

# iWay

iWay Integration Solution for EDIHL7 User's Guide Version 7.0.x and Higher Active Technologies, EDA, EDA/SQL, FIDEL, FOCUS, Information Builders, the Information Builders logo, iWay, iWay Software, Parlay, PC/FOCUS, RStat, Table Talk, Web390, WebFOCUS, WebFOCUS Active Technologies, and WebFOCUS Magnify are registered trademarks, and DataMigrator and Hyperstage are trademarks of Information Builders, Inc.

Adobe, the Adobe logo, Acrobat, Adobe Reader, Flash, Adobe Flash Builder, Flex, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Due to the nature of this material, this document refers to numerous hardware and software products by their trademarks. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

Copyright © 2018, by Information Builders, Inc. and iWay Software. All rights reserved. Patent Pending. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

# Contents

Prefa	ace	. 7
	Documentation Conventions	8
	Related Publications	9
	Customer Support	9
	Help Us to Serve You Better	. 10
	User Feedback	. 12
	Information Builders Consulting and Training	. 12
1. Int	troducing the iWay Integration Solution for EDIHL7	.13
	EDIHL7 Prerequisites	. 13
	Understanding the HL7 Protocol	. 14
	Event Driven Protocol.	.14
	Application to Application Protocol.	.14
	OSI Level 7 Protocol.	.14
	Exchange Protocol	. 14
	Standard Protocol	.14
	Components of an HL7 Message	. 15
	Messages	. 15
	Segments	.15
	Elements	. 16
	Delimiter Characters	. 16
	Escape Characters	.16
	Delimiter Redefinition	. 17
	Present but Null	. 17
	Repetition and Optionality of HL7 Segments	. 18
	HL7 Version 2.x Backward Compatibility	.18
	Withdrawn Segments	.19
	Sample EDIHL7 Integration Scenario	. 19
	HL7 Inbound and Outbound Flows Using MLLP	.20
	HL7 Inbound Flow Using MLLP	. 21
	HL7 Outbound Flow Using MLLP	. 22
	Features of the iWay Integration Solution for EDIHL7	. 22
	Components of the iWay Integration Solution for EDIHL7	. 23

	Ebix (e-Business Information Exchange) File	23
	MLLP Listener	24
	EDIHL7 Batch Splitter Preparser	28
	EDIHL7 Preparser	29
	Support for Z-Segments	30
	EDIHL7 Validation Report Service (com.ibi.agents.XDEDIHL7ValidationReportAgent)	31
	EDIHL7 Acknowledgement Service (com.ibi.agents.XDEDIHL7AckAgent)	31
	XML to EDIHL7 Transform Service (com.ibi.agents.XMLtoEDIHL7TransformAgent)	32
	MLLP Emit Service (com.ibi.agents.XDMLLPEmitAgent)	33
	Deidentification Service (com.ibi.agents.XDDeidentifyAgent)	34
2.	Deployment Information for Your iWay Integration Solution	37
	iWay Products and Components	37
	iWay Service Manager	37
	iWay Integration Tools Transformer	38
	iWay Integration Tools Designer	38
	iWay Correlation Facility	38
	Using a Channel to Construct a EDIHL7 Message Flow	38
	Components of a Channel	39
3.	Downloading, Extracting, and Importing HL7 Sample Data	43
	Downloading and Extracting HL7 Sample Data	43
	Importing HL7 Sample Data to iWay Integration Tools as a Workspace	46
4.	Configuring the EDI Activity Driver	53
	EDIHL7 EDI Activity Driver Overview	53
	Configuring the EDI Data Provider Using iWay Service Manager	53
	Configuring the EDI Activity Driver Using iWay Service Manager	56
5	Inhound Processing: HI 7 to XMI	63
•••	Configuring a Channel for HI 7 Inbound Processing	63
	Configuring Register Sets and Registers	83
	Importing an Ebix Into the Workspace	
	Configuring an iWay Integration Application for Inbound Processing	
	Setting HL7 System Registers	
	Testing the EDIHL7 Inbound Channel Application	. 105

6. Inbound Processing: HL7 to XML (Using MLLP)	
Configuring a Channel for HL7 Inbound Processing Using MLLP	
Configuring Register Sets and Registers	125
Importing an Ebix Into the Workspace	132
Configuring an iWay Integration Application for Inbound Processing	139
Setting HL7 System Registers Using MLLP	147
Testing the Inbound Channel Application Using MLLP	149
7. Outbound Processing: XML to HL7	151
Configuring a Channel for HL7 Outbound Processing	151
Configuring Register Sets and Registers	166
Importing an Ebix Into the Workspace	
Configuring an iWay Integration Application for Outbound Processing	180
Setting EDIHL7 Outbound System Registers	
Testing the Outbound HL7 Channel Application	
8. Outbound Processing: XML to HL7 (Using MLLP)	191
Configuring a Channel for HL7 Outbound Processing Using MLLP	
Configuring Register Sets and Registers	
Importing an Ebix Into the Workspace	
Configuring an iWay Integration Application for Outbound Processing	219
Setting Outbound HL7 System Registers Using MLLP	
Testing the Outbound Channel Application Using MLLP	228
A. Supported HL7 Versions	
Supported HL7 Versions and Messages	
B. Using HL7 Separators and Terminators	233
HL7 Separators and Terminators	
C. Using iWay Integration Tools to Configure an Ebix for HL7	
Using iIT to Configure an Ebix File for HI 70verview	
Using iIT to Configure an Ebix File for HI 7 Prerequisites	
Working With iWay Integration Tools (iIT)	

Contents

This documentation describes how to configure and use the iWay Integration Solution for EDIHL7. It is intended for integration specialists who wish to integrate healthcare enterprise systems by parsing, validating, and storing HL7 messages.

**Note:** This Release 7.0.x content is currently being updated to support iWay Release 8.0.x software. In the meantime, it can serve as a reference for your use of iWay Release 8. If you have any questions, please contact *Customer\_Success@ibi.com*.

# How This Manual Is Organized

	Chapter/Appendix	Contents
1	Introducing the iWay Integration Solution for EDIHL7	Describes the mandate of the Health Level Seven (HL7) protocol and how the components of the iWay Integration Solution for EDIHL7 streamline the flow of information between healthcare partners. Provides a roadmap to information on other products used with the iWay Integration Solution for EDIHL7.
2	Deployment Information for Your iWay Integration Solution	Describes the iWay products used with your iWay Integration Solution for EDIHL7 and provides a roadmap to full information on those products. Introduces the concept of a channel for the construction of a message flow in iWay Service Manager.
3	Downloading, Extracting, and Importing HL7 Sample Data	Describes how to download, extract, and import HL7 sample data.
4	Configuring the EDI Activity Driver	Describes how to configure the EDI Activity Driver using iWay Service Manager.
5	Inbound Processing: HL7 to XML	Describes how to configure a basic inbound message flow for the iWay Integration Solution for EDIHL7.
6	Inbound Processing: HL7 to XML (Using MLLP)	Describes how to configure a basic inbound message flow for the iWay Integration Solution for EDIHL7 using the Minimal Lower Layer Protocol (MLLP).

This manual includes the following chapters:

_	Chapter/Appendix	Contents
7	Outbound Processing: XML to HL7	Describes how to configure a basic outbound message flow for the iWay Integration Solution for EDIHL7.
8	Outbound Processing: XML to HL7 (Using MLLP)	Describes how to configure a basic outbound message flow for the iWay Integration Solution for EDIHL7 using the Minimal Lower Layer Protocol (MLLP).
A	Supported HL7 Versions	Summarizes the HL7 versions that are currently supported by the iWay Integration Solution for EDIHL7.
В	Using HL7 Separators and Terminators	Includes a list of separators and terminators that are allowed.
С	Using iWay Integration Tools to Configure an Ebix for HL7	Describes how to use iWay Integration Tools (iIT) to configure an Ebix for HL7.

# **Documentation Conventions**

The following table describes the documentation conventions that are used in this manual.

Convention	Description
THIS TYPEFACE or this typeface	Denotes syntax that you must enter exactly as shown.
this typeface	Represents a placeholder (or variable), a cross-reference, or an important term. It may also indicate a button, menu item, or dialog box option that you can click or select.
underscore	Indicates a default setting.
Key + Key	Indicates keys that you must press simultaneously.
{}	Indicates two or three choices. Type one of them, not the braces.
	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.

Convention	Description
	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis ().
	Indicates that there are (or could be) intervening or additional commands.

# **Related Publications**

Visit our Technical Documentation Library at *http://documentation.informationbuilders.com*. You can also contact the Publications Order Department at (800) 969-4636.

# **Customer Support**

Do you have any questions about this product?

Join the Focal Point community. Focal Point is our online developer center and more than a message board. It is an interactive network of more than 3,000 developers from almost every profession and industry, collaborating on solutions and sharing tips and techniques. Access Focal Point at <a href="http://forums.informationbuilders.com/eve/forums">http://forums.informationbuilders.com/eve/forums</a>.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our website, *http:// www.informationbuilders.com*. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of *http://www.informationbuilders.com* also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

Call Information Builders Customer Support Services (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 a.m. and 8:00 p.m. EST to address all your questions. Information Builders consultants can also give you general guidance regarding product capabilities and documentation. Please be ready to provide your six-digit site code number (*xxxx.xx*) when you call.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

# Help Us to Serve You Better

To help our consultants answer your questions effectively, be prepared to provide specifications and sample files and to answer questions about errors and problems.

The following tables list the environment information our consultants require.

	-
Platform	
Operating System	
OS Version	
JVM Vendor	
JVM Version	

The following table lists the deployment information our consultants require.

Adapter Deployment	For example, iWay Business Services Provider, iWay Service Manager
Container	For example, WebSphere
Version	
Enterprise Information System (EIS) - if any	
EIS Release Level	
EIS Service Pack	
EIS Platform	

The following table lists iWay-related information needed by our consultants.

iWay Adapter	
iWay Release Level	
iWay Patch	

The following table lists additional questions to help us serve you better.

<b>Request/Question</b>	Error/Problem Details or Information
Did the problem arise through a service or event?	
Provide usage scenarios or summarize the application that produces the problem.	
When did the problem start?	
Can you reproduce this problem consistently?	
Describe the problem.	
Describe the steps to reproduce the problem.	
Specify the error message(s).	
Any change in the application environment: software configuration, EIS/database configuration, application, and so forth?	
Under what circumstance does the problem <i>not</i> occur?	

The following is a list of error/problem files that might be applicable.

- □ Input documents (XML instance, XML schema, non-XML documents)
- Transformation files
- Error screen shots
- Error output files
- □ Trace files

Service Manager package to reproduce problem

- Custom functions and agents in use
- Diagnostic Zip
- Transaction log

For information on tracing, see the *iWay* Service Manager User's Guide.

# **User Feedback**

In an effort to produce effective documentation, the Technical Content Management staff welcomes your opinions regarding this document. Please use the Reader Comments form at the end of this document to communicate your feedback to us or to suggest changes that will support improvements to our documentation. You can also contact us through our website, *http://documentation.informationbuilders.com/connections.asp.* 

Thank you, in advance, for your comments.

# Information Builders Consulting and Training

Interested in training? Information Builders Education Department offers a wide variety of training courses for this and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our website (*http://education.informationbuilders.com*) or call (800) 969-INFO to speak to an Education Representative.



# Introducing the iWay Integration Solution for EDIHL7

The iWay Integration Solution for EDIHL7 transforms HL7 messages into standard XML format, or transforms XML representations into HL7 message format.

This section provides an overview of HL7 and describes the features that are provided by the iWay Integration Solution for EDIHL7.

#### In this chapter:

- EDIHL7 Prerequisites
- Understanding the HL7 Protocol
- Components of an HL7 Message
- Sample EDIHL7 Integration Scenario
- HL7 Inbound and Outbound Flows Using MLLP
- Features of the iWay Integration Solution for EDIHL7
- Components of the iWay Integration Solution for EDIHL7

# EDIHL7 Prerequisites

Before you use the iWay Integration Solution for EDIHL7 for inbound (HL7 to XML) and outbound (XML to HL7) processing, ensure that the following prerequisites are met:

- You have a working knowledge of iWay Service Manager (iSM) and iWay Integration Tools (iIT).
- □ iSM Version 8.0 is installed.
- iWay Integration Solution for EDIHL7 (Patch) is installed.
- □ iIT Version 8.0 is installed.
- System and channel Special Registers (SREGs) are updated to match your directory structure.

# **Understanding the HL7 Protocol**

Health Level Seven (HL7) is a standard for information exchange between medical applications. It is the seventh OSI layer protocol for exchanging information in healthcare systems. HL7 defines a protocol for data exchange. It also defines the format and the content of the messages that applications must use when exchanging data with one another in various circumstances. The following descriptions provide an outline for the protocol.

#### **Event Driven Protocol**

Real world events, such as the admission of a patient, cause messages to flow between applications. In other words, an application that encounters a real world event sends a message to other applications that need to be aware of this event.

#### **Application to Application Protocol**

It defines a communication between two independent applications, rather than between closely coupled, client-server type applications. The scope of interest for HL7 is the message exchange between the applications, rather than the specific role of each application in the health care delivery process.

# **OSI Level 7 Protocol**

The scope of HL7 is the format and content of the data exchanged between the applications, not how it is passed between computers or networks. HL7 does not specify how messages will be delivered between the applications. Usually a TCP/IP connection or FTP file transfer is used to deliver a message. However, within local area networks, the standard is the Lower Layer Protocol.

#### **Exchange Protocol**

HL7 specifies the way data exchange between applications will be accomplished. It does not specify how applications store or process this data. In an application, it is recommended that a database structure coincides with HL7 message definitions. However, this is not mandatory.

# **Standard Protocol**

When a proprietary, non-standard link between two systems is made, a message exchange that satisfies your requirements and fits into the data structure of your application can be created. However, the efforts invested in this link are useless when considering a connection to other third-party systems. When you are using HL7, your initial development effort can be reused.

For more information on the HL7 standard, please visit the following website:

#### http://www.hl7.org

#### Components of an HL7 Message

HL7 has a message-oriented architecture. This means that the application in which an event occurs will send a message to other applications rather than serving a request.

**Note:** The application that issues the message is called a Sender or Sending Application, and the addressee (recipient) of the message is called a Receiver or Receiving Application.

The structure of an HL7 Version 2.x message has the following format:

Message --> Segments --> Elements

#### Messages

This section provides a use case of a typical HL7 ADT^A04 message. This message is sent when a new patient arrives at the hospital. The demographics of the patient are entered into a hospital information system (HIS) and then the information is communicated to all the other systems to avoid multiple entries of the patient's demographic information.

```
MSH|^~\&|EPIC||SMS||201501011408||ADT^A04|9000123|D|2.7
EVN|A04|201501011408
PID||0493575^^21D1|454721||DOE^JOHN
PV1||0|168~219~C|R
IN1||ABC123|Blue Cross Blue Shield
```

HL7 messages are ASCII messages, and the standard defines them to be human readable. Messages are a defined sequence of segments and/or segment groups. Each segment, group, or message set within a message, can be optional and/or repeating. Each message consists of the segments that are delimited by carriage return characters ("\r" or 0x0D).

#### Segments

Every line in a message is called a segment and contains information of a specific type. For example, the MSH segment contains information about the Sender and Receiver of the message, the type of the message, a time stamp, and so on. The EVN segment contains information about the type of message, for example, A04 (Register a patient). The PID segment contains demographic information about the patient such as name, ID codes, address, and so on. The PV1 segment contains information regarding the patient's stay in the hospital, such as location assigned, referring doctor, and so on. The IN1 segment contains information regarding the patient's insurance plan.

The sample message in the previous section contains MSH, EVN, PID, PV1, and IN1 segments. There are more than 183 segments defined in Version 2.7 of the HL7 standard. An HL7 message is a combination of the segments represented in sequence. An HL7 message definition states whether each segment is mandatory or not. Segments consist of elements that may be composites. Elements are delimited by a vertical pipe (|). Each element is defined by the HL7 standard.

## Elements

Elements are the building blocks of segments. Elements may be a primitive data type (string, number, and so on), or be made up of other composites. Elements cannot have a recursive reference to themselves. The components of each element are delimited by characters and the sub-components of these components themselves can be delimited using ampersand (&) characters.

# **Delimiter Characters**

An important part of the HL7 protocol is the use of delimiter characters. The following table lists the default delimiter characters used in HL7:

Character	Purpose
0x0D – <cr> (Carriage Return)</cr>	Marks the end of each segment
(Vertical pipe)	Field Delimiter
^ (Caret)	Sub-Field Delimiter
& (Ampersand)	Sub-Sub-Field Delimiter

Two other important characters are the tilde character ( $\sim$ ), which is used to separate repeating fields, and the escape character ( $\setminus$ ).

# **Escape Characters**

Some user data may contain these special delimiter characters. For this reason, HL7 has a system for escaping them. Unlike a language like C, each character in the system has a unique escape sequence. The following table shows the escape sequences for each of the different characters:

Character	Escape Sequence
& (Ampersand)	\T\
^ (Caret)	\\$\
(Vertical pipe)	\F\
~ (Tilde)	\R\
∖ (Backward slash)	\E\

Unprintable hex characters are also escaped in the format \0xXX\, unicode characters in the format \UXXXX\, and multi-byte character sequences \MXXXXXX\ for far eastern language support.

#### **Delimiter Redefinition**

The delimiter characters can be redefined on the fly in the MSH segment of each message. Each HL7 message starts with a MSH segment. Some HL7 implementations may choose to use different delimiter characters in the message. For example:

MSH\$^~\&\$EPIC\$\$SMS\$\$201501011408\$\$ADT^A04\$9000123\$D\$2.7

MSH|^~\&|EPIC||SMS||201501011408||ADT^A04|9000123|D|2.7

# Present but Null

When a field is absent from a message, it should mean that the system does not use that field or that the field has not changed. But what if a system supports the given field, but the value of the field is null? The HL7 protocol requires a method of making this clear. This is done by putting two double quote characters ("") together in a field like this:

ZBR | | | | " " | | | |

# **Repetition and Optionality of HL7 Segments**

In the message definition, each segment can be either mandatory or optional. Each message starts with an MSH segment that is always mandatory (required). Another example of a mandatory field is PID (Patient identification). Without patient identification data, messages like ADT^AO4 (Register Patient) do not have any relevance. Some segments, such as AL1 (Allergies), are optional because patients may or may not have allergies. The message in the example below consists of MSH, EVN, PID, NK1 and PV1. According to the HL7 Version 2.7 definition, in an ADT^AO4 message the MSH, EVN, PID and PV1 segments are required and the NK1 segment is optional. DG1, PR1 and AL1 are also optional segments that could be in this message, but are not.

HL7 is order-sensitive. Order is important to both segments and fields inside the segment.

Segments in HL7 messages can also be repeating. For example, NK1 (Next of Kin/Associated Parties) will repeat several times if a person has several next of kin relationships:

```
MSH | ~~\& | EPIC | | SMS | SMSDT | 201501011408 | | ADT^A04 | 9000123 | D | 2.7 |
EVN | A04 | 201501011408
PID | 0493575^^^21D 1 | 454721 | | DOE^JOHN
NK1 | CONROY^MARI | SPO | (216)731-4359
NK1 | DOE^JOHNNY^^^ | CHD | | (216)731-4222
NK1 | DOE^ROBERT ^^^ | CHD | | (216)731-4222
PV1 | 0 | 168 ~219~C~P
```

A group of segments may be optional as a group and may repeat as a group. For example, ORU^R01 message has a group that contains one OBR (Observation request) and 0 to N OBX (Observation result) segments. This group is optional in the message so it may not appear at all. The message will then look like this:

#### MSH PID PD1

On the other hand, the message may contain a number of observation groups. Then the OBR-OBX group will repeat and the message will look like this:

MSH PID PD1 OBR OBX OBX OBX OBR OBX OBX

The first three observation results may belong to the first observation request, and the next two observation results may belong to the second observation request.

```
MSH PID PD1 - header and patient demographics
OBR OBX OBX OBX - first observation group (e.g. height and weight)
OBR OBX OBX - second observation group (e.g. lab results)
```

# HL7 Version 2.x Backward Compatibility

HL7 2.x is designed to be backward compatible.

