDQ_V2_Direct_ca” on page 24](#)
- [“IHE_PIX_Update_V2_Direct_ca” on page 25](#)

IHE_PIX_PDQ_V2_Direct

The `IHE_PIX_PDQ_V2_Direct` project handles HL7 v2 messages for patient admissions, updates, and merges as well as PIX queries and patient demographics queries. It contains a main BPEL process and several subprocesses, each designed to handle a different type of HL7 event. The main BPEL process uses a `Pick` activity to determine the type of incoming message and the subprocess to use to process it. The BPEL processes define mapping from the incoming messages to the master index, call the Domain Lookup service to resolve domain names, and define how the data is manipulated or transformed and routed. They also generate ACK/NACK responses and ATNA audit repository messages. The BPEL processes call the master index API to add, update, and merge records and to query the database.

The BPEL subprocesses all perform request/reply operations and are all called from the main BPEL process, IHEPixPdqv2BpelProcess. The IHE_PIX_PDQ_V2_Direct project includes the following subprocesses under the folder BpelProcesses:

- **processAdtA40MsgBpel:** Processes merge events (A40).
- **processAdtMsgBpel:** Processes patient add or update events (A01, A04, A05, and A08).
- **processQ22MsgBpel:** Processes patient demographics queries (Q22).
- **processQbpMsgBpel:** Processes PIX queries for associated patient identifiers (Q23).

The first three subprocesses in the above list map directly to the master index object structure. If you modify the master index object structure, the mapping in these processes might also need to be updated.

This project includes the corresponding WSDL documents for the main process and subprocesses, and XML schema files to support HL7 processing. The main WSDL document, hl7RecvWsd specifies all request and response message types, along with the required ports and HL7 bindings. This is the file where the HL7 WSDL extensibility elements (address, protocolproperties, and communicationcontrols) are defined. There are additional WSDL documents that are specific to the type of event being processed and one to support the ATNA repository. Each is associated with a specific BPEL process and is named similarly to the BPEL process for which it is designed.

IHE_PIX_Update_V2_Direct

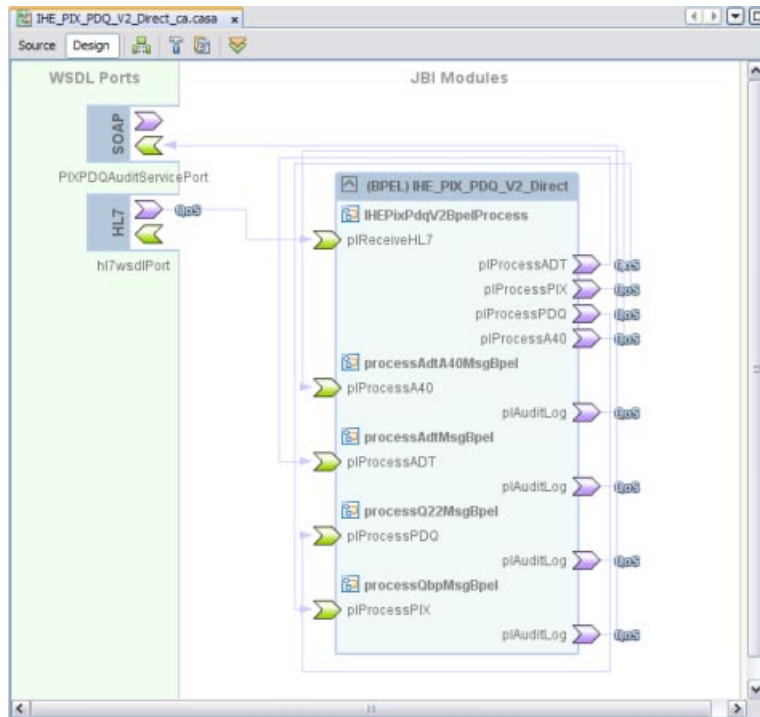
The IHE_PIX_Update_V2_Direct project handles outbound HL7 v2 PIX update notifications. In this type of transaction, updates to the master index data are written to a JMS topic and then processed through the Patient Update Handler MDB to separate HL7 v2 and HL7 v3 topics, making the updates available to any domains that subscribe to one of those topics. The BPEL process in this case receives the message from the HL7 v2, calls the Domain Lookup service to resolve domain names, processes it according to the IHE framework, and then generates appropriate outbound message (an HL7 A31 message). These messages are sent dynamically to subscribed HL7 systems through the HL7 Binding Component. The BPEL process also generates the corresponding ATNA record and writes it to the audit log.