A good example to clarify this comes from considering a Patient ID field in a PID segment. Early versions of HL7 declare a Patient ID to be just a simple one field identifier, such as:

```
|PatientID|
```

In more recent versions, the Patient ID has become embellished with composite data types:

PatientID^Check Digit^Assigning Authority NameSpace&...

Most of the time, extra fields are optional in HL7 2.x. When additional fields are not present, the trailing delimiters are not displayed.

```
234324^^^
```

is equivalent to:

234324

#### Withdrawn Segments

In newer versions of HL7, segments can become obsolete or withdrawn. This is noted by the W data type.

These elements remain in the metadata as placeholders. Their data types have been changed to allow any legacy data to pass through transformation. The user may modify the Ebix to reintroduce validation for these elements if desired.

# Sample EDIHL7 Integration Scenario

A hospital needs a number of software applications to efficiently and accurately manage daily functions. A dedicated application system is available to cater to the needs of individual functional entity, such as Patient Registration, Billing, Nursing and Bed Management, Order Management, Pharmacy, and Laboratory. These systems are provided by different vendors and are typically built using different technologies. HL7 plays the vital role of providing mechanism for interconnecting these systems. Each individual application entity has the capability of exchanging information using HL7 messages. They often maintain the same information in different forms and use different codes to represent common values. This requires an integration engine that has HL7 capabilities.

This diagram illustrates a typical integration scenario. Data is exchanged over different interfaces to ensure a connected and functional application system environment.



# HL7 Inbound and Outbound Flows Using MLLP

This section provides diagrams that illustrate HL7 inbound and outbound flows using MLLP.

# HL7 Inbound Flow Using MLLP



# HL7 Outbound Flow Using MLLP



# Features of the iWay Integration Solution for EDIHL7

The iWay Integration Solution for EDIHL7 reduces the amount of effort required to integrate HL7 messages with your internal enterprise applications. It includes conversion of messages from HL7 to XML format, making it easy to include HL7 messages in your integration projects. Features of the iWay Integration Solution for EDIHL7 include:

- □ Integration with iWay Service Manager to provide bi-directional (synchronous and asynchronous) conversion of HL7 formats and XML.
- □ Interactions with application servers, integration brokers, third-party software packages, and messaging services are also supported.
- □ Integration with more than 200 other information assets, including J2EE-based back-office systems, data structures, such as DB2, IMS, VSAM, and ADABAS, and front-office systems based on Sybase.

❑ Support for HL7 messages. For more information, see Supported HL7 Versions on page 231.

# Components of the iWay Integration Solution for EDIHL7

iWay business components used in the construction of a message flow for HL7 messages include:

- □ Ebix (e-Business Information Exchange) File
- MLLP Listener
- EDIHL7 Batch Splitter Preparser
- EDIHL7 Preparser
- EDIHL7 Validation Report Service
- EDIHL7 Acknowledgement Service
- □ XML to EDIHL7 Transform Service
- MLLP Emit Service
- Deidentification service

#### Ebix (e-Business Information Exchange) File

iWay Software provides various e-Business Information Exchange (Ebix) files used in conjunction with the iWay integration solutions. In iWay Service Manager, the iWay Integration Solution for EDIHL7 contains several Ebix files, one for each supported HL7 version. An Ebix file for HL7 is named using the following format:

HL7\_*version*.ebx

where:

#### version

Is the HL7 version number. For example, the Ebix file for HL7 Version 2.7 is named HL7\_2.7.ebx.

An Ebix is a collection of metadata that defines the structure of data. The Ebix supplied with the iWay Integration Solution for EDIHL7 defines the structure of supported HL7 messages.

Each Ebix includes:

- Pre-built data dictionaries. The structure of each HL7 document is described by two data dictionaries:
  - □ Header dictionary, which describes the enveloping structure of the document.
  - Document dictionary, which describes the segments and elements that compose each document.

The dictionaries from the Ebix are used to transform the structure of a document based on the definition of the HL7 standard.

- Pre-built XML schemas that define the structure and content of XML messages in detail.
- □ Pre-built HL7 to XML transformation templates, and XML to HL7 templates, for the supported HL7 versions.
- □ Pre-built rule files for each message. The iWay Integration Solution for EDIHL7 uses these rule files to validate inbound and outbound documents.

# **MLLP Listener**

The MLLP listener uses the Minimal Lower Layer Protocol (MLLP) to receive messages within a channel from configured endpoints.

The following table lists and describes the parameters for the MLLP listener.

Parameter	Description
IP Properties	
Port*	The TCP port where MLLP messages are received.
Local Bind Address	Local bind address for multi-homed hosts. This parameter is usually left blank.
Persistent Connection	If set, the connection is maintained until client closes or the Persistence Timeout expires.

Parameter	Description
Maximum Connections	Maximum number of simultaneous connections allowed. When this threshold is reached, new connections will not be accepted until current connections have ended and the total number of connections is below the limit. Leave blank or set to zero for no maximum.
Persistence Timeout value in Minutes	Maximum length of time (in minutes) that a connection can persist with no activity. 0 or blank will default to 60.
Set Response NoDelay	If true, disables Nagle's Algorithm on the response. This will result in faster line turnaround at the expense of an increased number of packets.
Reuse Address	If true, when the connection is closed, immediately make the address available, bypassing the TCP defaults.
Allowable Clients	If supplied, only messages from this list of fully qualified host names and/or IP addresses are accepted. Enter as comma-separated list or use FILE().
Secure Connection	Use a secure connection over SSL.
SSL Context Provider	iWay Security Provider for SSL Context. If this component is secure and SSL Context Provider is left blank, then the default provider will be used.
MLLP	
Start Block Character	The encoding of the Start of Block character in decimal, or hexadecimal format. The default is 0x0B.
End Block Character	The encoding of the End of Block character in decimal, octal, or hexadecimal format. The default is 0x1C.
Maximum Input Size	Maximum number of bytes read before the end of the message is found. This prevents denial of service attacks with very large messages or a large number of bytes between messages.
Tuning	

Parameter	Description
Multithreading	Indicates the number of worker threads (documents or requests) that iWay Service Manager can handle in parallel. Setting this to a value of greater than 1 enables the listener to handle a second request while an earlier request is still being processed. The total throughput of a system can be affected by the number of threads operating. Increasing the number of parallel operations may not necessarily improve throughput. The default is 1. The max value is 99.
Maximum Threads	The parallel threads can grow to this count automatically on demand. Over time, the worker count will decrease back to the multithreading level. Use this parameter to respond to bursts of activity.
Optimize Favoring	Use this option to customize how the listener performs. For smaller transactions, select <i>performance</i> . For large input documents that could monopolize the amount of memory used by iWay Service Manager, select <i>memory</i> .
Events	
Failed ReplyTo Flow	Name of a published process flow to run if a message cannot be emitted on an address in its reply address list.
Dead Letter Flow	Name of a published process flow to run if an error cannot be emitted on an address in its error address list.
Channel Failure Flow	Name of a published process flow to run if this channel cannot start or fails during message handling. iWay Service Manager will attempt to call this process flow during channel shut down due to the error.
Parse Failure Flow	The name of a published process flow to run if XML parsing fails for the incoming message.

Parameter	Description
Channel Startup Flow	The name of a published process flow to run prior to starting the channel.
Channel Shutdown Flow	The name of published process flow to run when the channel is shut down.
Other	
Whitespace Normalization	Specifies how the parser treats whitespace in Element content. Select <i>preserve</i> to turn off all normalization as prescribed by the XML Specification. Select <i>condense</i> to remove extra whitespace in pretty printed documents and for compatibility with earlier versions.
Accepts non-XML (flat) only	If set to <i>true</i> , the input data is sent directly to the business logic step. The data is not preparsed, parsed, or validated. This flag is used primarily to send non-XML to the business logic or replyTo without processing it.
Execution Time Limit	The maximum time that a request may take to complete. Used to prevent runaway requests. Any request that takes longer to complete than this value will be attempted to be terminated.
Default Java File Encoding	The default encoding if the incoming message is not self- declaring (that is, XML).

Parameter	Description
Agent Precedence	Sets the order by which iWay Service Manager selects agents. iWay Service Manager selects the agent or agents to process the document by searching through the configuration dictionary. Usually, it looks for a document entry in the configuration and when a match is found, the agent specified in that document entry is selected. If a matching document entry is not found, or no agent is specified, the engine looks in the input protocol configuration (listener). To have the processing agent taken directly from the listener (thus ignoring the document entry), use <listener> overrides <document>. Possible values are <document> overrides <listener> and <listener> overrides <document>.</document></listener></listener></document></document></listener>
Error Documents treated normally	If set to <i>true</i> , error documents are processed by any configured preemitters.
Listener is Transaction Manager	If set to <i>true</i> , agents run within a local transaction.
Record in Activity Log(s)	If set to <i>true</i> , activity on this channel will be recorded in the activity logs, otherwise the activity will not be recorded.
AES Key	If the channel will receive encrypted AFTI messages, set the AES key (maximum 16 characters) to be used for decrypting.
Startup Dependencies	A comma-separated list of channel names that must be started before this one is called.

# EDIHL7 Batch Splitter Preparser

The EDIHL7 Batch Splitter preparser (com.ibi.preparsers.EDIHL7BatchSplitter) splits the HL7 batch into individual HL7 messages. It splits on the Message Header (MSH) prior to parsing the data into XML. The EDIHL7 Batch Splitter preparser should only be used in conjunction with the EDIHL7 preparser. In addition, the EDIHL7 Batch Splitter preparser should be placed in order first if multiple preparsers are used.

The EDIHL7 Batch Splitter preparser will extract all messages from a batch. It does not write the following format to your output XML document: FHS - BHS - MSH - BTS - FTS

The following table lists and describes the parameters for the EDIHL7 Batch Splitter preparser.

Parameter	Description
timestamp	Select <i>true</i> from the drop-down list to write a timestamp to the log file. By default, this parameter is set to <i>false</i> .

# EDIHL7 Preparser

The EDIHL7 preparser (com.ibi.preparsers.XDEDIHL7PreParser) is available for the iWay Integration Solution for EDIHL7. The preparser for the iWay Integration Solution for EDIHL7 converts an incoming HL7 formatted document to iWay XML format.

Parameter	Description
Template	The pattern used to lookup a document inside the Ebix. The following format is used:
	HL7_%_^toXML.xch
	where:
	8
	Represents the message type.
	~
	Represents the version.
Timestamp	Select <i>true</i> from the drop-down list to write a timestamp to the log file. By default, this parameter is set to <i>fal</i> se.

The following table lists and describes the parameters for the EDIHL7 preparser.

Parameter	Description
Z-segment	Controls how Z-segments should be formatted in the output XML document. If this parameter is set to <i>false</i> , then the fields and components in the segment are not parsed. By default, this parameter is set to <i>false</i> .
	For more information, see <i>Support for Z-Segments</i> on page 30.
Segment	Select <i>true</i> from the drop-down list to add the <i>name</i> attribute to the segment. By default, this parameter is set to <i>false</i> .
Composite	Select <i>true</i> from the drop-down list to add the <i>name</i> attribute to the composite. By default, this parameter is set to <i>false</i> .
Element	Select <i>true</i> from the drop-down list to add the <i>name</i> attribute to the element. By default, this parameter is set to <i>false</i> .

# Support for Z-Segments

HL7 is a proprietary message structure that allows you to define your own structure. Since HL7 messages can be customized, the HL7 committee provides a framework for the addition of custom information into HL7 messages.

You can define a custom segment in any HL7 message using iWay Integration Tools (iIT). These segments must be named starting with the letter Z.

The EDIHL7 preparser (com.ibi.preparsers.XDEDIHL7PreParser) provides support for Z-segments. The *Make floating structure* parameter controls how Z-segments should be formatted in the output XML document. By default, this parameter is set to *false*.

For example, consider the following segment:

#### ZMS | 20150401 | FOL | 100

If the *Make floating structure* parameter is set to *false*, then the fields and components in the segment are not parsed. The resulting output in the XML document is:

<Z>ZMS | 20150401 | FOL | 100</Z>

If the *Make floating structure* parameter is set to *true*, then the fields and components are parsed. The resulting output in the XML document is:

```
<Z segment-name="ZMS">
        <_01_0 name="">20150401</_01_0>
        <_02_0 name="">FOL</_02_0>
        <_03_0 name="">100</_03_0>
</Z>
```

# EDIHL7 Validation Report Service (com.ibi.agents.XDEDIHL7ValidationReportAgent)

#### Syntax:

com.ibi.agents.XDEDIHL7ValidationReportAgent

#### **Description:**

The EDIHL7 Validation Report service is used to generate a validation report after validating an inbound HL7 message or an outbound XML message.

#### **Parameters:**

Parameter	Description
Input Message	Select <i>true</i> to add the input message to the generated validation report.
Output Message	Select <i>true</i> to add the output message to the generated validation report.

#### EDIHL7 Acknowledgement Service (com.ibi.agents.XDEDIHL7AckAgent)

#### Syntax:

com.ibi.agents.XDEDIHL7AckAgent

#### **Description:**

The EDIHL7 Acknowledgement service is used to generate acknowledgement messages using predefined rules. The service chooses ACK / NACK status based on the validity and verification status of the received HL7 message. The generated acknowledgement message can then be sent using the default path of the listener.

#### **Parameters:**

Parameter	Description
Sending Application *	The name of the sending application.
Sending Facility *	The name of the sending facility.
Control ID *	The Control ID for the message.
Mode *	Select one of the following acknowledgment modes from the drop-down list:
	Original
	Enhanced
	The default value is Original.
Acknowledgment *	Specify when an acknowledgement message should be generated by selecting the corresponding value from the drop-down list:
	Always
	On Error
	On Success
	The default value is Always.

# XML to EDIHL7 Transform Service (com.ibi.agents.XMLtoEDIHL7TransformAgent)

#### Syntax:

com.ibi.agents.XMLtoEDIHL7TransformAgent

#### **Description:**

This service is used in outbound processing to convert the XML-formatted HL7 document to an HL7 formatted document.

# Parameters:

Parameter	Description
timestamp	Select <i>true</i> from the drop-down list to write a timestamp to the log file. By default, this parameter is set to <i>fal</i> se.

# MLLP Emit Service (com.ibi.agents.XDMLLPEmitAgent)

#### Syntax:

com.ibi.agents.XDMLLPEmitAgent

#### **Description:**

This service emits a message using the Minimal Lower Layer Protocol (MLLP). This protocol allows you to wrap an HL7 message with a header and footer to ensure you know where a message starts, where a message stops, and where the next message starts.

#### **Parameters:**

Parameter	Description
Host *	The machine name or IP address of the MLLP destination.
Port *	The designated TCP/IP port that is being used to receive MLLP messages.
Secure Connection	Select <i>true</i> from the drop-down list if you want use a secure connection through Secure Sockets Layer (SSL). By default, this parameter is set to <i>false</i> .
SSL Context Provider	If configured, specify the name of an available iWay Security Provider for SSL Context. If the Secure Connection parameter is set to <i>true</i> and the SSL Context Provider field is blank, then the default provider will be used.
Set TCP No Delay	If set to <i>true</i> , then Nagle's Algorithm on the client socket will be disabled. This will result in faster line turnaround at the expense of an increased number of packets. By default, this parameter is set to <i>false</i> .

Parameter	Description
Socket Timeout	The timeout value in seconds. When a non-zero timeout value is specified, a read() function call on the socket will block for only the amount of time specified (in seconds). If the timeout expires, a <i>java.net.SocketTimeoutException</i> is generated. The default timeout is dependent on the operating system being used.
Persistence	If set to <i>true</i> , then iWay Service Manager (iSM) is instructed to maintain the connection.
Persistence Timeout value in Minutes	The maximum length of time (in minutes) that a connection can persist with no activity. If this parameter value is zero (0) or left blank, then the default is set to 60 minutes.
Retry Count	The number of times to try to send the message after an initial failure is encountered. By default, this parameter is set to <i>0</i> .
Retry Pause	The amount of time (in milliseconds) to wait between retry attempts. By default, this parameter is set to 1000.
Start Block Character	The encoding of the Start of Block character in decimal, octal, or hexadecimal format. The default is <i>0x0B</i> .
End Block Character	The encoding of the End of Block character in decimal, octal, or hexadecimal format. The default is <i>0x1C</i> .
Maximum Message Size	The maximum size of a message that can be sent or received through this emitter. If this parameter value is zero (0) or left blank, then the default is set to 256KB.
Output Document	Determines whether the output document is a <i>response</i> document, <i>status</i> document, or the original <i>input</i> document. By default, this parameter is set to <i>response</i> .

# Deidentification Service (com.ibi.agents.XDDeidentifyAgent)

# Syntax:

com.ibi.agents.XDDeidentifyAgent

#### **Description:**

The Deidentification service provides algorithms that can be used to implement the deidentification of protected health information in accordance with the Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule. Multiple algorithms can be configured since a combination of algorithms will be needed to deidentify the data correctly.

The Deidentification service takes an XML document as input. The first configured algorithm takes this document as input and modifies it in place. The result is fed into the next configured algorithm and so on. The result of the last configured algorithm is the XML document returned by the service.

For more information on configuring and using the Deidentification service, see the *iWay* Service Manager Component Reference Guide.


# Deployment Information for Your iWay Integration Solution

This topic describes the iWay products used with your iWay Integration Solution for EDIHL7 and provides a roadmap to full information on those products.

It also introduces the concept of a channel for the construction of a message flow in iWay Service Manager.

#### In this chapter:

- iWay Products and Components
- Using a Channel to Construct a EDIHL7 Message Flow

#### iWay Products and Components

Your iWay integration solution works in conjunction with one or more of the following products and components:

- iWay Service Manager
- iWay Integration Tools Transformer
- iWay Integration Tools Designer
- iWay Correlation Facility

#### iWay Service Manager

iWay Service Manager is the heart of the Universal Integration Framework and is an open transport service bus. Service Manager uses graphical tools to create sophisticated integration services without writing custom integration code by:

- Using metadata from target applications
- Transforming and mapping interfaces
- Managing stateless processes

Its capability to manage complex integration interactions makes it ideally suited to be the foundation of a service-oriented architecture.

For more information, see the IWay Service Manager User's Guide.

#### iWay Integration Tools Transformer

iWay Integration Tools (iIT) Transformer (previously known as iWay Transformer) is a GUI tool that is delivered as a plugin with iIT. iIT Transformer is a rule based data transformation tool that converts an input document of one data format to an output document of another data format or structure. The easy-to-use graphical user interface and function tool set facilitate the design of transform projects that are specific to your requirements.

For more information, see the *iWay Integration Tools Transformer User's Guide*.

#### iWay Integration Tools Designer

iWay Integration Tools (iIT) Designer (previously known as iWay Designer) is a GUI tool that is delivered as a plugin with iIT.

The capability of graphically visualizing a business process is a powerful and necessary component of any e-Business offering. iWay Integration Tools Designer, a Windows-based design-time tool, provides a visual and user-friendly method of creating a business process, also called a process flow. Through a process flow, you control the sequence in which tasks are performed and the destination of the output from each task.

For more information, see the *iWay Integration Tools Designer User's Guide*.

#### iWay Correlation Facility

The iWay Correlation Facility (also known as the Correlation Manager) maintains records of anticipated activities occurring in the system. Correlation actions take the correlation from OPEN to CLOSED state, and allow history to be recorded. Agents are provided to implement Correlation Facility interactions within process flows, however, it is possible to use this API to accomplish this same purpose within your own exits.

For more information on using the iWay Correlation Facility, see the *iWay Service Manager User's Guide* and the *iWay Service Manager Programmer's Guide*.

# Using a Channel to Construct a EDIHL7 Message Flow

The use of iWay Service Manager is centered on a channel. A channel is a container for all the iWay business components used in an EDI message flow.

At a high level, a channel accepts input data through an **inlet**, processes the data using a **route**, and outputs the resulting data through an **outlet**. Another component in the process is an e-Business Information Exchange (**Ebix**).

The following diagram shows the channel components available in the construction of a message flow.

In the following diagram, the value **n** underneath a component name indicates how many instances of that component you can have in a channel configuration—zero, one, or more than one. For example, n = 1 for Inlet means that you can have only one inlet on the channel.

Required components are in boldface type.



#### **Components of a Channel**

A channel consists of:

- □ An inlet, which defines how a message enters a channel.
- A route, which defines the path a message takes through a channel.

- Outlets, which define how transformed messages leave a channel.
- ❑ An e-Business Information Exchange (Ebix), which is a collection of metadata that defines the structure of data.

iWay Service Manager provides a design-time repository called the Registry, where you assemble and manage the components in a channel.

An inlet can contain:

- A listener (required), which is a protocol handler responsible for picking up an incoming message on a channel.
- A decryptor, which applies a decryption algorithm to an incoming message and verifies the security of the message.
- ❑ A preparser, which is a logical process that converts an incoming message into a document that can be processed. The preparsed document then passes through the standard transformation services to reach the designated processing service.

A route can contain:

- An in transformer, which is an exit sequence that applies to a message before processing occurs.
  - ❑ A reviewer, which is either the first exit to receive a document after parsing (inbound), or the last exit to receive a document prior to the actual emit operation (outbound). These exits are intended for envelope handling but can be used for any desired purpose.
  - ❑ Validation rules, which apply validation using the rules validation engine. Rules are provided when the iWay Integration Solution for EDI is installed.
  - ❑ A transform, which is a transformation definition file that contains sets of rules, interpreted and executed by a transformation engine. Transformation is the process by which data is transformed from one structure/format to another.
- ❑ A process, which is a stateless, lightweight, short-lived microflow that is executed by iWay Service Manager on a message as it passes through the system. Processes that are published using iIT Designer are available in the Registry and can be bound to channels as routes.
  - □ A process flow.
  - An agent list.
    - A service, which is an executable Java procedure that handles the business logic of a message.

- □ An adapter, which refers to a target that represents a specific instance of a connection to a back-end system.
- An out transformer, which is an exit sequence that applies to a message after processing occurs.
  - ❑ A transform, which is a transformation definition file that contains sets of rules, interpreted and executed by a transformation engine. Transformation is the process by which data is transformed from one structure/format to another.
  - ❑ Validation rules, which apply validation using the rules validation engine. Rules are provided when the iWay Integration Solution for EDI is installed.
  - ❑ A reviewer, which is either the first exit to receive a document after parsing (inbound), or the last exit to receive a document prior to the actual emit operation (outbound). These exits are intended for envelope handling but can be used for any desired purpose.
- ❑ An outlet (optional), which is responsible for all aspects of preparing a document for emission and then emitting it.
  - A preemitter, which is a logical process that handles a document immediately before transmission. Normally it converts an XML document into non-XML format.
  - □ An encryptor, which can be called to encrypt an outgoing document.
  - □ An emitter, which is a transport protocol that sends a document to its recipient.
- An outlet can contain:
- A preemitter.
- An encryptor.
- Multiple emitters.

For details on the preceding components, see the *iWay* Service Manager User's Guide.



# Downloading, Extracting, and Importing HL7 Sample Data

This chapter describes how to download, extract, and import HL7 sample data.

#### In this chapter:

- Downloading and Extracting HL7 Sample Data
- □ Importing HL7 Sample Data to iWay Integration Tools as a Workspace

### Downloading and Extracting HL7 Sample Data

This section describes how to download and extract HL7 sample data.

#### Procedure: How to Download and Extract User Samples for HL7

1. Download the HL7\_usr\_samples.zip file containing HL7 user sample workspace from the following website:

http://techsupport.informationbuilders.com

The downloaded HL7\_usr\_samples.zip contains the following files:

- HL7\_Accelerator.zip
- HL7\_usr\_samples\_ilT\_workspace.zip
- 2. Save the HL7\_usr\_samples\_ilT\_workspace.zip file to a folder on your local drive.

3. Save and extract the HL7\_Accelerator.zip file to a location where you want to store your data, as shown in the following image.



- 4. The HL7\_Accelerator.zip file contains sample input and output data that you can use.
  - □ Inbound test data is located in the following folder:

\HL7\_Accelerator\HL7\_in\IB\_Archive

There is a folder called HL7\_Data.

For example:

🕌 HL7_Data								
C: \HL7_Accelerator \HL7_in \IB_Archive \HL7_Data								
🕘 Organize 👻 📗 Views	•							
Equarita Linka	Name 🔺	-	Date modified	- T	уре	-	Size	
	hl7_adta01c01_001		3/20/2014 9:22 PM	H	IL7 File			3 KB
Documents	hl7_adta04c01_001		3/20/2014 9:22 PM	Н	IL7 File			2 KB
Pictures	hl7_adta08c01_001		3/20/2014 9:22 PM	H	IL7 File			2 KB
🚯 Music	hl7_adta11c01_001		3/20/2014 9:22 PM	Н	IL7 File			1 KB
Recently Changed	hl7_adta13c01_001		3/20/2014 9:22 PM	H	IL7 File			3 KB
	hl7_adta28c01_001		3/20/2014 9:22 PM	H	IL7 File			3 KB
2 Searches	hl7_adta40c01_001		3/20/2014 9:22 PM	H	IL7 File			1 KB
Public	hl7_adta44c01_001		3/20/2014 9:22 PM	H	IL7 File			1 KB
	hl7_barp01c01_001		3/20/2014 9:22 PM	H	IL7 File			3 KB
	hl7_ormo01c01_001		3/20/2014 9:22 PM	Н	IL7 File			2 KB
	hl7_orro02c01_001		3/20/2014 9:22 PM	Н	IL7 File			2 KB
	hl7_orur01c01_001		3/20/2014 9:22 PM	Н	IL7 File			2 KB
	hl7 rdeo11c01 001		3/20/2014 9:22 PM	Н	IL7 File			4 KB

□ Outbound test data is located in the following folder:

```
\LT_Accelerator \LT_out \OB_Archive
```

There is a folder called HL7\_Xml.

For example:

🕌 HL7_xml								
	C:\HL7_Accelerator\HL7_out\OB_Archive\HL7_xml							
🕒 Organize 👻 📗 Views	-							
Eavorite Links	Name 🔺	-	Date modified	-	Туре	-	Size	-
	hl7_adta01c01_001		3/20/2014 9:28 P	М	XML File		21	9 KB
Documents	hl7_adta04c01_001		3/20/2014 9:26 P	М	XML File		17	70 KB
E Pictures	hl7_adta08c01_001		3/20/2014 9:26 P	М	XML File		17	70 KB
Nusic	hl7_adta11c01_001		3/20/2014 9:26 P	М	XML File		6	51 KB
Recently Changed	hl7_adta13c01_001		3/20/2014 9:26 P	М	XML File		21	8 KB
	hl7_adta28c01_001		3/20/2014 9:26 P	М	XML File		19	18 KB
2 Searches	hl7_adta40c01_001		3/20/2014 9:26 P	М	XML File		4	14 KB
Public	hl7_adta44c01_001		3/20/2014 9:26 P	М	XML File		3	31 KB
	hl7_barp01c01_001		3/20/2014 9:25 P	М	XML File		16	9 KB
	hl7_ormo01c01_001		3/20/2014 9:25 P	М	XML File		16	6 KB
	hl7_orro02c01_001		3/20/2014 9:25 P	м	XML File		5	57 KB
	hl7_orur01c01_001		3/20/2014 9:26 P	М	XML File		14	юкв
	hl7_rdeo11c01_001		3/20/2014 9:26 P	М	XML File		18	35 KB

# Importing HL7 Sample Data to iWay Integration Tools as a Workspace

This section describes how to import HL7 sample data to iWay Integration Tools (iIT) as a workspace.

#### *Procedure:* How to Import HL7 User Samples to iWay Integration Tools as a Workspace

- 1. Start iWay Integration Tools (iIT).
- 2. Right-click anywhere inside the Integration Explorer tab and select *Import...* from the context menu, as shown in the following image.

🖂 Integration - iWay Integration Tools	
File Edit Navigate Search Project Run Wir	ndow Help
📬 • 🔛 🗟 🖆 🕯 🖓 🖗 🗰	☆・○・�
🔏 Inte 🔀 😇 iWa 🛋 Libra 🖓 🗖	
수 수 🗟 🗖 🕏 🎽	
New	
Сору	
🛱 Paste	
Duplicate	
🗙 Delete	
inport	
🛃 Export	
🔊 Refresh	

The Import dialog opens, as shown in the following image.

🕺 Import	
Select Create new projects from an archive file or directory.	Ľ
Select an import source:	
type filter text	
General     Archive File     Existing Projects into Workspace     File System     File System     Preferences     CVS     CVS     Preferences     Install     Plug-in Development     Plug-in Development     Plug-in Development     Preferences     XML	
Control C	nish Cancel

3. Expand the General folder, select Existing Projects into Workspace, and then click Next.

The Import Projects pane opens, as shown in the following image.

🛃 Import	
Import Projects Select a directory to search for existing Eclipse projects.	
<ul> <li>Select root directory:</li> <li>Select archive file:</li> <li>Projects:</li> </ul>	Browse Browse
	Select All Deselect All Refresh
Copy projects into workspace  Working sets  Morking sets	Select
	Seject

4. Click Select archive file and then click Browse.

The Select archive containing the projects to import pane opens, as shown in the following image.

🗌 Integration - iWay Integration Tools			
File Edit Navigate Search Project Run Window He	p		
i 🗈 • 🖫 🕼 🗁 i 🍋 🖄 % i 🛲 i 🏇 • 1	🕽 • 💁 • 🗄 🛷 • 🗄 🛃 🙍 Import		
💋 Inte 🛛 😇 iWa 🛋 Libra 🖓 🗆	Select archive contai	ining the projects to import	2 🛛
	Look in: 🛅 il	T-7.0.0 🕑 🔮	) 📂 🛄 •
	My Recent Documents Desktop My Documents My Documents My Documents	Jusr_samples_IIT_workspace.stp           7_usr_samples_IIT_workspace           riguration           :           stures           TA-INF           igins           adire           HL7_usr_samples_IIT_workspace.zip	V Open
	My Network	. Jai, Jaip, Itai, Ital. 92, 192	

5. Select the HL7\_usr\_samples\_ilT\_workspace.zip file and click Open.

You are returned to the Import Projects pane, as shown in the following image.

🕺 Import		
Import Projects Select a directory to searc	:h for existing Eclipse projects.	
<ul> <li>Select root directory:</li> <li>Select archive file:</li> <li>Projects:</li> </ul>	E:\IIT_Builds\JAN 16 2014\iIT-7.0.0\HL7_usr_	Browse
HL7_usr_samples	_prj (HL7_usr_samples_prj)	Select All Deselect All Refresh
Copy projects into working sets	rkspace ng sets	Select
?	<pre>Back Next &gt; Finish</pre>	Cancel

6. Click Finish.

The HL7 user samples are loaded into your iIT workspace, as shown in the following image.



The Integration Explorer tab on the left pane displays a hierarchy of all the imported channel components (for example, Ebixes, listeners, outlets, preparsers, routes, process flows, and so on). The Console tab on the bottom provides a status as each channel component is imported.



# **Configuring the EDI Activity Driver**

This section describes how to configure the EDI Activity Driver using iWay Service Manager.

#### In this chapter:

- EDIHL7 EDI Activity Driver Overview
- Configuring the EDI Data Provider Using iWay Service Manager
- Configuring the EDI Activity Driver Using iWay Service Manager

## EDIHL7 EDI Activity Driver Overview

The EDI Activity Driver is an extension of the Activity Facility in iWay Service Manager. It is used to log events as messages are processed. Logging can occur when:

- a message is acquired.
- a message is emitted.
- an error occurs.

a component such as an agent or process flow is called.

For more information about the Activity Facility, see the iWay Service Manager User's Guide.

Using iWay Service Manager, you must first configure the EDI data provider and then the Activity Facility handler.

#### Configuring the EDI Data Provider Using iWay Service Manager

This section describes how to configure the EDI data provider.

#### Procedure: How to Configure the EDI Data Provider

To configure the EDI data provider:



1. In the left console pane of the Server menu, select *Data Provider*.

The Data Provider pane opens.

#### Data Provider

Listed below are the data provider definitions that are available in the base configuration of this server.

JDBC Connections - JDBC or Java Database platform and a wide range of databases listings below define JDBC connections provider as a DataSource by setting the jdbc/provider name.	Connectivity is a standar , The JDBC interface prov used within iWay Service I initial context factory to (	d for database-independent connec ides a call-level API for SQL-based Manager. iWay components that us com.ibi.jndi.XDInitialContextFactory	tivity between the Java database access. The e JNDI can access a JDBC / and using the name
Name		Driver	
No connections have been define	d		
New			
JLINK Servers - JLINK is a technology that ca The servers listed below are defined for	an be used to access inform ruse with JLINK.	mation hosted by iWay, WebFOCUS	and EDA data servers.
Name	Description		Туре
No servers have been defined			
New			

The tables that are provided list the configured JDBC and JLINK data providers that are available. By default, no data providers are configured.

2. In the JDBC area, click New to configure a new JDBC data provider.

JDBC Connection Pool Pre	operties
Name *	Enter the name of the JDBC data provider to add.
	EDI_Activity_DB
Driver Class	The JDBC driver class is the name of the class that contains the code for this JDBC Driver.
	com.mysql.jdbc.Driver
	Select a predefined database or enter your own.
Connection URL	The JDBC connection URL to use when creating a connection to the target database. The URL generally includes the server name or IP address, the port or service, the data source name, and a driver specific prefix.
	jdbc:mysql://localhost:3306/IWay
	Select a predefined connection URL template or enter your own.
User	User name with respect to the JDBC URL and driver.
	iway
Password	Password with respect to the JDBC URL and driver.
	****
Connection Pool Properti	es
Initial Pool Size *	Number of connections to place in the pool at startup.
	1
Maximum Number of Idle Connections *	Maximum number of idle connections to retain in the pool. O means no limit except what is enforced by the maximum number of connections in the pool.
	1
Maximum Number of	Maximum number of connections in the pool. O means no limit.
Connections *	1
Login Timeout	Time in seconds to wait for a pooled connection before throwing an exception. O means wait forever.

The configuration pane for the JDBC data provider opens.

- 3. In the Name field, enter a name for the new JDBC data provider, for example, EDI\_Activity\_DB.
- 4. From the Driver Class drop-down list, select an appropriate driver or enter the specific driver name (class) that you are using, for example:

```
com.mysql.jdbc.Driver
```

5. From the Connection URL drop-down list, select an appropriate connection URL or enter the specific driver connection URL that you are using, for example:

```
jdbc:mysql://localhost:3306/IWay
```

- 6. In the User field, enter a user name with respect to the JDBC URL and driver.
- 7. In the Password field, enter a password with respect to the JDBC URL and driver.
- 8. In the Initial Pool Size field, enter the number of connections to place in the connection pool during startup.

9. In the Maximum Number of Idle Connections field, enter the maximum number of idle connections to retain in the connection pool.

A value of zero means that there is no limit, except what is enforced by the maximum number of connections in the connection pool.

10. In the Maximum Number of Connections field, enter the maximum number of connections in the connection pool.

A value of zero means that there is no limit.

11. Click Add.

The JDBC data provider that you configured is added to the JDBC Connections list, as shown in the following image.

Data Frovider	Data	Provider	
---------------	------	----------	--

Listed below are the data provider definitions that are available in the base configuration of this server.

JDBC							
Name		Driver					
EDI_Activity_DB		com.mysql.jdbc.Driver					
New Delete Rename Copy							
Servers - JLINK is a technology that c The servers listed below are defined for	an be used to acce r use with JLINK.	ss information hosted by iWay, WebFOCUS	3 and EDA data servers.				
Name	Description		Туре				
No servers have been defined							
New							

# Configuring the EDI Activity Driver Using iWay Service Manager

This section describes how to configure the EDI Activity Driver.

#### *Procedure:* How to Configure the EDI Activity Driver

To configure the EDI Activity Driver:



1. In the left console pane of the Server menu, select Activity Facility.

The Activity Facility pane opens.

Activity Facility Listed below are the activity (sometimes called audit) handlers that have be server has to be stopped and started for any change to take effect.	en configured. You can add to this li	st or delete from it. The
Name	Туре	Active
No activity handlers have been defined		
Add		

The table that is provided lists the configured Activity Facility handlers. Initially, no handlers are shown.

2. Click Add to configure a new Activity Facility handler.

The configuration pane for the Activity Facility handler opens.

Activity			
Туре	The type is the specific class of handler in use		
	EDI Activity Logs		
Name The handler will be known by this name in the system. Names must be unique.			
	EDI Activity Logger		
Description	Describe the purpose of this handler		
Active	Active handlers perform work in the server. Inactive handlers remain defined but are not used during this server		
	run. To change the active state, after updating you must cold restart the server.		
	true		
	Pick one		

- 3. From the Type drop-down list, select EDI Activity Logs.
- 4. Enter a unique name for the EDI Activity Driver and a brief description.
- 5. From the Active drop-down list, select *true*.

6. Configure the JDBC driver for the database you are using.

<b>Configuration Parameter</b>	rs	
JNDI Factory Name JNDI initial context factory class used to access data source. Use com.ibi.jndi.XDInitialContextFa JDBC provider or leave blank for JVM default.		
	com.ibi.jndi.XDInitialContextFactory	
JNDI Name *	JNDI Name for the data source this driver will use. To use an iWay JDBC provider, enter the JNDI name as jdbc/provider name otherwise the defined provider's information will be used.	
	jdbc/EDI_Activity_DB	
Table *	Table name to which to write log.	
	IAM_ACTIVITY	
Compression	What form of compression, if any, should be used on the messages. Compression saves space at the expense of time.	
	none	
	Pick one	

If the database tables do not exist, they will be automatically created when the iSM is restarted.

7. Provide values for the remaining parameters, as defined in the following table.

Parameter Name	Туре	Description
JNDI Factory Name	String	The JNDI initial context factory class that is used to access a data source. Use com.ibi.jndi.XDInitialContextFactor y for an iWay JDBC provider or leave this field blank for the JVM default.
JNDI Name	String	The JNDI name for the data source this driver will use. To use an iWay JDBC provider, enter the JNDI name as jdbc/< <i>data provider</i> <i>name</i> >, where <i>data provider name</i> is the name of the EDI Activity Driver that was specified in step 4. Otherwise the information for the defined provider will be used.
Table	String	Table name for the activity log. This must be a valid identifier in the database being used. If the table does not exist at startup, it will be created automatically.