This project only contains one BPEL process, PixOBFeedV2Bpel, that defines all of the processing logic. The BPEL process includes mappings to the master index object structure. If you modify the master index object structure, the mapping in this process might also need to be updated.

IHE_PIX_PDQ_V2_Direct_ca

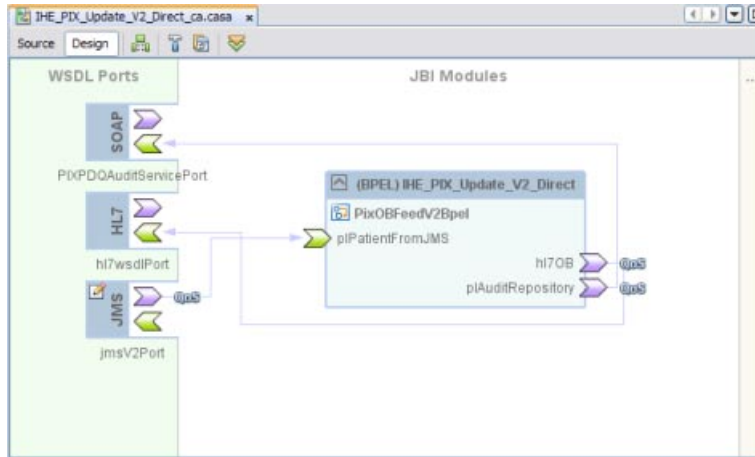
The IHE_PIX_PDQ_V2_Direct_ca project defines the service assembly for the IHE_PIX_PDQ_V2_Direct project. In the graphic below, you can see all messaging to and from

the HL7 systems is transmitted through the HL7 Binding Component, and all ATNA logging activities are handled through the HTTP Binding Component (using SOAP).



IHE_PIX_Update_V2_Direct_ca

The IHE_PIX_Update_V2_Direct_ca project defines the service assembly for the IHE_PIX_Update_V2_Direct project. In the graphic below, you can see that messages from the JMS topic are processed through the JMS Binding Component into the BPEL process, and the response is then processed through the HL7 Binding Component. ATNA logging activities are handled through the HTTP Binding Component (using SOAP).



About the HL7 V3 Projects

The HL7 v3 projects provide the business logic to support HL7 v3 message processing in accordance with the IHE framework. These projects use the HTTP Binding Component with the SOAP protocol instead of the HL7 Binding Component, and process HL7 v3 messages into and out of the PIX/PDQ Manager using BPEL processes to define the logic. The projects implement both SOAP versions 1.1 and 1.2.

The HL7 v3 projects include three BPEL projects and one Composite Application project that provides the service assembly for all three BPEL projects. The PDQ_HL7V3_Direct BPEL project handles patient identity feeds and demographic queries. The PIX_HL7V3_Direct BPEL project handles PIX queries for patient identifiers. The PIX_HL7V3_UpdateNotification_Direct BPEL project handles outbound patient updates from the master index.

The following topics provide information about the HL7 v3 projects:

- “PDQ_HL7V3_Direct” on page 26
- “PIX_HL7V3_Direct” on page 27
- “PIX_HL7V3_UpdateNotification_Direct” on page 28
- “PIXPDQ_HL7V3_Direct_ca” on page 28

For a list of HL7 v3 message types that are included in the PIX/PDQ Manager, see “[HL7 V3 Messaging](#)” on page 29.