Parameter Name	Туре	Description
Compression	Drop-down list	Specify whether the messages are to be compressed. Values include:
		none (default)
		smallest
		□ fastest
		□ standard
		Huffman
Start Events	Boolean Drop-down list	If set to <i>true</i> (default), the input messages will be recorded in the activity log. This values must be set to <i>true</i> for use of the audit reports in the console.
Internal Events	Boolean Drop-down list	If set to <i>true</i> , system events are included in the activity log. System events include activities such as parsing and transformations (optional). False is selected by default.
Security Events	Boolean Drop-down list	If set to <i>true</i> (default), security events are recorded. This includes digital signature, and so on. However, console activity is not recorded.
Business Error Events	Boolean Drop-down list	If set to <i>true</i> , business errors are recorded, such as rules system violations. False is selected by default.
Emit Events	Boolean Drop-down list	If set to <i>true</i> (default), output messages from emitter services will be recorded. This is required for use of the audit log reports in the console.

Parameter Name	Туре	Description
End Events	Boolean Drop-down list	If set to <i>true</i> (default), the end of message processing will be recorded in the activity log. This is required for use of the audit log reports in the console.
Notes Table	String	Table name for the notes table, which contains log annotations. If the table does not exist at startup, it will be created automatically.
MAC Algorithm	String Drop-down list	The Message Authentication Code (MAC) algorithm. None (default) indicates a MAC should not be computed.
MAC Provider	String Drop-down list	The Message Authentication Code (MAC) provider. Not Specified indicates the default provider should be used. The remaining available value is <i>SunJCE</i> .
MAC Secret Key	String	The Message Authentication Code (MAC) secret key to use.

8. Click Update.

If necessary, start the database services.

9. Restart iSM to start the EDI Activity Driver and begin logging.

The EDI Activity Driver inserts records into the configured activity database. The records are designed for fast writing rather than for ease of later analysis. A set of inquiry service agents suitable for use in a process flow is available to assist during the analysis of the log. Users are cautioned that iWay does not guarantee the layout of the record from release to release, and this should be checked against the actual schema.

Database Field	Description
recordkey	Unique record identifier.

Database Field	Description	
recordtype	Type of this record - the event being recorded.	
	101 - Message start.	
	131 - Entry to event (see subtype codes below).	
	132 - Normal exit from event.	
	133 - Failed exit from event.	
	151 - Ancillary message (usually rules violation).	
	□ 181 - Emit.	
	191 - Message end.	
signature	Encoding of the listener name and protocol.	
protocol	Name of the protocol.	
address	Address to which an emit is to be issued. The format depends on the protocol.	
tstamp	Timestamp of record.	
correlid	The Message Control ID assigned to this message.	
tid	The Transaction ID assigned to this message.	
msg	Message appropriate to this record type. For example, an input message contains the original message received, if possible. Streaming input does not contain a record.	
context	Serialized special registers that were in the context at the time the record was written.	
text	Message text for business errors (rules system violations).	

Database Field	Description	
status	Status code recorded.	
	O - Success	
	1 - Success, message end (191 record)	
	10 - Rules error	
subtype	Event code for event records.	
	1 - Preparser	
	2 - Parser	
	3 - In reviewer	
	□ 5 - In validation	
	□ 6 - In transform	
	7 - Agent or flow	
	8 - Out transform	
	9 - Out validation	
	□ 11 - Preemitter	
	1000 - input record written to table before transformation	
partner_to	The name of the receiving partner.	
partner_from	The name of the sending partner.	
encoding	Encoding of the listener that obtained the document.	
mac	Not used in this version.	
Driver version	1.0 in 8.0 SM	



# Inbound Processing: HL7 to XML

This section describes how to configure a basic inbound message flow for the iWay Integration Solution for EDIHL7. The message flow represents the movement and tasks in the conversion of a message from HL7 format to XML format.

#### In this chapter:

- Configuring a Channel for HL7 Inbound Processing
- Configuring Register Sets and Registers
- Importing an Ebix Into the Workspace
- Configuring an iWay Integration Application for Inbound Processing
- Setting HL7 System Registers
- Testing the EDIHL7 Inbound Channel Application

## Configuring a Channel for HL7 Inbound Processing

The inbound channel creates an XML representation of a HL7 inbound message, and an acknowledgement message. The documents are routed to designated folders based on the success or failure results of the transformation and HL7 rule validation.

#### Procedure: How to Create a Channel for Inbound Processing

1. Start iWay Integration Tools (iIT).

2. Right-click the Integration Explorer pane, click *New*, and then select *Integration Project* from the context menu, as shown in the following image.



The New Integration Project dialog box opens, as shown in the following image.

A New Integration Project	_ 🗆 X
Integration Project	
Create a new Integration project.	1
Project name EDIHL7_usr_sample_proj	
Project location	
I Use default	
Directory C:\OFFSHORE\IT704PATCHHL7SEP172015\IT-7.0.4\EDIHL7_usr Br	rowse
Additional options	
Create integration folders	
Target Server Version 7.0.4	•
Install additional Target Se	erver Version
(?) Finish	Cancel

3. Enter a name for the Integration Project (for example, *EDIHL7\_usr\_sample\_proj*) in the Project name field, and then click *Finish*.

A new Integration Project node called *EDIHL7\_usr\_sample\_proj* is added to the Integration Explorer.

4. Expand the *EDIHL7\_usr\_sample\_proj* Integration Project node, right-click the *Channels* folder, select *New*, and then click *Channel* from the context menu, as shown in the following image.



5. Enter a name for the Channel (for example, EDIHL7\_IB\_QS\_pFlow\_Channel), and then click *Next*, as shown in the following image.

🦼 Channel Object		_ 🗆 🗵
Channel Gener	al Properties	
Please choose a na	me and location for this new Channel.	
Project Folder	/EDIHL7_usr_sample_proj/Channels	Browse
Name	EDIHL7_IB_QS_pFlow_Channel	
Description		-
		-
Target Cenver Versio		
Target Server versit	Install additional Target Service Version	-
	Create in current folder	

The Inbound/Outbound Protocols pane appears.

6. Click the *Add* button to add a File Listener in the Inbound section, as shown in the following image.

1	Channel Object				
In	Inbound/Outbound Protocols				
1	Specify the inbound and outbound protocols to be used in the channel				
	Inbound:			1 ×	
	Name	Listener Type	Description		
	Have an inlet created for	or each inbound protocol			
				,	
	Outbound:				
	Name	Emitter Type	Description		
	Have an outlet created for each outbound protocol				
	0	< <u>B</u> ack <u>N</u> e	ext > Einish	Cancel	

7. From the Types Filter list, select *File* and then click *Finish*, as shown in the following image.

1		_ 🗆 🗙
Define listener type		
Select type of listener you want to create. Alterna box to filter types.	tevily, use search	
Types filter		
Use '*' to narrow filter matching: a*, *b, a*c		
AQ AS1 AS2 AS2 [nonblocking] Backup Heartbeat Server ConnectDirect Console CS3 Email Envoy Exchange FILE FTP[S] Client (Clear text or SSL FTP Clients) FTP[S] Client (Clear text or SSL FTP Clients) FTP[S] Server (Clear text or SSL FTP Server) HL7-MLLP-Listener		
Accepts documents from files in directories		
?	Finish	Cancel

8. Select the *Have an Inlet created for each inbound protocol* check box to create an Inlet for the channel, as shown in the following image.

A Channel Object			_ O ×	1
Inbound/Outbou	nd Protocols			
Specify the inbound a	and outbound protocols to be	used in the channel		
				1
				1
Inbound:			÷ ×	
Name	Listener Type	Description		
listener.1	Fle	Accepts documents	from files in	
		_		
Have an inlet cre	ated for each inbound proto	:0		
When unchecke	d, a single inlet will be create	d containing all the specifie	d listeners: otherwise.	a separate inlet will be created for each listener.
Outbound:			* ×	
Name	Emitter Type	Description		
Have an outlet of	reated for each outbound pro	otocol		
?	< <u>B</u> ack	Next > Einish	Cancel	

9. Click the Add button in the Outbound section to define an emitter.

	Channel Object							
1	nbound/Outbound	Protocols						
	Specify the inbound and o							
	Inhormalia							
	Potento:	1	1					
	Name	Listener Type	Description					
	Have an inlet created	for each inbound proto	sol loc					
	C. danat							
	Outbounds							
	Name	Emitter Type	Description					
	Have an outlet created for each outbound protocol							
	2	c Back	Next & Fride	Carvel				
	U.	- Back	Cristing Cristing					

The Define emitter type pane appears.

10. In the Types filter list, select *Passthrough* to define the passthrough emitter, and then click *Finish*, as shown in the following image.

Define emitter type	
Select type of emitter you want to create. Alternatevily, use search box to filter types.	
Types filter	_
Use '*' to narrow filter matching: a*, *b, a*c	
FTP[S] Client (Deprecated FTP Clients) HTTP 1.0 [deprecated] HTTP 1.1 [nonblocking] (nhttp) Internal Queue Java Message Service (jmsq) MLLP MQ MQJMS Ordered Queue Passthrough print SFTP Client (Secure Shell version FTP Client)	
Sonic TCP Tibry	
	-
Discards messages	
Einish Cancel	

**Note:** Passthrough does not emit data from the listener. Instead, it just passes the control here and does nothing.
11. Select the *Have an outlet created for each outbound protocol* check box to create an outlet for the channel, and then click *Finish*, as shown in the following image.

1	Channel Object			- <b>D</b> ×					
1 m	Inbound/Outbound Protocols								
Specify the inbound and outbound protocols to be used in the channel									
	Inbound:			÷ ×					
1	Name								
	listener.1	n files in							
- [	Have an inlet created f	or each inbound protocol							
	Outbound:								
	Name	Emitter Type	Description						
	emitter.1	Passthrough	Discards messages						
-									
	✓ Have an outlet created for each outbound protocol								
-									
	2	< Back No	vt > Einish	Cancel					
	J	- Ener							



The Channel Builder pane appears, as shown in the following image.

12. Under *inlet: inlet.1*, click *listener: listener.1* (*File*) and then expand the *Main* (*Missing 2 required fields*) configuration parameter on the right pane, as shown in the following image.

EDIHL7_IB_QS_pFlow_Channel 🛛			
Channel Builder <u>3 errors detected</u>			:
EDBH2_IB_Q5_pflow_channel	Istener.1         Accepts documents from files in directories         Type: File (shange type)         Filter (enter string to filter properties)         Clear         * Hain (Hissing 2 required fields)         Input Path ①         Bestination ①         Removal Destination ①         Suffix In Filter ①         smi         Scan subdirectories ①         false         Do not unzip ZIP files ①         false         Suffix Out ①	i	

13. Enter a valid Input Path, Destination, and Removal Destination (optional) path, then change the Suffix In Filter parameter from *xml* to \* to receive all types of input files (for example, hl7 and txt), and then set the Suffix Out parameter value to *xml*, as shown in the following image.



14. Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.



- 15. Import the *EDIHL7toXML\_pflow\_QS\_AckAgent\_ValidRpt* process flow from the local drive or create one in the EDIHL7\_usr\_sample\_proj directory in the Flows folder.
- 16. Under the *route: route.1(default)* node, select *process: process.1* and click the process icon on the right panel to reference the process flow into your channel, as shown in the following image.



17. Select a process flow from the integration project and then click OK.

A Resource Selection	_ <b>D</b> X
EDIHL7_usr_sample_pr	roj pflow_QS_AckAgent_ValidRpt
?	OK Cancel

**Note:** Process flows should already be built and available in the iIT integration project. They can be found in the EDIHL7\_usr\_sample\_proj directory inside the Flows folder.

For more information, see *Process Flow Used for Inbound Processing: HL7 to XML* on page 81.

Your screen should resemble the following image.

*EDIHL7_JB_QS_pFlow_Ebix_Channel X		٥
Channel Builder Process: process. 1: Process 'process. 1' not defined		
EDIHL7_IB_QS_pFlow_Ebix_Channel	process i 🖉	1
B→B       channek EDP4.7_IB_QS_priory_Ebic_Channel         B→B       inlet: niet.1         L→S       isterner. isterer. 1 (File)         B→D       rotes: cout.1 (default)         IND process: EDIFL7 INIM_priory_QS_Addagent_ValidRpt       Image: cout.1 (default)         B→D       outlet: cout.0.1 (Passtbrough)       Image: cout.1 (default)	Select process from workspace that you want to be referenced by this channel component	]

 Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.

EDIHL7_IB_QS_pFlow_Ebix_Channel 🔀			-
Channel Builder			
EDIHL7_IB_Q5_pFlow_Ebix_Channel		process i	Ø
□       ■	<ul><li><b>¥</b></li><li>♦</li></ul>	Select process from workspace that you want to be referenced by this channel component  EDIH.7_usr_sample_proj/Flows/EDIH.7toXML_pflow_QS_AdvAgent_ValidRpt.iwp/EDIH.7toXML	flo

19. Right-click *inlet: inlet.1*, select *Add Channel Component*, and then click *Preparser* from the sub context menu, as shown in the following image.



20. Click the *change type* link provided in the preparser.1 configuration section, as shown in the following image.



21. From the Types filter list of the Define preparser type pane, select *EDIHL7BatchSplitter* and click *Finish*, as shown in the following image.

ndify preparser type	
Define preparser type	
Select type of preparser you want to create. Alternatevily, use search box to filter types.	
Types filter	
Use '*' to narrow filter matching: a*, *b, a*c	
Append C Char Filter Create Stream Doc PP Cross-Origin Resource Sharing Del Val Del Val Del Val Stream EDIBatchSplitter EDIFACTBrebArsker EDIFACTPreParser EDIFL7PreParser EDIFL7PreParser EDIFL7PreParser EDIX12PreParser EDIX12PreParser EDIX12PreParser EDIX12SplitterPreParser En Tag Error Filter Excel reader	
Splits a batch of HL7 messages	
C Back Next > Finish	Cancel

22. Repeat steps 20 and 21 and choose the *EDIHL7PreParser* preparser type when you are in step 21, as shown in the following image.

A Modify preparser type	_ 🗆 ×
Define preparser type	
Select type of preparser you want to create. Alternatevily, use search box to filter types.	
Types filter	
Lice 's' to parrow filter matching: a* *b. a*c	
Append C Char Filter Create Stream Doc PP Cross-Origin Resource Sharing Del Val Del Val Del Val Stream EDIBatchSplitter EDIFACTBatchSplitter EDIFACTPreParser EDIHL7PetParser EDIHL7PetParser EDIX12SplitterPreParser EDIX12SplitterPreParser En Tag Error Filter Excel reader	
message type	
Sack Next > Finish Ca	ncel

23. Click Finish.

Your screen should resemble the following image.

🔞 locahost 🗧 EDIH.7.JB. QS. pFlow_Ebix_Channel 🕮 " 🗖						
Channel Builder						
EDIHL7_IB_QS_pflow_Ebix_Channel         Image: Second Sec	preparser.2         The HL7 preparser. Accepts a % in the template name, which will get filed in by message type         Type: EDBL7PreParser (com.bl.preparsers.XDEDIH.7PreParser) <b>&gt; Set Condition</b> Filter (enter string to filter properties)         Clear <b>&gt; Main</b> Template (i)         HL7_%_^hoWL.xch         Timestamp (i)         false         Z-segment (i)					

#### Reference: Process Flow Used for Inbound Processing: HL7 to XML

This section provides an overview of the process flow used for inbound processing: HL7 to XML. This process flow (EDIHL7toXML\_pflow\_QS\_AckAgent\_ValidRpt) is already built and available in the iIT integration project. It is located under the *EDIHL7\_usr\_sample\_proj* node inside the Flows subfolder.

The following image shows the entire inbound process flow, including all of the nodes that are used and their connections.



In this process flow, an HL7 formatted document is read from a validation report file. The XML tags are stripped and the document is written to a directory.

**Note:** Using a Catch service (com.ibi.agents.XDCatchAgent) in an EDI flow is not supported. The error handling does not work as a result.

#### *Procedure:* How to Update Basic Details for Channel Components

1. Select the channel component *inlet:inlet.1* from the channel and click the update channel component icon on the top right side panel, as shown in the following image.

\$\$• <b>0</b> • <b>0</b> • <b>0</b> •]\$!•]21•21•5:\$++++		🗄 🙂 Integration
locahost 🗧 "EDIH.7_IB_QS_pFlow_Ebix_Channel 🔯		88-
Channel Builder		
EDIHL7_IB_Q5_pFlow_Ebix_Channel		inlet.1 i
B = B channel: EDH-17_IB_QS_pFlow_Ebic_Channel	<b>∳</b> ∲	and Preparsers.

2. Rename or update the inlet details (for example, XMLEDIHL7\_IB\_QS\_pFlow\_Ebix\_Inlet), and click *OK*, as shown in the following image.



After renaming or changing the channel components, your iIT Channel Builder should resemble the following image.



# **Configuring Register Sets and Registers**

This section describes how to configure register sets and registers using iWay Integration Tools (iIT).

# Procedure: How to Configure Register Sets and Registers

1. In the EDIHL7\_usr\_sample\_proj project tree, right-click the Registers folder, and select *New*, and then click *Register Set* from the context menu, as shown in the following image.

🖉 Inte 🔀 😈	iWa 🖹 Libra 🗋 🗖	🦓 lo	calhost		7_IB_QS_pFlow	_Ebix_Channel	×
Mar south a	÷ ⇒ &   <b>⊇ ⅍ ▽</b>	Ch	annel	Builder			
CDIFIL/_USr_sample_proj     Adapters     Applications			EDIHL7_IB_QS_pFlow_Ebix_Channel				
<ul> <li>├- Channels</li> <li>↓- EDIHL7_IB_QS_pFlow_Ebix_Chanr</li> <li>↓- Ebixes</li> <li>↓- Flows</li> <li>↓- toxML_pflow_QS_AdkAgen</li> </ul>			□- □- □- □- □- □- □- □-				
Registers     New     Schemas     Go Into     Transfor     Open in New Window		Integration Project     Zosspritting Lowrep     Lo		efault) ent_ValidRpt			
	Copy Paste Duplicate Collecte Move Rename Duplicate Move Rename		Appli     Appli     Chan     M     Proce     Trans     Regis     Schei     f() IFL E      Exam	cation inel ess Flow sform ster Set ma Set xpression mple		_pFlow_Ebix_Em	itter (Passthrough)
	Refresh		C) Othe	r	Ctrl+N	ala 👫 Dashlara	_
An outline is not availa	Validate tline is not avaik Validate Run As Debug As Team Compare With Restore from Local History	•	perties Problems     pace Log     bitrar taxt				
		) ) (	age Unhan Unhan Unhan	dled event loop dled event loop dled event loop	Plug-in           event loop exception         org.eclipse.           event loop exception         org.eclipse.           event loop exception         org.eclipse.		
	Properties						

2. In the Name field, enter a name for the register set and click *Finish*, as shown in the following image.

🔬 New Register Set	Wizard	_ 🗆 ×					
General Properties							
Please select a projec	t location and choose a name for the new Register Set						
Project Folder	/EDIHL7_usr_sample_proj/Registers	Browse					
Name	EDIHL7						
Description		-					
		-					
Tacast Canvas Varsian							
Target Server version	7.0.4	•					
	Install additional Target Server Version						
	Create in current folder						
?	Finish	Cancel					
0							

The new register set appears under Registers in the Registers folder, as shown in the following image.

🚽 Integration - EDIHL7_usr_sample_proj/Registers/EDIHL7.iwr/EDIHL7.iwr - iWay Integration Tools									
File Edit Navigate Search Project Run Window Help									
121 • 17 15 🛆 ] 🗞 🎯 🦓 ] 🗛 ]	\$•••••	2 • 8 • • • p • • •							
🔏 Inte 🕮 🙂 IWa 🖻 🗆	localhost 🗧 EDDHL7_18_0	(S_pFlow_Ebix_Channel	ML7.iver 88						
+ + 2 🛛 🖯 🗧 😤									
EDDHL7_usr_sample_proj									
- 🔄 Adapters	Name	Type	Value	Description					
Applications									
😑 🧽 Channels									
EDIHL7_IB_QS_pFlow_Ebix_Chan	e								
- Ebixes									
B- Plows									
EDIHL7toXML_pflow_QS_AdkAgen	1								
E _ Registers									
EDIHL7									
Schemas Transforme									
- Iranstorms									
- APL									
al la la									

3. Click the *Add a property* icon to add a register to the register set, as shown in the following image.

Integration - EDIHL7_usr_sample_proj/Re	gisters/ED	HL7.iwr/EDIHL7.iwr - iWay	Integration Tools			
e Edit Navigate Search Project Run Win	dow Help					
📬 • 🗟 🗞 💧 🖓 🌺 🛛 🛲 🗍	<b>≫•0</b> ·	• 🏊 • 🛛 🛷 • 🗍 🛬 • 🤅	8 • to 🗢 • •			🗈 🔋
🕯 Inte 🔀 🤴 Wa 😫 Libra 🖓 🗖	🔌 localhos	EDIHL7_JB_QS_pFlo	w_Ebix_Channel 📃 EDIHL7.iwr 🖇	3		
						+ 🗆 /* 🗙
EDIHL7_usr_sample_proj			-	Les .	[	lådd a property
- 🗁 Adapters	Name		Туре	Value	Description	provide propertyry
- 😂 Applications						
🖯 🗁 Channels						
EDIHL7_IB_QS_pFlow_Ebix_Chanr						
- 😂 Ebixes						
E Elows						
EDIHL7toXML pflow QS AckAgen						
Registers						
EDIHL7						
Cob Schemas						
Ch Transforms						
Co YM						
2 XHL						

4. Enter a name for the new register in the Name field, select a register type from the Type drop-down list (set to *string* by default), and then enter a value in the Value field, as shown in the following image.

🦼 R	legister	Wizard			×
Ne	w Reg	ister			
En	ter detai	ls regarding t	he new register.		
	Name:	Ack			
	Type:	string			•
	Value:	sreg(EDIHL7	_Input)\OB_Output		•
					A
Desc	ription:				
					<u>*</u>
?				ОК	Cancel

- 5. Click OK.
- 6. Create the following registers under the EDIHL7 registers set you just created, along with the values shown in the table below.

Register Name	Value
Input	Name=Input, Value=sreg(EDIHL7_INPUT)
Error	Name=Error, Value=sreg(EDIHL7_INPUT)\IB_Error
GoodOutput	Name=Output, Value=sreg(EDIHL7_INPUT)\IB_TransformGood
BadOutput	Name=BadOutput, Value=sreg(EDIHL7_INPUT)\IB_Error
ValidReport	Name=ValidReport, Value=sreg(EDIHL7_INPUT)\IB_Report
Archive	Name=Archive, Value=sreg(EDIHL7_INPUT)\IB_Archive

After creating all of the required registers in register set, your iIT screen should resemble the following image.

Integration - EDIHL7_usr_sample_proj/Re	gisters/EDIHL7.iwr/EDIHL7.iwr - iWay	Integration Tools		
Elle Edit Navigate Search Project Bun Wind	dow Help			
🔁 • 🗄 🐁 🛆 ] 🗞 😹 🗶 ] 🛲 ] :	🎄 • 🔕 • 🌡 🖉 • ] 🖗 •	$a \rightarrow + \diamond \rightarrow +$		
🖌 Inte 🛛 😈 iWa 🛤 Libra 🖓 🗆	🗞 localhost 🧧 EDIHL7_I8_QS_pFlo	w_Ebix_Channel		
+ + 4   😑 🗞 🏹				6
EDDHL7_usr_sample_proj				
- 🗁 Adapters	Name	Type	Value	Description
- 🗁 Applications	Input	string	sreg(EDIHL7_INPUT)	HL7 inbound flow scans this directory for HL7 files
Channels	Archive	string	sreg(EDDHL7_INPUT)\JB_Archive	Archive of transformed HL7 files
EDIHL7 IB OS pFlow Ebix Chann	Ack	string	sreg(EDIHL7_INPUT)\O8_Output	Output directory for acknowledgement
- Cob Ebixes	Error	string	sreg(EDDHL7_INPUT)\JB_Error	
E - Co Flows	BadOutput	string	sreg(EDIHL7_INPUT)\JB_Error	XML where ack status is not equal to A(accept)
E TO EDIHL 7toXML offow OS AckAgen	GoodOutput	string	sreg(EDIHL7_INPUT)\JB_TransformGood	XML where ack status equal to A (accept)
Registers	ValdRpt	string	sreg(EDIHL7_INPUT)\JB_Report	Validation Rpt
EDIHL 7		-		
Schemas				
- Cransforms				
Ce YM				

# *Procedure:* How to Add a Register Set to an Inbound Channel as a Dependency

1. Click the XMLEDIHL7\_IB\_QS\_pFlow\_Ebix\_Channel node and then click the Add dependency icon on the far right side panel of the channel properties pane, as shown in the following image.

EDIHL7_IB_QS_pFlow_Ebix_Channel 🔯 🄞 localhost		- [
Channel Builder		
EDIHL7_IB_QS_pFlow_Ebix_Channel	EDIHL7_IB_QS_pFlow_Ebix_Channel i (	2
channek 2014/2 /10. (25 příov, Edv. Charze)         ************************************	Tick the check-box below to enable or disable dynamic routing for this channel.	

2. From the Registers folder under the integration folder, select *EDIHL7* and click *OK*.

Aesource Selection		
合 今 今		
EDIHL7_usr_sample_pro	)j	
- Cer Registers		
EDIHL7		
,		
2	OK	Cancel
		Caricer

The iIT page should resemble the following image.

EDIHL7_JB_QS_pFlow_Ebix_Channel 🛛 🗞 localhost			- 0
Channel Builder			::
EDIHL7_IB_QS_pFlow_Ebix_Channel		EDIHL7_IB_QS_pFlow_Ebix_Channel	i 🙆
B     ■     channel: EDH-7_IB_QS_pFlow_Ebx_Channel       B     ■     ■       B     ■     CS_pFlow_Ebx_Channel       B     ■     ■       B     ■     CS_pFlow_Ebx_Channel       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       Cotten     EDH-7_IB_QS_pFlow_Ebx_Channel       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■     ■       B     ■        B     ■	* * *	Tick the check-box below to enable or disable dynamic routing for this channel.	

3. Click the Save icon to save your changes. You can also use the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

# Importing an Ebix Into the Workspace

This section describes how to import an Ebix into the workspace using iWay Integration Tools (iIT).

# Procedure: How to Import an Ebix Into the Workspace

1. In the integration project EDIHL7\_usr\_sample\_proj, right-click the Ebixes folder and then select *Import* from the context menu, as shown in the following image.



2. Expand the iWay Integration folder, select *Ebix*, and then click *Next*, as shown in the following image.



3. Click the ellipses (...) button to browse and import the ebix from a specific folder location, as shown in the following image.

A Import				
General Proper (2) The name field is	ties Page required.			
Project Folder	/EDIHL7_usr_sam	ple_proj/Ebixes		Browse
Import				
Name				
Description				
Target Server Version	n 7.0.4			-
	Install additional Ta	arget Server Vers	ion	
	Create in curren	nt folder		
?	< <u>B</u> ack	Next >	Einish	Cancel

4. Select *HL7\_2.6 ebix* from the folder location and click *Open*, as shown in the following image.

🦼 Open					X
() - ()	S (C:) + OFFSHORE + Ebix_Builds + H	L7 🔻 🔯 Sea	rch HL7		2
Organize 🔻 New f	older		8≡	- 🗊	0
🚖 Favorites	Name ↑	Date modifie	ed Typ	e	
E Desktop	HL7_2.6.ebx	3/25/2015	5:22 PM EBX	File	
Downloads					
and Reconcruces					
Nesktop					
Cibraries					
Documents					
<ul> <li>Pictures</li> </ul>					
Subversion	-				
Videos					
🔒 Administrator					
Computer					
Network					
Control Panel	• •				
	File name: HL7_2.6.ebx	▼ Ebio	File (*.ebx)		-
			<u>O</u> pen	Cancel	
					11.

- 5. In the Import wizard, click Next.
- 6. Expand the hI7\_2.6 ebix from the left panel and select the 2.6 folder.

7. Select the messages you wish from the right side panel and click *Finish*, as shown in the following image.

Expression to view available ebix entries. Transform ebix are specially designed archive files that contain ebix entries. Ebix entry represents transform configuration and dependencies used by iWay transform engine. Set ebix version to view available ebix entries. Import as System Ebix Transform ebix are specially designed archive files that contain ebix entries. Ebix entry represents transform engine. Import as System Ebix Transform ebix are specially designed archive files that contain ebix entries. Ebix entry represents transform engine. Transform ebix are specially designed archive files that contain ebix entries. Transform ebix are specially designed archive files that contain ebix entries. Ebix entry represents transform engine. Transform to view available ebix entries.	d Import			
Ebix Entries         Select ebix version to view available ebix entries.         Transform ebix are specially designed archive files that contain ebix entries. Ebix entry represents transform configuration and dependencies used by iWay transform engine. Select ebix version to view available ebix entries.         Import as System Ebix         Import as System Ebix         Ebix Entries         Ebix Entries         Ebix Entries         Ebix Entries         Ebix ADT_A03         Ebix ADT_A04         Ebix ADT_A03         Ebix AD	2 mpore			그민스
Select ebix version to view available ebix entries. Transform ebix are specially designed archive files that contain ebix entries. Ebix entry represents transform engine. Select ebix version to view available ebix entries. Import as System Ebix bix Entries Bix	Ebix Entries			
Transform ebix are specially designed archive files that contain ebix entries. Ebix entry care ebix every available ebix entries.         Import as System Ebix         Imp	Select ebix version t	o view available ebix	x entries.	
Transform ebix are specially designed archive files that contain ebix entries. Ebix entry species used by iWay transform engine.         Import as System Ebix         Import abix         Import abix <td></td> <td></td> <td></td> <td></td>				
represents transform configuration and dependencies used by iWay transform engine. Import as System Ebix bix Entries bix Entries bix Entries bix Entries bix Entries	Transform ebix are s	pecially designed are	chive files that contain ebix entries. Ebix er	itry
Select ebix version to view available ebix entries.  Import as System Ebix  bix bix bix bix bix bix bix bix bix	represents transform	n configuration and o	dependencies used by iWay transform engi	ne.
Import as System Ebix         bix       Ebix Entries         Import as Data       ADR_A19         Import as Data       ADR_A19         Import as Data       ADR_A19         Import as Data       ADR_A19         Import as Data       Import as Data         Import as Data       ADR_A19         Import as Data       Import as Data         Import as Data       Im	Select ebix version t	o view available ebix	x entries.	
bix       Ebix Entries         Image: Bit 2.6       Image: Bit 2.6	Import as System	n Ebix		
Data       Data Elinies         Image: Bit 2.6       Image: Bit 2.6	Ibiy		Ehiy Entries	
Image: Second State Sta				
Image: Second	2.6		ADR A19	-
Image: Control of the second secon			ADT_A01	
			ADT_A02	
Image: Control of the second seco			ADT_A03	
→ Back       Mext >       Einish       Cancel			ADT A05	
ADT_A07			ADT_A06	
Cancel			ADT_A07	
Description: Entry: ADT_A05 Run Time Mode: N/A Description:			ADT_A08	
Description: Entry: ADT_A05 Run Time Mode: N/A Description:			ADT A10	
Description: Entry: ADT_A05 Run Time Mode: N/A Description:			ADT_A11	
Description: Entry: ADT_A05 Run Time Mode: N/A Description:	1		ADT A12	
Entry: ADT_A05 Run Time Mode: N/A Description:	Description:			
Run Time Mode: N/A       Description:       Image: Concel       Image: Concel	Entry: ADT_A05			^
Cancel	Run Time Mode: N/	'A		
Cancel	Description:			-
< Back     Mext >     Einish     Cancel				
< Back				
Cancel				
Cancel     Can				
	<b>(()</b>	< <u>B</u> ack	Next > Einish Ca	ncel



The iIT page should resemble the following image.

#### *Procedure:* How to Add an Ebix to an Inbound Channel as a Dependency

1. Click the *channel: XMLEDIHL7\_IB\_QS\_pFlow\_Ebix\_Channel* node and then click the *Add dependency* icon on the right side panel, as shown in the following image.

EDIHL7_IB_QS_pFlow_Ebix_Channel 🕮 🔞 localhost				- 8
Channel Builder				::
EDIHL7_IB_QS_pFlow_Ebix_Channel  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel  Channel: EDIHL7_IB_QS_pFlow_Ebix_Inlet  Channel: EDIHL7_IB_QS_pFlow_Ebix_Listener (File)  Preparser: EDIHL7_IB_QS_pFlow_Ebix_Route (default)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Route (default)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Route (default)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel: Ebix_IIIII (IIIII)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel: Ebix_IIIII (IIIII)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel: Ebix_IIIII (IIIII)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel: Ebix_IIIII (IIIII)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel: Ebix_IIIIII (IIIIII)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Channel: Ebix_IIIIII (IIIIIIIII)  Channel: EDIHL7_IB_QS_pFlow_Ebix_Ebix_Ebix_IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	<ul> <li></li> <li></li></ul>	EDIHL7_IB_QS_pFlow	Ebix_Channel to enable or disable dynamic routing for this channel. encies, such as Ebixes and Registers: Location	i Ø

2. Expand *EDIHL7\_usr\_sample\_proj*, *Ebixes*, *HL7*, and then select ebix *hI7\_2.6* and click *OK*, as shown in the following image.

Aresource Selection		_ <b>D</b> ×
EDIHL7_usr_sample_proj	1	
?	ок	Cancel

3. Click the Save button to save your changes or press the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

Your iIT workspace should resemble the following image.

EDIHL7_I8_QS_pFlow_Ebix_Channel 🛛 💊 localhost	
Channel Builder	
EDIHL7_IB_QS_pFlow_Ebix_Channel	EDIHL7_IB_QS_pFlow_Ebix_Channel i
B	Tick the check-box below to enable or disable dynamic routing for this channel.         Dynamic Routing         Manage channel dependencies, such as Ebixes and Registers:

# Configuring an iWay Integration Application for Inbound Processing

This section describes how to configure an iWay Integration Application (iIA) for inbound processing using iWay Integration Tools (iIT).

## Procedure: How to Configure an iWay Integration Application for Inbound Processing

1. Right-click the Applications folder under the EDIHL7\_usr\_sample\_proj integration project, select *New*, and then click *Application* from the context menu, as shown in the following image.



2. Enter a value in the Name field (for example, *EDIHL7\_usr\_sample\_App*), and click *Next*, as shown in the following image.

A New Application Wizard		_ 🗆 ×
General Properties		
Please select a project locatio	n and choose a name for the new application	
Project Folder	/EDIHL7_usr_sample_proj/Applications	Browse
Name	EDIHL7_usr_sample_App	
Description		<u> </u>
		v
Target Server Version	7.0.4	Ŧ
	Install additional Target Server Version	
	Create in current folder	
	Use Maven	
		const 1
(I)	< Back Next > Finish	Cancel

3. Select the *XMLEDIHL7\_IB\_QS\_pFlow\_Ebix\_Channel* check box from Resource Selection pane and keep clicking *Next* until you get to the Bindings pane, as shown in the following image.

A New Application Wizard					_ O ×
Resource Selection					
Add channels, transforms and proc	cesses to your	applicati	on.		
News		T		Lemme	
FDTHL7 TB OS oFlow Fbb	c Channel	inlin	Aut	EDIH 7 um	Select All
TO EDIHL/ToXML of ow OS A	ckAgent Valid	proc	100	/EDIHL7_Usr	Decelect All
		procini		1000101_000	Desective
•				<b>)</b>	
				_	
3	< Back	Next	>	Finish	Cancel

4. In the Bindings pane, add Bindings for Registers you have added in the XMLEDIHL7 Register Set to the iIA EDIHL7\_usr\_sample\_App and click *Finish*, as shown in the following image.

🔏 New Applicatio	n Wizar	d			<u> </u>
Bindings					l l
List the application resolved at deploy	bindings ment time	, such as Special R e.	egisters and Provi	iders, to be	
Name	Type	Default Value	Description		_
	1/25	Derout Forde	occorpton.		- 1
2					×
Nev	Appli	cation Bindin	ig		
Ente	er details	regarding the new	application bindin	9	
	Name:	Adk			
	Type:	Register			-
Defa	lt Value:	sreg(EDIHL7_IN	PUT)\OB_Output		
					*
De	scription:				-
	)			ОК	Cancel
?		< Back	Next >	Finish	Cancel

After adding all register bindings, your iIT screen should resemble the following image.

New Applicat	ion Wizard		
ndings			
ist the application	on bindings, s	uch as Special Registers and Providers, to be	
esolved at deplo	oyment time.	• •	
	,		
Name	Туре	Default Value	Description
Ack	Register	sreg(EDIHL7_INPUT)\OB_Output	نييتني الم
Error	Register	sreg(EDIHL7_INPUT)\IB_Error	1*
ValidRpt	Register	sreg(EDIHL7_INPUT)\IB_Report	
GoodOutput	Register	sreg(EDIHL7_INPUT)\IB_TransformGood	- 6 <sup>6</sup>
BadOutput	Register	sreg(EDIHL7_INPUT)\IB_Error	
Input	Register	sreg(EDIHL7_INPUT)	× .
Archive	Register	sreg(EDIHL7_INPUT)\IB_Archive	
			And a second design of the second s

5. To build the application, right-click *EDIHL7\_usr\_sample\_App*, select *Integration Tools*, and then click *Build* from the context menu, as shown in the following image.

🚀 Inte 🛛 😈 Wa 🛋 Libra ⊂ 🗖	EDIHL7_IB_QS_pFlow_Ebix_Channel	EDIHL7.iwr	SEDIHL 7_US	r_sample_App.iab 🛛	
(+ + &   ⊟ 🕸 ▽	Add channels, transforms and processes	to your application.	Autostart column le	ts you control channel start behavior on serv	er startup.
EDIHL7_usr_sample_proj	Name	Туре	Auto Start	Location	Description
Adapters     Adapters     Components     Diff. 7 usr_sample_App     Diff. 7 usr_sample_App     Diff. 7 usr_sample_App     Diff. 7 usr_sample_App.iii     Diff. 7 usr_sample_App.iii	EDIHL7_IB_QS_pFlow_Ebix_Channel	inlineChannel	yes	/EDIHL7_usr_sample_proj/Channels/	
Channels     B EDIHL7_IB_QS_pFlow_Ebix_Channels     DiHL7_IB_QS_pFlow_Ebix_Channel     Discrete					
EDIHL7toXML_pflow_QS_AdkAgen C ← Registers EDIHL7 ← Schemas					
- 🧀 Transforms - 🧀 XML					
<u>د ا</u>	Components Process Business Services Lib	oraries Resources	Bindings		
E Outine 🛛 🗖	Properties 🔮 Error Log 📮 Console	🔀 🔣 Problems			
An outline is not available.	IT Message Console				
	[INFO]06:20:35 Process flow [INFO]06:44:30 Application	<pre>/ 'EDIHL7toXN 'EDIHL7 usr</pre>	L_pflow_QS_ sample App'	AckAgent_ValidRpt' compiled built successfully.	successfully

- 6. To publish the iIA, right-click EDIHL7\_usr\_sample\_App, select *Integration Tools*, and then click *Publish to* from the context menu.
- 7. Provide the iSM server details in the Server URL text box and the other credentials, then click *Finish*.