PDQ_HL7V3_Direct

The PDQ_HL7V3_Direct project handles HL7 v3 messages for patient demographics queries. It contains a main BPEL process, bpelPDQManager, and one subprocess, bpelExecuteCoreQuery. The main BPEL process calls the subprocess, which in turn call the

Domain Lookup service to resolve domain names, checks whether the configured query uses wildcard characters, and then calls the Master Index Facade EJB to retrieve any matching patient records. It then resolves domain names again and sends the response back out to the main BPEL process.

The main BPEL process also processes query continuation messages and query cancel messages (QUQI_IN000003UV01). A query continuation message is sent by the consumer of the query results when multiple patient records are found as the result of a query and they are returned to the consumer in batches. When the consumer has processed on batch, it sends a message requesting to either continue to the next batch or cancel the query results. The main BPEL process uses On Event activities to determine whether to proceed with the next batch of results or to cancel the process.

The main BPEL process also defines fault handling, generates the ACK/NACK responses, and generates the ATNA audit repository messages. The subprocess in this project maps directly to the master index object structure. If you modify the master index object structure, the mapping in the `bpelExecuteCoreQuery` process might also need to be updated.

This project includes the corresponding WSDL documents for the main process and subprocess, and references the XML schema files to support HL7 v3 processing.

PIX_HL7V3_Direct

The `PIX_HL7V3_Direct` project handles patient identity feeds and PIX queries for identifiers, which includes the following types of HL7 v3 messages:

- Patient Registry Record Added
- Patient Registry Record Revised
- Patient Registry Duplicates Resolved
- Patient Registry Get Identifiers Query

This project only contains one BPEL process, `bpelPIXManager`, that defines all of the processing logic for each message type. The BPEL process receives the incoming request and uses `OnMessage` activities to determine which type of processing to perform. For each message type, the process calls the Domain Lookup service to resolve domain names, and then calls the Master Index Facade EJB to perform the required operation. For patient record adds, the Master Index either adds a new record or updates an existing record depending on whether it finds a match in its database. For an identifiers query, the Master Index returns a list of patient identifiers associated with the given identifier if any are found.

The BPEL process also defines fault handling, generates the ACK/NACK responses, and generates the ATNA audit repository messages. The BPEL process includes mappings to the master index object structure. If you modify the master index object structure, the mapping in this process might also need to be updated.

This project includes the corresponding WSDL documents for each supported message type, and references the XML schema files to support HL7 v3 processing.