The iIT page should resemble the following image.

te cui nangate search project par min	NOW HOP						
🔁 • 🕫 🗞 🛆 ] 🗞 📓 🦓 🖉 🖉	🌼 • 🔘 • 🗛 • 🛛 🛷 • 🗍 🥹 • 🤅	• + + +	÷ •			🔡 😈 Integ	ration
💰 Inte 🔀 🤴 Ma 🛋 Libra 🖓 🗖	EDIHL7_IB_Q5_pFlow_Ebix_Channel	EDDHL7.iwr	G EDDHL 7_us	r_sample_App.iab 💠			• 0
~ -> @   🖻 🎕 🗸	Add channels, transforms and processes	to your application.	Autostart column le	ts you control channel start behavior on serv	er startup.		
EDIHL7_usr_sample_proj	Name	Type	Auto Start	Location	Description		and the
Applers     Appleations	EDDHL7_JB_QS_pFlow_Ebix_Channel	inineChannel	yes	/EDIHL7_usr_sample_proj/Channels/			<u> </u>
G G EDIHL7_usr_sample_App							×
🕀 🥭 Components							_
EDD1.7 ust sample Ann.ia							
🖯 🗁 Channels							
EDIHL7_IB_QS_pFlow_Ebix_Chanr							
Grad Educes     Grad Educes     Grad Educes							
EDIHL 7toXML_pflow_QS_AckAgen							
Constant      Constant							
EDIPL7							
- 🗁 Transforms							
🗁 XML							
	1						
	Components Process Business Services Lib	raries Resources	Bindings				
E Outine 💠 🔍 🗆	🗆 Properties 🕙 Error Log 📮 Console	😂 😫 Problems				🗋 🔓 📑 🖻 - 📬	•
in outline is not available.	IT Message Console						
	[INF0]06:20:35 Process flow [INF0]06:44:30 Application	'EDIHL7toX	sample App'	Ackagent_validRpt' compiled built successfully.	anccessfully		~
ſ	[INFO]07:55:10 Application	'EDIHL7_usr	sample_App'	successfully built and pub	lished to http://iwh17:9000		
			_				

8. Deploy the iIA application on the iSM registry.

9. Have the following folder structure created before starting your application in the iSM console, as shown in the following image.



# Setting HL7 System Registers

This section describes how to set system registers using the iWay Service Manager (iSM) Administration Console.

## Procedure: How to Set System Registers

- 1. Open the iSM Administration Console and select *EDIHL7\_usr\_sample\_App\_IB* [down] from the Management drop-down list.
- 2. From the console bar, click Server, Register Settings, and then click Add, as shown in the following image.

Properties General Properties Java Properties	Register Settings Special registers are in available to all compon Listed below are the r	named variables that reference values which nents of the system. Any changes to the reg egister settings for the HL7_usr_samples_Ap	are carried throughout the system. Once defined, ister settings do not take effect until the server is r p configuration of this server.	these variables t estarted/redeplo
Settings	□ Name	¥alue	Description	Туре
Console Settings	iwayversion	unavailable	system defined (readonly)	string
Java Settings	iwayhome	unavailable	system defined (readonly)	string
Register Settings	iwaydata	unavailable	system defined (readonly)	string

3. For the deployed EDIHL7\_usr\_sample\_App\_IB application, define the system registers as listed in the following table:

Register Name	Value
EDIHL7_INPUT	<pre>sreg(EDIHL7_Installdir)\HL7_in</pre>
EDIHL7_installdir	C:\EDIHL7_Accelerator
EDIHL7_OUTPUT	<pre>sreg(EDIHL7_Installdir)\HL7_out</pre>
ValidateEDIHL7	true

Special Register De	finition
Name *	Enter the name of the special register to add.
	EDIHL7_Installdir
Туре	Select a type for the value of this special register.           string
Value *	Enter a value for this special register. The value can be a constant or a call to the evaluation functions. C:\EDIHL7_Accelerator
Description	Enter a description for this special register.

After adding the registers, your Register Settings page in the iSM Administration Console should resemble the following image.

iway.serverfullhost	iwhl7.ibi.com	system defined (readonly)	string
iway.pid	1668	system defined (readonly)	string
EDIHL7_INPUT	sreg(EDIHL7_Installdir)\HL7_in		string
EDIHL7_Installdir	C:\EDIHL7_Accelerator		string
EDIHL7_OUTPUT	sreg(EDIHL7_Installdir)\HL7_out		string
ValidateEDIHL7	true		string

4. Start the deployed application in the iSM Administration Console and ensure that the channel is up and running in the Monitoring section of the console.

# Testing the EDIHL7 Inbound Channel Application

This section describes how to test the inbound channel application.

#### Procedure: How to Test the Inbound Channel Application

1. Place your input files under EDIHL7\_Accelerator\HL7\_in, as shown in the following image.

DIHL7_Acce	lerator + HL7_in +			
-	Name ^	Date modified	Туре	Size
	3 IB_Archive	10/1/2015 11:31 AM	File folder	
	IB_Error	10/1/2015 11:32 AM	File folder	
	IB_Output	10/1/2015 11:32 AM	File folder	
	IB_Report	10/1/2015 11:32 AM	File folder	
	IB_TransformGood	10/1/2015 11:32 AM	File folder	
	DB_Archive	2/12/2015 6:36 AM	File folder	
1	DB_Error	2/12/2015 6:36 AM	File folder	
	B_Output	10/1/2015 11:32 AM	File folder	
	DB_Report	2/12/2015 6:36 AM	File folder	
	DB_TransformGood	2/12/2015 6:36 AM	File folder	
	75 hl7_adta01c01_001_Sample.hl7	9/23/2015 9:50 AM	HL7 Files	3 KB

2. Observe the transformed XML output under EDIHL7\_Accelerator $HL7_in \IB_TransformGood$ , as shown in the following image.

EDIHL7_Accelerator • HL7_in • IB_TransformGood										
re with 🔻	New folder									
<b>_</b>	Name ^	Date modified	Туре	Size	ĺ					
	hl7_adta01c01_001_Sample2015-10-01T15_43_21_9532001.xml	10/1/2015 11:43 AM	XML File	20 KB						

3. Monitor the reports under *EDIHL7\_Accelerator*\*HL7\_in*\*IB\_report*, as shown in the following image.



4. Observe the Acknowledgement under *EDIHL7\_Accelerator*\*HL7\_in*\*OB\_Output*, as shown in the following image.

EDIHL7_Accelerator + HL7_in + OB_Output										
re with 🔻	<ul> <li>New folder</li> </ul>									
<u> </u>	Name	•	Date modified	Туре	Size					
	hl7_adta01c01_001_Sample2015-10-01T15_43_21_9722002.	nl7	10/1/2015 11:43 AM	HL7 Files	1 KB					

- 5. If the input data contained any errors, you can review this error data in the output directory that you have configured for error handling (for example, EDIHL7\_Accelerator\HL7\_in \IB\_Error).
- 6. After inbound processing has completed, a copy of the input data that was used during the transformation is stored under the archive directory that you specified (for example, EDIHL7\_Accelerator\HL7\_in\IB\_Archive).

# Chapter 6

# Inbound Processing: HL7 to XML (Using MLLP)

This section describes how to configure a basic inbound message flow for the iWay Integration Solution for EDIHL7 using the Minimal Lower Layer Protocol (MLLP). The message flow represents the movement and tasks in the conversion of a message from HL7 format to XML format.

#### In this chapter:

- Configuring a Channel for HL7 Inbound Processing Using MLLP
- Configuring Register Sets and Registers
- Importing an Ebix Into the Workspace
- Configuring an iWay Integration Application for Inbound Processing
- □ Setting HL7 System Registers Using MLLP
- Testing the Inbound Channel Application Using MLLP

# Configuring a Channel for HL7 Inbound Processing Using MLLP

The inbound channel creates an XML representation of a HL7 inbound message, and an acknowledgement message. The documents are routed to designated folders based on the success or failure results of the transformation and HL7 rule validation.

## Procedure: How to Create a Channel for Inbound Processing

1. Start iWay Integration Tools (iIT).

2. Right-click the Integration Explorer pane, click *New*, and then select *Integration Project* from the context menu, as shown in the following image.


The New Integration Project dialog box opens, as shown in the following image.

A New Integration Project	×
Integration Project	
Create a new Integration project.	
Project name EDIHL7_usr_sample_proj	and the second
Project location	1
I ✓ Use default	
Directory C:\OFFSHORE\IT704PATCHHL7SEP172015\IT-7.0.4\EDIHL7_usr_ Browse	
Additional options	
✓ Create integration folders	
Target Server Version 7.0.4	
Install additional Target Server Versio	n
Finish Cancel	

3. Enter a name for the Integration Project (for example, *EDIHL7\_usr\_sample\_proj*) in the Project name field, and then click *Finish*.

A new Integration Project node called *EDIHL7\_usr\_sample\_proj* is added to the Integration Explorer.

4. Expand the *EDIHL7\_usr\_sample\_proj* Integration Project node, right-click the *Channels* folder, select *New*, and then click *Channel* from the context menu, as shown in the following image.



5. Enter a name for the Channel (for example, XMLEDIHL7\_IB\_QS\_pFlow\_MLLP\_Channel), and then click *Next*, as shown in the following image.

A Channel Object		_ 🗆 🗙
Channel Genera Please choose a name	I Properties and location for this new Channel.	1
Project Folder	/EDIHL7_usr_sample_proj/Channels	Browse
Name	EDIHL7_IB_QS_pFlow_MLLP_Channel	
Description		*
Target Server Version	7.0.4	-
	Install additional Target Server Version	
?	< Back Next > Finish	Cancel

The Inbound/Outbound Protocols pane appears.

6. Click the *Add* button to add a File Listener in the Inbound section, as shown in the following image.

1	Channel Object					
Ir	bound/Outbound P	rotocols				
1	Specify the inbound and ou	tbound protocols to be used	in the channel			
	Inbound:			<u>.</u>		
	Name	Listener Type	Description			
	Have an inlet created for	or each inbound protocol				
	Outbound:					
		(	( <b>D</b>			
	Name	Emitter Type	Description			
	Have an outlet created	for each outbound protocol				
—	~					
	?	< <u>B</u> ack	ext > Einish	Cancel		

7. From the Types Filter list, select *MLLP* and then click *Finish*, as shown in the following image.

d		_ 🗆 X
Define listener type		
Select type of listener you want to create. Alterna box to filter types.	tevily, use search	
Types filter		
Use '*' to narrow filter matching: a*, *b, a*c		
Exchange File FTP[S] Client (Clear text or SSL FTP Clients) FTP[S] Client (Deprecated FTP Clients) FTP[S] Server (Clear text or SSL FTP Server) HL7-MLLP-Listener HTTP 1.0 [deprecated] HTTP 1.1 [nonblocking] (nhttp) iEI Internal Queue Java Message Service (jmsq) LDAP High Watermark/File LDAP Listener LogListener MOL		
Minimal Lower Layer Protocol for HL7		
•	Finish	Cancel

8. Select the *Have an Inlet created for each inbound protocol* check box to create an Inlet for the channel, as shown in the following image.

A Channel Object			_ <b>D</b> >	a .
Inbound/Outbo Specify the inbound	aund Protocols d and outbound protocols to b	e used in the channel		
Inbound:			<b></b>	-
Name	Listener Type	Description	1	
listener.1	MLLP	Minimal Lower Layer Proto	ocol fo	
When unchecke Outbound: Name	reated for each inbound proto d, a single inlet will be created Emitter Type	containing all the specified listene Description	ers; otherwise, a	e separate inlet will be created for each listener.
Have an outlet	created for each outbound pr	rotocol		

9. Click the *Add* button in the Outbound section to define an emitter, as shown in the following image.

Listener Type	used in the channel
Listener Type MLLP	Description
Listener Type MLLP	Description
Listener Type MLLP	Description
MLLP	
	Minimal Lower Layer Protocol fo
ted for each inbound protoc	Description
ated for each outbound pro	stocol
	Emitter Type

10. In the Types filter list, select *Passthrough* to define the passthrough emitter, and then click *Finish*, as shown in the following image.

A Contraction of the second se	
Define emitter type	
Select type of emitter you want to create. Alternatevily, use search box to filter types.	
Types filter Use '*' to narrow filter matching: a*, *b, a*c FTP[S] Client (Deprecated FTP Clients) HTTP 1.0 [deprecated] HTTP 1.1 [nonblocking] (nhttp) Internal Queue	<b>_</b>
Java Message Service (jmsq) MLLP MQ MQJMS Ordered Queue Passthrough	_1
print SFTP Client (Secure Shell version FTP Client) Sonic TCP Tibrv	*
Discards messages	
? Einish	Cancel

**Note:** Passthrough does not emit data from the listener. Instead, it just passes the control here and does nothing.

11. Select the *Have an outlet created for each outbound protocol* check box to create an outlet for the channel, and then click *Finish*, as shown in the following image.

Channel Object	ê.	
bound/Outbourd	and Protocols and outbound protocols to be	used in the channel
Inbound:		<b>*</b> ×
Name	Listener Type	Description
listener.1	MLLP	Minimal Lower Layer Protocol fo
Have an inlet or     Outbound:     Name     emitter.1	eated for each inbound protoc	col E Description Discards messages
I▼ Have an outlet o	created for each outbound pri	otocol

The Channel Builder pane appears.

🔏 Inte 🕄 😈 Wa 🛋 Libra 🐃 🗆	EDIHL7_IB_QS_pFlow_MLIP_Channel 🛛					- D
	Channel Builder Process: process. 1: Process 'process. 1' not defined					
Bit Constants     Account of the Dool     Account	CDRH.7_IB_QS_prive_MLP_Channel         □       Channel: 2016 2 II QS prive_MLP_Charge         □       <	<b>₽</b> <b>X</b>	EDIHL7_IB_QS_pFlow_HI Tick the check-box below to O Dynamic Routing Manage channel dependence Type	LIP_Channel enable of daable dynamic routing for this channel. es, such as Eboxes and Registers: Location	i	

12. Under *inlet: inlet.*1, click *listener: listener.*1 (*MLLP*) and then expand the *IP Properties* configuration parameter on the right pane, as shown in the following image.

EDIHL7_IB_QS_pFlow_MLLP_Channel 🛛			- 6
Channel Builder Process: process. 1: Process 'process. 1' not defined	ned		
EDIHL7_IB_Q6_pflow_HLLP_Channel □ -		Istener.1 Minimal Lower Layer Protocol for H_7 Type: MLIP chance hore: Filter (enter string to filter properties) Clear incagroup.main IP Properties Port () 0 Local Bind Address () Persistent Connection () True Haximum Connections () 0 Persistence Timeout value in Hinutes () 0 Set Besenous Hofbelay ()	i Ø 🗞

13. Enter a valid Port number (for example, 778) to listen for the input HL7 file, as shown in the following image.

		1
EDBL7_IB_QS_priow_MLLP_Channel Channel: EDDM.7_IB_QS_priow_MLLP_Channel Channel: EdDM.7_IB_QS_priow_MLLP_Channe	Interer.1         Minimal Lower Layer Protocol for H.7         Type: MLP chance hose.         Filter (enter string to filter properties)         Image: String to filter properties         Image: String to filter properties	i ⊘   % 

14. Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.



- 15. Import the *XMLToEDIHL7\_Ebix\_2\_MLLP\_AckAgent\_ValidRpt* Process Flow from the local drive or create the one in the EDIHL7\_usr\_sample\_proj directory in the Flows folder.
- 16. Under the *route: route.1(default)* node, select *process: process.1* and click the process icon on the right panel to reference the process flow into your channel, as shown in the following image.



17. Select a process flow from the integration project and then click OK.

A Resource Selection	_ 🗆 🗙
EDIHL7_usr_sample_proj	ValidRpt
ОК	Cancel

**Note:** Process flows should already be built and available in the iIT integration project. They can be found in the EDIHL7\_usr\_sample\_proj directory inside the Flows folder.

For more information, see *Process Flow Used for Inbound Processing: HL7 to XML (MLLP)* on page 123.

Your screen should resemble the following image.

DIHL7_IB_Q5_pFlow_MLLP_Channel 🖾 💊 localhost	- 8
Channel Builder Process: process. 1: Process 'process. 1' not defined	
EDIHL7_IB_QS_pFlow_MLLP_Channel	process i  Select process from workspace that you want to be referenced by this channel component CDIHL7_usr_sample_proj/Flows/EDIHL7toXML_pRow_MLLP_AckAgent_ValkRpt.iwp/EDIHL7toX

18. Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.

EDIHL7_IB_QS_pFlow_MLLP_Channel 🔀 💊 localhost	
Channel Builder	
EDIHL7_IB_QS_pFlow_MLLP_Channel	process     i     i       Select process from workspace that you want to be referenced by this channel component       image: the select process of the select proces of the select process of the select proces of the select proces

19. Right-click *inlet: inlet.1*, select *Add Channel Component*, and then click *Preparser* from the context menu, as shown in the following image.



You iIT screen should resemble the following image.

*EDIHL7_IB_Q5_pFlow_MLLP_Channel X  localhost		- E
Ochannel Builder preparser: preparser.1: specify component type		
EDIHL7_IB_Q5_pFlow_MLLP_Channel	preparser.1	i 🛛 🖄
	A logical process that handles docum to convert from non-Will to xml. Type: undefined change type.	ents before they are parsed by the system. Usually used

20. In the preparser.1 configuration pane, click the *change type* link and select *EDIHL7Preparser* from the Types filter list and click on *Finish*, as shown in the following image.

🛃 Modify preparser type	- 🗆 ×
Define preparser type	100
Select type of preparser you want to create. Alternatevily, use search box to filter types.	
Types filter	
Lice "#" to parrow filter matching: a#_#b_a#c	
Append C Char Filter Create Stream Doc PP Cross-Origin Resource Sharing Del Val Del Val Del Val Stream EDIBatchSplitter EDIFACTBatchSplitter EDIFACTPreParser EDIHL7BatchSplitter EDIHL7PreParser EDIX12PreParser	•
The HL7 preparser. Accepts a % in the template name, which will get filled message type	in by
(?) < Back Next > Finish C	ancel

The following image shows the iIT screen.

EDIHL7_IB_Q5_pFlow_MLLP_Channel	preparser.1
Channel: EDIAT. JB. (25. pFlow, MLP_Channel S ≪ Thet: net. 1 G v The preparate preparate. Channel: The preparate preparate. Channel: EDIAT. JB. (25. pFlow, MLP) Channel: EDIAT. State of the preparate preparate. Channel: EDIAT. JB. (25. pFlow, MLP) Channel: EDIAT. JB. (25. pFlow, MLP) Chan	The HL7 preparser. Accepts a % in the template name, which will get filled in by message type Type: EDIH.7PreParser (com.bb.preparsers:/DEDIH.7PreParser) <u>chance type</u> Set Condition Filter (enter string to filter properties) Clear  Wain Template HT7_%_^toXML.sch Timestamp false Z-segment false Segment fille

#### *Reference:* Process Flow Used for Inbound Processing: HL7 to XML (MLLP)

This section provides an overview of the process flow used for inbound processing: HL7 to XML (MLLP). This process flow (XMLToEDIHL7\_Ebix\_2\_MLLP\_AckAgent\_ValidRpt) is already built and available in the iIT integration project. It is located under the *EDIHL7\_usr\_sample\_proj* node inside the Flows subfolder.

The following image shows the entire inbound process flow, including all of the nodes that are used and their connections.



In this process flow, an HL7 formatted document is read from a validation report file. The XML tags are stripped and the document is written to a directory.

**Note:** Using a Catch service (com.ibi.agents.XDCatchAgent) in an EDI flow is not supported. The error handling does not work as a result.

#### Procedure: How to Update Basic Details for Channel Components

1. Select the channel component *inlet:inlet.1* from the channel and click the update channel component icon on the top right side panel, as shown in the following image.



2. Rename or update the inlet details (for example, EDIHL7\_IB\_QS\_pFlow\_MLLP\_Inlet), and click *OK*, as shown in the following image.

A Modify channel component properties	×
Properties for inlet.1	
Change properties for inlet.1	
Name	
EDIHL7_IB_Q5_pFlow_MLLP_Inlet	
Description	
Inlets are conduits which represent the entry into a channel. Inlets contain a Listener, Decryptor,	, and Preparsers. 📩
ОК ОК	Cancel

After renaming or changing the channel components, your iIT Channel Builder should resemble the following image.

EDIHL7_IB_QS_pFlow_MLLP_Channel 🛛 🔞 localhost	
Channel Builder	
EDIHL7_IB_QS_pFlow_MLIP_channel  Ediannel: EDIHL7_IB_QS_pFlow_MLIP_channel  ediannel: EDIHL7_IB_QS_pFlow_MLIP_channel  edianter: EDIHL7_IB_QS_pFlow_MLIP_latener (MLIP)  edianter: EDIHL7_IB_QS_pFlow_MLIP_latener (MLIP_latener (MLIP)  edianter: EDIHL7_IB_QS_pFlow_MLIP_latener (MLIP_latener (MLIP))  edianter: EDIHL7_IB_QS_pFlow_MLIP_latener (MLIP)  edianter: EDIHL7_IB_QS_pFlow_MLIP_IAtener (MLIP)  edianter	EDIHL7_IB_QS_pFlow_MLLP_Channel         Tick the check-box below to enable or disable dynamic routing for this channel.         Ivpanic Routing         Manage channel dependencies, such as Ebixes and Registers:         Type         Location

# **Configuring Register Sets and Registers**

This section describes how to configure register sets and registers using iWay Integration Tools (iIT).

## Procedure: How to Configure Register Sets and Registers

1. In the EDIHL7\_usr\_sample\_proj project tree, right-click the Registers folder, and select *New*, and then click *Register Set* from the context menu, as shown in the following image.

🔏 Integrati 🕅	🕴 🙂 iWay Ex 📑 Library M		EDIHL7_IB_QS_pFlow_I	MLLP_Channel 🔀 🍓 localhost
EDIHL7_us Adaptic Applica C Anno Ebixes Applica Applic	← → & = sample_proj ers stions els IHL7toXML_pflow_MLLP_AckAgent_	validRpt	Channel Builder	MLLP_Channel M.7_IB_QS_pFlow_MLLP_Channel M.7_IB_QS_pFlow_MLLP_Inlet pr: EDIHL7_IB_QS_pFlow_MLLP_Listener (MLLP) seen_FDIHL7_IB_QS_pFlow_MLLP_Listener (MLLP)
Cutline 15 not av	New       Go Into         Open in New Window       Open in New Window         Copy       Paste         Duplicate       Paste         Duplicate       Paste         Move       Rename         Import       Paste         Refresh       Validate         Run As       Debug As         Debug As       Team         Compare With       Restore from Local History         Source       Paste	Appli Chan Strans Regis Chan Strans Regis Chan Regis Chan Chan Chan Chan Chan Chan Chan Chan	ration Project cct ccation nel ess Flow sform ster Set xpression r Ctrl+N	Ser: EUIRI, _IE_QS_priow_MLP_reparser [7_IB_QS_priow_MLP_Route (default) EDIFIT_TOXINI_priow_MLP_AckAgent_ValidRpt U7_IB_QS_priow_MLP_Outlet EDIFIT_IB_QS_priow_MLIP_Emitter (Passthrough)
	Properties			

2. In the Name field, enter a name for the register set and click *Finish*, as shown in the following image.

🔬 New Register Set	Wizard	_ 🗆 ×
General Propert	ies	
Please select a projec	t location and choose a name for the new Register Set	
Project Folder	/EDIHL7_usr_sample_proj/Registers	Browse
Name	XMLEDIHL7	
Description		
Target Server Version	7.0.4 Install additional Target Server Version	•
	Create in current folder	
?	Finish	Cancel

The new register set appears under Registers in the Registers folder, as shown in the following image.

🔏 Inte 🕄 🔍 Wa 🛋 Libra 💆 🗆	💩 localhost	S XMLEDIHL7_08_QS
(> for   = 😫 🏹		
E EDIHL7_usr_sample_proj		
🗁 Adapters	Name	
- 🗁 Applications		
🖻 🗁 Channels		
Image: State St		
- 🗁 Ebixes		
🗄 🗁 Flows		
Image: State St		
E- 🗁 Registers		
XMLEDIHL7		
- 🧀 Schemas		
🧀 Transforms		
🧀 XML		

3. Click the *Add a property* icon to add a register to the register set.

🕯 Inte 🛛 🤨 iWa 🛋 Libra 🖓 🗆	Violation State International	pFlow_MLLP_Channel	7.iwr 🕄		
는 🔶 😧 😑 😵 🏹					
EDIHL7_usr_sample_proj					
- 🗁 Adapters	Name	Туре	Value	Description	
- 🗁 Applications					
🖻 🗁 Channels					
E = XMLEDIHL7_OB_QS_pFlow_MLLP_					
- 🗁 Ebixes					
🕀 🗁 Flows					
I XMLToEDIHL7_Ebix_2_MLLP					
E 🗁 Registers					
XMLEDIHL7					
- 🗁 Schemas					
- 🧀 Transforms					
- 🧀 XML					

4. Enter a name for the new register in the Name field, select a register type from the Type drop-down list (set to string by default), and then enter a value in the Value field, as shown in the following image.

<u>a</u> P	legiste	r Wizard	×
Ne	w Reg ter detai	<b>gister</b> ils regarding the new register.	
	Name:	Adk	
	Type:	string	•
	Value:	sreg(EDIHL7_Input)\OB_Output	•
3			×.
Desc	ription:		
			<u>×</u>
(	D		OK Cancel

- 5. Click OK.
- 6. Create the following registers under the XMLEDIHL7 registers set you just created, along with the values shown in the table below:

Register Name	Value
Ack	Name=Ack, Value=sreg(EDIHL7_INPUT)\OB_OUTPUT
Input	Name=Input, Value=sreg(EDIHL7_INPUT)
Error	Name=Error, Value=sreg(EDIHL7_INPUT)\IB_Error
GoodOutput	Name=Output, Value=sreg(EDIHL7_INPUT)\IB_TransformGood
BadOutput	Name=GoodOutput, Value=sreg(EDIHL7_INPUT)\IB_Error
ValidRpt	Name=ValidRpt, Value=sreg(EDIHL7_INPUT)\IB_Report

Register Name	Value
Archive	Name=Archive, Value=sreg(EDIHL7_INPUT)\IB_Archive

🖌 Integrati 23 😈 NWay Ex 🚔 Library M	S FDIHL1_IR_G2_bHow_b	ALLP_Channel Vo localhost	EDIHL7.WY 23		
(> 🔬   🚍 🤽 🗸					A C /* X A A A
EDIHL7_usr_sample_proj		1			
- 🗁 Adapters	Name	Туре	Value	Description	
- C Applications	Ack	string	sreg(EDIHL7_INPUT)\OB_Output		
E Channels	Input	string	sreg(EDIHL7_INPUT)		
- 😂 Ebixes	Error	string	sreg(EDIHL7_INPUT)\IB_Error		
E 😂 Flows	GoodOutput	string	sreg(EDIHL7_INPUT)\IB_Transform		
EDIHL7toXML pflow MLLP AckAgent ValidRpt	BadOutput	string	sreg(EDIHL7_INPUT)\IB_Error		
E 🕞 Registers	ValidRpt	string	sreg(EDIHL7_INPUT)(IB_Report		
EDIHL7	Archive	string	sreg(EDIHL7_INPUT)\IB_Archive		
Schemas					
- Construction - Cons					
Ce XMI					

## *Procedure:* How to Add a Register Set to an Inbound Channel as a Dependency

1. Click the *channel: XMLEDIHL7\_IB\_QS\_pFlow\_MLLP\_Channel* node and then click the Add dependency icon on the far right side panel of the channel properties pane, as shown in the following image.

EDIHL7_IB_Q5_pFlow_MLLP_Channel 🛛 🍪 localhost 🔲 EDIHL7.kw		1
Channel Builder		
EDJHL7_IB_QS_pFlow_MLLP_Channel	DIHL7_IB_QS_pFlow_MLLP_Channel i (	2
	Dynamic Routing Manage channel dependencies, such as Ebixes and Registers:	
	Type Location	1
	×	]

2. From the Registers folder under the integration folder, select XMLEDIHL7 and click OK.

A Resource Selection	
EDIHL7_usr_sample_proj	
?	Cancel

The iIT page should resemble the following image.

biocatiost State Stat				
Channel Builder				
XMLEDIHL7_08_QS_pFlow_MLLP_Channel		XMLEDIHL7_OB_QS_pFlow_MLLP_Channel i		
Channet: JMEDIAT, CB_QS_prion, MLP_Charnel      Set State: JMEDIAT, CB_QS_prion, MLP_Init      Set State: JMEDIAT, CB_QS_prion, MLP_Init      Set State: JMEDIAT, CB_QS_prion, MLP_Route (default)      Set State: JMEDIAT, CB_QS_prion, MLP_Anter      Set State: JMEDIAT, CB_QS_prion, MLP_Initian (Passifrought)      Set State: JMEDIAT, CB_QS_prion, MLP_Emitter (Passifrought)	<ul><li>↓</li><li>↓</li><li>↓</li></ul>	Tick the check-box below to enable or disable dynamic routing for this channel.		

3. Click the Save icon to save your changes. You can also use the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

🗏 EDIHL7_IB_QS_pFlow_MLLP_Channel 🔯 🔞 localhost 🔲 EDIHL7.iwr					
Channel Builder					
EDIHL7_IB_QS_pFlow_MLLP_Channel		DIHL7_IB_Q5_pFlow_MLLP_Chan	nel		
Channel: EDIM17_IB_QS_pFlow_MLIP_Channel     Set inlet: EDIM17_IB_QS_pFlow_MLIP_Inlet     Set instener: EDIM17_IB_QS_pFlow_MLIP_Listener (MLL     Deparser: EDIM17_IB_QS_pFlow_MLIP_Preparser	*	Tick the check-box below to enable or disable dynamic routing for this channel. 「」Dynamic Routing Manage channel dependencies, such as Ebixes and Registers:			
Control Contro Control Control Control Control Control Control Control Control C	Ŷ	Type register	Location /EDIHL7_usr_sample_proj/Registers/EDIHL7		

## Importing an Ebix Into the Workspace

This section describes how to import an Ebix into the workspace using iWay Integration Tools (iIT).

#### Procedure: How to Import an Ebix Into the Workspace

1. In the integration project EDIHL7\_usr\_sample\_proj, right-click the Ebixes folder and then select *Import* from the context menu, as shown in the following image.



2. Expand the iWay Integration folder, select *Ebix*, and then click *Next*, as shown in the following image.



3. Click the ellipses (...) button to browse and import the ebix from a specific folder location, as shown in the following image.

🚮 Import		
General Propert  Other name field is re	ies Page equired.	
Project Folder	/EDIHL7_usr_sample_proj/Ebixes	Browse
Import	[	
Name		
Description		
Target Server Version	7.0.4	-
	Install additional Target Server Version	
?	< Back Next > Einish	Cancel

4. Select *HL7\_2.6 ebix* from the folder location and click *Open*, as shown in the following image.



- 5. In the Import wizard, click Next.
- 6. Expand the hI7\_2.6 ebix from the left panel and select the 2.6 folder.

7. Select the messages you wish from the right side panel and click *Finish*, as shown in the following image.

🦼 Import		
Ebix Entries		
Select ehiv version to	view available chiv entries	
Select ebix version to	view available ebix entries.	
Transform ebix are sport represents transform Select ebix version to Im Import as System	ecially designed archive files that contain ebix entries. El configuration and dependencies used by iWay transform view available ebix entries. Ebix	bix entry engine.
bix	Ebix Entries	
☐ HI7_2.6 2.6		
Description: Entry: ADT_A05 Run Time Mode: N/A Description:		
•	< Back Next > Einish	Cancel

/ Integration - EDIHL7\_usr\_sample\_proj/Ebixes/HL7/hl7\_2.6/ebix/1.0/2.6/dictionaries/HL7\_ADT\_A01\_2.6.dic - iWay Integration Tools File Edit Navigate Search Project Run Window Help s | 🚱 🕅 📲 | 🛝 | 🎄 • 🔘 • 🗛 • | 🚀 • | 🧐 • 🗄 • 😓 🖕 • 🗉 • 🔏 Inte... 🙁 😈 iWa... 🚔 Libra... 🖓 🗖 📑 EDIHL7\_IB\_QS\_pFlow\_MLLP\_Channel 🛛 🚳 localhost EDIHL7.iwr H7\_2.2.6.ADT\_A01 🕅 EDI FHS EDIHL7\_usr\_sample\_proj 01 [File Field Separator]
 02 [File Encoding Characters] Adapters Applications E C00069\_03 [File Sending Application] E Channels C00070\_04 [File Sending Application]
 C00071\_05 [File Receiving Application] EDIHL7\_IB\_Q5\_pFlow\_MLLP\_Chanr Ebixes E 🙆 C00072\_06 [File Receiving Facility] E - B Flows 07 [File Creation Date Time] EDIHL7toXML\_pflow\_MLLP\_AckAge 08 [File Security] E C Registers 10 [File Header Comment] 🗁 Schemas 11 [File Control ID] 🗁 Transforms 12 [Reference File Control ID]
 20269\_13 [File Sending Network Address]
 C02270\_14 [File Receiving Network Address] B XML E S Group\_Loop B BHS 🗄 🖏 BTS E S FTS 01 [File Batch Count]
 02 [File Trailer Comment] - 0 E Outline 23 Header:HL7Header Structure:HL7\_ADT\_A01\_2.6 Schema:HL7\_ADT\_A01\_2.6 Rule:HL7\_ADT\_A01\_2.6\_Rules HL7\_ADT\_A01\_2.6toXML

The iIT page should resemble the following image.

#### *Procedure:* How to Add an Ebix to an Inbound Channel as a Dependency

1. Click the *channel: XMLEDIHL7\_IB\_QS\_pFlow\_MLLP\_Channel* node and then click the *Add dependency* icon on the right side panel, as shown in the following image.



2. Expand *EDIHL7\_usr\_sample\_proj*, *Ebixes*, *HL7*, and then select ebix *hl7\_2.6* and click *OK*, as shown in the following image.

Resource Selection		
EDIHL7_usr_sample_pro	23	
?	ОК	Cancel

3. Click the Save button to save your changes or press the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

EDIHL7_IB_Q5_pFlow_MLLP_Channel 🛛 🔞 localhost			- 0
Channel Builder			II II
EDIHL7_IB_Q5_pFlow_MLLP_Channel	DIHL7_IB_Q5_pFlow	_MLLP_Channel	i 🖉
Channel: EDIM 7_B, QS_DFlow, MLP_Channel State: EDIM 7_B, QS_DFlow, MLP_Listener (ML State: EDIM 7_B, QS_DFlow, MLP_Listener (ML State: EDIM 7_B, QS_DFlow, MLP_Preparer State: EDIM 7_B, QS_DFlow, MLP_Preparer	Tick the check-box belo Dynamic Routing Manage channel dependence	w to enable or disable dynamic routing for this channel.	
route: EDIHL7_IB_Q5_pFlow_MLLP_Route (default)	Туре	Location	151
So outlat: 50/5/ 7 /8 OS office Mile Octat	register	/EDIHL7_usr_sample_proj/Registers/EDIHL7	
emitter: EDIHL7_IB_Q5_pFlow_MLLP_Emitter (Pass	ebix	/EDIHL7_usr_sample_proj/Ebixes/hl7_2.6	36
			÷
			÷.

## Configuring an iWay Integration Application for Inbound Processing

This section describes how to configure an iWay Integration Application (iIA) for inbound processing using iWay Integration Tools (iIT).

## *Procedure:* How to Configure an iWay Integration Application for Inbound Processing

1. Right-click the Applications folder under the EDIHL7\_usr\_sample\_proj integration project, select *New*, and then click *Application* from the context menu, as shown in the following image.

🖌 Inte 🛛 😈	IWa ■ Lbra □ □	EDIHL7_IB_QS_pFlow_MLLP_Chan	nel 🔀 🔞 localhost
	sample_proj s Go Into Open in New Window Copy Paste Duplicate X Delete Move Rename	EDIHL7_IB_QS_pFlow_MLLP_Cl A Integration Project Project Application Channel Process Flow Find Transform Register Set Schema Set	hannel w_MLLP_Channel w_MLLP_Inlet 75_pFlow_MLLP_Listener (ML 7_QS_pFlow_MLLP_Preparser ow_MLLP_Route (default) L_pflow_MLLP_AckAgent_Val iow_MLLP_Outlet 75_pFlow_MLLP_Emitter (Pas
	i≥s Import i≤i Export ≷ Refresh	f(J) IFL Expression       C Example       C Other       C Other	
Outline      An outline is not ava	Validate Run As Debug As Team Compare With Restore from Local History Source Properties	* * *	

2. Enter a value in Name field (for example, EDIHL7\_usr\_sample\_App), and click *Next*, as shown in the following image.

Application Wizard		_ 🗆 ×
General Properties		
Please select a project location	n and choose a name for the new application	
Project Folder	/EDIHL7_usr_sample_proj/Applications	Browse
Name	EDIHL7_usr_sample_App	
Description		Â
	1	<b>_</b>
Target Server Version	7.0.4	7
	Install additional Target Server Version	
	- occharch	
2	<back next=""> Finish</back>	Cancel
Ū		

3. Select the *XMLEDIHL7\_IB\_QS\_pFlow\_MLLP\_Channel* check box from Resource Selection pane and keep clicking *Next* until you get to the Bindings pane, as shown in the following image.

	1700	HUL	Location	Descri	Select All
EDIHL7_IB_QS_pFlow_MLL EDIHL7toXML_pflow_MLLP_	proc	yes	/EDIHL7_usr		Deselect All

4. In the Bindings pane, add Bindings for Registers you have added in the XMLEDIHL7 Register Set to the iIA EDIHL7\_usr\_sample\_App and click *Finish*, as shown in the following image.

🔏 New Ap	plication Wizar	d			_ 🗆 🗙
Bindings List the ap resolved a	s oplication bindings at deployment tim	, such as Special e.	Registers and Pro	viders, to be	
Name	Туре	Default Value	Description		-
	New Appli Enter details	cation Bindi regarding the ne	ng w application bindi	ng	
	Name: Type: Default Value:	Ack Register sreg(EDIHL7_I)	VPUT)\O8_Output		I
•	Description:			OK	≍ ≤
?		< Back	Next >	Finish	Cancel

After adding all register bindings, your ilT screen should resemble the following image.

d	1*
d	1*
d	Co Co
d	S
	-
	×
5. To build the application, right-click EDIHL7\_usr\_sample\_App, select *Integration Tools*, and then click *Build* from the context menu, as shown in the following image.

iWa 🖹 Libra 🗖 🗖	EDIHL7_IB_QS_pFlow_MLLP_Channel	ocalhost 🔞	SEDIHL7_us	r_sample_App.iab 🛛	
← → ☆   🖻 🍇 ▽	Add channels, transforms and processes	to your application.	Autostart column l	lets you control channel start behavior on serv	er startup.
_sample_proj	Name	Туре	Auto Start	Location	Description
s	EDIHL7 IB QS pFlow MLLP Channel	inlineChannel	ves	/EDIHL7 usr sample proj/Channels/	
ions			1	1	
1L7_usr_sample_App					
Components					
bindings					
Duild.xmi					
Danicz_usr_sample_App.ilä					
7 ID OS oFlow MUD Chapr					
7_to_Q3_priow_MLLP_Chanr					
INC/COMPL_DROW_PILLP_ACK					
24					
_6.0					
7toYML offour MLD AckAger					
C CONTRESTON STORE ACTAGE					
7					
ms					
				100	
A Deserve diam iPhre	The Way and have with the balance	e Helidbert		una candu l la	
24 FLOCESS IIOM , FDIE	spreaker bilos upph ackades	~_vallakbt.	compiled St	accessrarry	

- 6. To publish the iIA, right-click *EDIHL7\_usr\_sample\_App*, select *Integration Tools*, and then click *Publish to* from the context menu.
- 7. Provide the iSM server details in the Server URL text box and the other credentials, then click *Finish*.

e 🛛 😈 iWa 🛋 Libra 🖓 🗖	EDIHL7_IB_QS_pFlow_MLLP_Channel	ocalhost 🔞	G EDIHL7_us	r_sample_App.iab 🕅	
EDIHL7 usr sample proj	Add channels, transforms and processes	to your application.	Autostart column I	lets you control channel start behavior on serv	ver startup.
Adapters	Name	Туре	Auto Start	Location	Description
C Applications	EDIHL7_IB_QS_pFlow_MLLP_Channel	inlineChannel	yes	/EDIHL7_usr_sample_proj/Channels/	
EDIHL7_usr_sample_App					
E Components					
🗄 🊾 Bindings					
-if build.xml					
EDIHL7 usr sample App.ia					
Channels					
E EDIHL7 IB OS pFlow MLLP Chang					
EDIHL ZtoXML of ow MLLP Ack					
EDIHL7					
H					
- Fbixes					
C Flows					
F TO EDIHL7toXML pflow MLP AckAge					
Registers					
EDIH 7					
Chamar					
- Are					
<b>&gt;</b>					
isole 🔀					
ana Concela					
age Console				0.0000000000000000000000000000000000000	

The iIT page should resemble the following image.

- 8. Deploy the iIA application on the iSM registry.
- 9. Have the following folder structure created before starting your application in the iSM console, as shown in the following image.



# Setting HL7 System Registers Using MLLP

This section describes how to set system registers using the iWay Service Manager (iSM) Administration Console.

#### Procedure: How to Set System Registers

- 1. Open the iSM Administration Console and select *EDIHL7\_usr\_sample\_App\_MLLP\_IB* [down] from the Management drop-down list.
- 2. From the console bar, click Server, Register Settings, and then click Add, as shown in the following image.

Properties General Properties Java Properties	Register Settings Special registers are name available to all component Listed below are the regist Special Registers	ed variables that reference values which s of the system. Any changes to the regi er settings for the HL7_usr_samples_App	are carried throughout the system. Once defined, ster settings do not take effect until the server is n configuration of this server.	these variables estarted/redepl
Settings	Name	Value	Description	Туре
Console Settings	iwayversion	unavailable	system defined (readonly)	string
Java Settings	iwayhome	unavailable	system defined (readonly)	string
Register Settings	iwaydata	unavailable	system defined (readonly)	string

3. For the deployed EDIHL7\_usr\_sample\_App\_MLLB\_IB application, define the system registers as listed in the following table:

Register Name	Value
EDIHL7_INPUT	<pre>sreg(EDIHL7_Installdir)\HL7_in</pre>
EDIHL7_installdir	C:\EDIHL7_Accelerator
EDIHL7_OUTPUT	<pre>sreg(EDIHL7_Installdir)\HL7_out</pre>

Register Name	Value
ValidateEDIHL7	true

#### **Register Settings**

Special Register De	finition
Name *	Enter the name of the special register to add.
	EDIHL7_Installdir
Туре	Select a type for the value of this special register.
	string
Value *	Enter a value for this special register. The value can be a constant or a call to the evaluation functions.
	C:\EDIHL7_Accelerator
Description	Enter a description for this special register.

After adding the registers, your Register Settings page in the iSM Administration Console should resemble the following image.

iway.serverfullhost	iwhl7.ibi.com	system defined (readonly)	string
iway.pid	1668	system defined (readonly)	string
EDIHL7_INPUT	sreg(EDIHL7_Installdir)\HL7_in		string
EDIHL7_Installdir	C:\EDIHL7_Accelerator		string
EDIHL7_OUTPUT	sreg(EDIHL7_Installdir)\HL7_out		string
ValidateEDIHL7	true		string

4. Start the deployed application in the iSM Administration Console and ensure that the channel is up and running in the Monitoring section of the console.

## Testing the Inbound Channel Application Using MLLP

This section describes how to test the inbound channel application using the Minimal Lower Layer Protocol (MLLP).

# Procedure: How to Test the Inbound Channel Application Using MLLP

1. Use one of the tools to send messages to a specified port (for example, 7Edit, Telnet Client), set the sending port to 778, and then send the HL7 message to that port, as shown in the following image.

Z 7Edit Professional - C:\EDIHL7_Accelerator\HL7_in\IB_Archive\hl	7_adta01c01_001_5ample_NI7	
File Edit Document Message XML Search Tools Window Hep		_
	S 12 (2) S : K ( 1/1 ) N (2) : 100	
Message d7.4 ×	C:EDIH.7_Acc001_Sample.hl7 ×	4 1
B. AddAvit addication (ADT_ADT)           B. Marsay Ready (DSD)           B. Marsay Ready (DSD)           B. District Sequent (ST)           B. Distri	INDIC         VIECTADING	(900) 485- ACT           OHPANY         GE     A     Send
	Teme: 00:00:00   Nr: 0   A-[ 0   Rr] 0	
[ message ] [ Documenc ]	Same and the second sec	

2. Configure a receiver port (for example, 12001) to receive the ACK file through the MLLP emit, as shown in the following image.

3	Receiver				a +×
Pri	ofile: Listern_i	12001 (localhost:12001)	💌 🔳 Profiles   🔍	Messages	📒 Stop
	Started	From	Time		
8	02:16:13	127.0.0.1	10/11/2015 02:16:13	Sending acknowledge	×
			10/11/2015 02:16:13	M5H[^~\&   IFENG  20151011021613  ACK 00000007 P 2.6	
			10/11/2015 02:16:13	MSA[AA]00000007	
			10/11/2015 02:16:13	Waiting for message	
			10/11/2015 02:16:13	Error reading message> Software caused connection abort: recv failed	
			10/11/2015 02:16:13	Connection closed	*
			Time: 00:00:00 N:	0	
	)Validator   🤏	🕨 Watch   🦘 Search   🌭 Ser	der 😵 Receiver		

3. Observe the transformed XML output under the EDIHL7\_Accelerator\HL7\_in \IB\_TransformGood location, as shown in the following image.

:h 🔻	New folder			
-	Name ^	Date modified	Туре	Size
		10/11/2015 2:12 AM	XML File	20 KB
	2015-10-11T06_16_13_542Z003.xml	10/11/2015 2:16 AM	XML File	20 KB

4. Monitor the reports under the EDIHL7\_Accelerator\HL7\_in\IB\_report, as shown in the following image.

EDIHL7_Acc	celerator - HL7_in - IB_Report	<ul> <li>Search IB_Report</li> </ul>			
re with 🔻	New folder				)III 🕶 🛄
1	Name 1	Date modified	Туре	Size	
		10/11/2015 2:12 AM	XML File	23 KB	
		10/11/2015 2:16 AM	XML File	23 KB	

5. Observe the Acknowledgement in the EDIHL7\_Accelerator $HL7_inOB_output$  location, as shown in the following image.

EDIHL7_Accelerator + HL7_in + OB_Output					<ul> <li>Search OB_Output</li> </ul>
are with 🔻	New folder				)III 🔹 🛅
1	Name *	Date modified	Туре	Size	
	72015-10-11T06_12_01_334Z002.hl7	10/11/2015 2:12 AM	HL7 Files	1 KB	
	2015-10-11T06_16_13_542Z003.hl7	10/11/2015 2:16 AM	HL7 Files	1 KB	

- 6. If the input data contained any errors, you can review this error data in the output directory that you have configured for error handling (for example, EDIHL7\_Accelerator\HL7\_in \IB\_Error).
- 7. After inbound processing has completed, a copy of the input data that was used during the transformation is stored under the archive directory that you specified (for example, EDIHL7\_Accelerator\HL7\_in\IB\_Archive).

# **Outbound Processing: XML to HL7**

This section describes how to configure a basic outbound message flow for the iWay Integration Solution for EDIHL7. The message flow represents the movement and tasks in the conversion of a message from XML format to HL7 format.

#### In this chapter:

Chapter

- Configuring a Channel for HL7 Outbound Processing
- Configuring Register Sets and Registers
- Importing an Ebix Into the Workspace
- Configuring an iWay Integration Application for Outbound Processing
- Setting EDIHL7 Outbound System Registers
- Testing the Outbound HL7 Channel Application

## Configuring a Channel for HL7 Outbound Processing

The outbound channel creates an HL7 message from XML and a XML-formatted validation report. The documents are routed to designated folders based on the success or failure results of the transformation.

#### *Procedure:* How to Create a Channel for Outbound Processing

1. Start iWay Integration Tools (iIT).

2. Right-click the Integration Explorer pane, click *New*, and then select *Integration Project* from the context menu, as shown in the following image.



The New Integration Project dialog box opens, as shown in the following image.

A New Integration Project	
Integration Project	
Create a new Integration project.	1
Project name EDIHL7_usr_sample_proj	
Project location	
✓ Use default	
Directory C:\OFFSHORE\IT704PATCHHL7SEP172015\IT	-7.0.4\EDIHL7_usr_ Browse
Additional options	
Create integration folders	
Target Server Version 7.0.4	Install additional Target Server Version
•	Finish Cancel

3. Enter a name for the Integration Project (for example, *EDIHL7\_usr\_sample\_proj*) in the Project name field, and then click *Finish*.

A new Integration Project node called *EDIHL7\_usr\_sample\_proj* is added to the Integration Explorer.

4. Expand the *EDIHL7\_usr\_sample\_proj* Integration Project node, right-click the *Channels* folder, select *New*, and then click *Channel* from the context menu, as shown in the following image.



5. Enter a name for the Channel (for example, XMLEDIHL7\_OB\_QS\_pFlow\_Ebix\_Channel), and then click *Next*, as shown in the following image.

🦼 Channel Object		_ <b>D</b> ×
Channel Genera Please choose a nam	al Properties e and location for this new Channel.	1
Project Folder	/EDIHL7_usr_sample_proj/Channels	Browse
Name	XMLEDIHL7_OB_QS_pFlow_Ebix_Channel	
Description		K
	1	-
Target Server Version	7.0.4	-
	Install additional Target Server Version	
?	< Back Next > Finish	Cancel

The Inbound/Outbound Protocols pane appears.

6. Click the *Add* button to add a File Listener in the Inbound section, as shown in the following image.

A Channel Object					
Inbound/Outbour	nd Protocols				
Specify the inbound ar	nd outbound protocols to b	e used in the channel			
Inbound:					
Name	Listener Type	Description			
Have an inlet crea	ted for each inbound proto	col			
Outbound:			÷ 🔀		
Name	Emitter Type	Description			
- Norrice	Liniter Type	Description			
Have an outlet or	Line an outlat craated for each outbound protocol				
, have an outlet of					
	1		1		
C	< <u>B</u> ack	Next > Einish	Cancel		

7. From the Types Filter list, select *File* and then click *Finish*, as shown in the following image.

1		_ 🗆 🗙
Define listener type		
Select type of listener you want to create. Alterna box to filter types.	atevily, use search	
Types filter		
1		
Use '*' to narrow filter matching: a*, *b, a*c		
AQ AS1 AS2 AS2 [nonblocking] Backup Heartbeat Server ConnectDirect Console CS3 Email Envoy Exchange FTP[S] Client (Clear text or SSL FTP Clients) FTP[S] Client (Clear text or SSL FTP Clients) FTP[S] Server (Clear text or SSL FTP Server) HL7-MLLP-Listener		
Accepts documents from files in directories		
0	Finish	Cancel

8. Select the *Have an Inlet created for each inbound protocol* check box to create an Inlet for the channel, as shown in the following image.

d Channel Object				
Inbound/Outbou	und Protocols			
Specify the inbound	and outbound protocols to be	used in the channel		-
Inbound:				
Name	Listener Type	Description		
listener.1	File	Accepts documents	from files in	
Have an inlet cre	eated for each inbound protoc	lo		
When uncheck	ed, a single inlet will be create	d containing all the specifie	d listeners; otherwise, a	a separate inlet will be created for each listener.
Outbound:			* ×	
Name	Emitter Type	Description		
Have an outlet of	reated for each outbound pro	otocol		
•	< <u>B</u> ack	Next > Einish	Cancel	

9. Click the *Add* button in the Outbound section to define an emitter.

The Define emitter type pane appears.

10. In the Types filter list, select *Passthrough* to define the passthrough emitter, and then click *Finish*, as shown in the following image.

Define emitter type	
Select type of emitter you want to create. Alternatevily, use search box to filter types.	
Types filter	
Use '" to narrow filter matching: a", "b, a"c	
FTP[S] Client (Deprecated FTP Clients)	-
HTTP 1.0 [deprecated]	
Internal Queue	
Java Message Service (jmsq)	
MLLP	
MQ MOIMS	
Ordered Queue	
Passthrough	
SETP Client (Secure Shell version ETP Client)	
Sonic	
TCP	
TIDEV	-
Discards messages	
olacal da measages	
	1
Enish C	ancel

**Note:** Passthrough does not emit data from the listener. Instead, it just passes the control here and does nothing.

11. Select the *Have an outlet created for each outbound protocol* check box to create an outlet for the channel, and then click *Finish*, as shown in the following image.

1	Channel Object			- O ×	
Ir	Inbound/Outbound Protocols				
Specify the inbound and outbound protocols to be used in the channel					
	Inbound:			÷ ×	
Г	Name	Listener Type	Description		
	listener.1	File	Accepts documents from	n files in	
- [	Have an inlet created for	or each inbound protocol			
	Outbound:				
	Name	Emitter Type	Description		
	emitter.1	Passthrough	Discards messages		
-					
	I ave an outlet created	for each outbound protocol			
-					
	2	< Back No	Vt > Einich	Capital	
		- Dack		Cancer	

The Channel Builder pane appears.

🖌 Inte 🐰 😈 Wa 🛋 Libra 🖓 🗆	SXMLEDIHL7_08_QS_pFlow_Ebix_Channel 23	
Inte ⊠ Wa ■ Libra □     Adapters     Adapters     Channels     Chann	XMLEDHL7_08_QS_pFlow_Ebix_Channel      Channel Builder <u>3 errors detected</u> XMLEDIHL7_08_QS_pFlow_Ebix_Channel      Get Channel: MEEDIHL7_08_QS_pFlow_Ebix_Channel      Get Channel: MEEDIHL7_08_QS_pFlow_Ebix_Channel	<ul> <li>♣</li> <li>₩</li> <li>☆</li> <li>⇒</li> </ul>

12. Under *inlet: inlet.1*, click *listener: listener.1* (*File*) and then expand the *Main* (*Missing 2 required fields*) configuration parameter on the right pane, as shown in the following image.

XMLEDIHL7_OB_QS_pFlow_Ebix_Channel X		
Ochannel Builder 3 errors detected		::
XHLEDIHL7_08_QS_pFlow_Ebix_Channel	i 2 Accepts documents from files in drectories Type: File change type Filter (enter string to filter properties) Filter (enter string to filter properties) Filter (enter string to filter properties) Filter (Hissing 2 required fields)	<b>42</b>
	Input Path ①  Destination ①  Removal Destination ①	
	Suffix In Filter (1) xml Scan subdirectories (1)	
	false 💌	
	false 💌	

13. Enter a valid Input Path, Destination, and Removal Destination (optional) path and then select *hI7* in the Suffix Out drop-down list, as shown in the following image.

*XMLEDIHL7_08_QS_pFlow_Ebix_Channel X			L
Channel Builder Process: process. 1: Process 'process. 1' not defined			::
XMLEDIHL7_06_Q5_pflow_Ebix_Channel  Channek IMLEDIHL7_08_Q5_pflow_Ebix_Channel  Channek IMLEDIHL7_08_Q5_pflow_Ebix_Channel  Channek Internet Intern	* * *	Filter (enter string to filter properties)  Filter (enter string to filter properties)  Filter (enter string to filter properties)  Filter Input Path ()  Filter Input Path ()	

14. Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.



15. Import the *XMLToEDIHL7\_Ebix\_2\_* Process Flow from the local drive or create the one in the EDIHL7\_usr\_sample\_proj directory in the Flows folder.

16. Under the *route: route.1(default)* node, select *process: process.1* and click the process icon on the right panel to reference the process flow into your channel, as shown in the following image.

SMLEDIHL7_OB_QS_pFlow_Ebix_Channel 23 💊 localhost			
Channel Builder Process: process. 1: Process 'process. 1' not defined			•
XMLEDIHL7_OB_QS_pFlow_Ebix_Channel		process	i <table-cell></table-cell>
□     channet: VMEDIPL Z_ CB_QS_pFlow_Ebbc_Channel       □     4% instement latener.1 (File)       □     0% instement latener.1 (File)       □     1% emitter: emitter.1 (Passtbrough)	* * *	seect process from workspace that you want to be referenced by this channel component	

17. Select a process flow from the integration project and then click OK.

A Resource Selection	_ 🗆 🗙
<pre>EDIHL7_usr_sample_proj Environ En</pre>	
Э ок с	Cancel

**Note:** Process flows should already be built and available in the iIT integration project. They can be found in the EDIHL7\_usr\_sample\_proj directory inside the Flows folder.

For more information, see *Process Flow Used for Outbound Processing: XML to HL7* on page 164.

Your screen should resemble the following image.

*XMLEDIHL7_OB_QS_pFlow_Ebix_Channel X localhost		- 0
Channel Builder Process: process. 1: Process 'process. 1' not defined		
XMLEDIHL7_OB_QS_pFlow_Ebix_Channel	process	i 🙆
Compared American Compared American Compared     Compared American Compared Ame	Select process from workspace that you want to be referenced by this channel component	
	/EDIHL7_usr_sample_proj/Flows//MLToEDIHL7_Ebix_2.wp/AMLToEDIHL7_Ebix_2.wp	

 Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.

SXMLEDIHL7_OB_QS_pFlow_Ebix_Channel 🛛 🗞 localhost	
Channel Builder	
XMLEDIHL7_08_QS_pFlow_Ebix_Channel	process $i$ Select process from workspace that you want to be referenced by this channel component
B     channet: WAEDUR 7_OB_QS_OFlow_Ebic_Channel       B     channet: MetDUR 7_OB_QS_OFlow_Ebic_Channel       B     channet: Methers. I (File)       C     Sistemer. I. kiterer. I (File)	EDIH.7_usr_sample_proj/Flows/MMLToEDIH.7_Ebix_2.iwp/MMLToEDIH.7_Ebix_2.iwp

### Reference: Process Flow