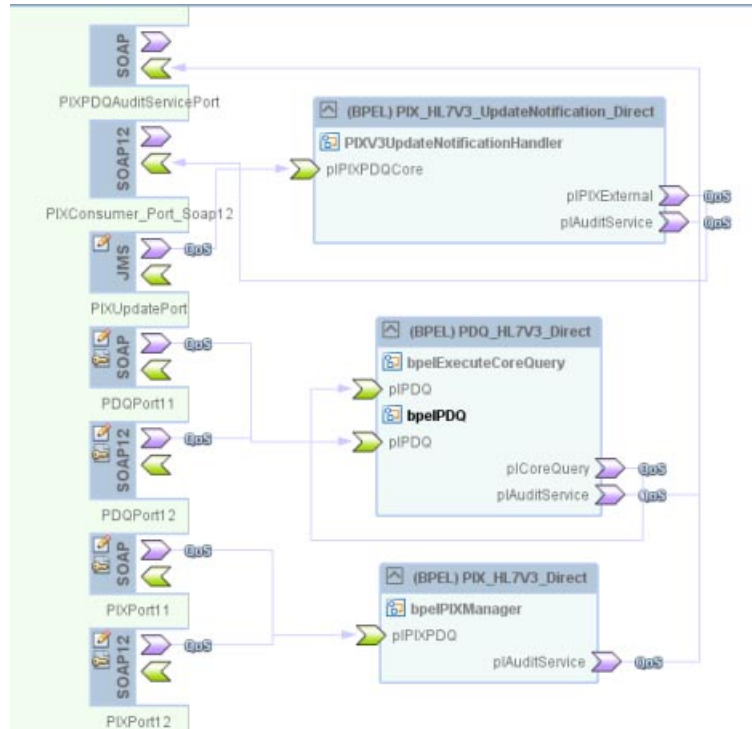
PIX_HL7V3_UpdateNotification_Direct

The `PIX_HL7V3_UpdateNotification_Direct` project handles outbound HL7 v3 PIX update notifications. In this type of transaction, updates to the master index data are written to a JMS topic and then processed through the Patient Update Handler MDB to separate HL7 v2 and HL7 v3 topics, making the updates available to any domains that subscribe to one of those topics. The BPEL process in this case receives the message from the HL7 v3 topic, calls the Domain Lookup service to resolve domain names, processes the message according to the IHE framework, and then generates the appropriate outbound message (a `PRPA_IN201302UV02` message). These messages are sent dynamically to subscribed HL7 systems through the HTTP Binding Component. The BPEL process also performs fault handling, and generates the corresponding ATNA record and writes it to the audit log.

This project only contains one BPEL process, `PIXV3UpdateNotificationHandler`, that defines all of the processing logic.

PIXPDQ_HL7V3_Direct_ca

The `PIXPDQ_HL7V3_Direct_ca` project defines the service assembly for the all of the HL7 v3 BPEL projects. In the graphic below, you can see that messages are processed in through the HTTP Binding Component using either SOAP 1.1 or 1.2. Each BPEL process generates a response, which is then processed back out through the same Binding Component. ATNA logging activities are handled through the HTTP Binding Component (using SOAP) for all three BPEL modules.



HL7 V3 Messaging

The PIX/PDQ solution provides schema definition files (XSD files) for a subset of the HL7 V3 library. The XSD files include definitions for the following message types:

Common Message Elements

COCT_MT030000UV04	COCT_MT150007UV
COCT_MT030007UV	COCT_MT230100UV
COCT_MT030202UV01	COCT_MT240000UV01
COCT_MT030203UV02	COCT_MT240003UV02
COCT_MT030207UV	COCT_MT260003UV
COCT_MT040008UV	COCT_MT280000UV04
COCT_MT040203UV01	COCT_MT290000UV06
COCT_MT050000UV01	COCT_MT300000UV04

COCT_MT060000UV01	COCT_MT310000UV04
COCT_MT070000UV01	COCT_MT440001UV
COCT_MT080000UV	COCT_MT490000UV04
COCT_MT090000UV01	COCT_MT500000UV04
COCT_MT090002UV01	COCT_MT510000UV06
COCT_MT090003UV01	COCT_MT530000UV
COCT_MT090100UV01	COCT_MT600000UV06
COCT_MT090102UV02	COCT_MT670000UV04
COCT_MT090108UV	COCT_MT710000UV01
COCT_MT090300UV01	COCT_MT710007UV
COCT_MT090303UV01	COCT_MT740000UV04
COCT_MT140007UV	COCT_MT810000UV
COCT_MT150000UV02	COCT_MT820000UV
COCT_MT150002UV01	COCT_MT960000UV05
COCT_MT150003UV03	

Message Act Infrastructure

MCAI_MT900001UV01

Message Control Infrastructure

MCCI_IN000002UV01	MCCI_MT000200UV01
MCCI_MT000100UV01	MCCI_MT000300UV01

Master File Management Infrastructure

MFMI_MT700701UV01	MFMI_MT700711UV01
-------------------	-------------------

Patient Administration

PRPA_IN201301UV02	PRPA_MT201301UV02
PRPA_IN201302UV02	PRPA_MT201302UV02

PRPA_IN201304UV02

PRPA_MT201303UV02

PRPA_IN201305UV02

PRPA_MT201304UV02

PRPA_IN201306UV02

PRPA_MT201306UV02

PRPA_IN201309UV02

PRPA_MT201307UV02

PRPA_IN201310UV02

PRPA_MT201310UV02

Query Infrastructure

QUQL_IN000003UV01

QUQL_MT021001UV01

QUQL_MT000001UV01

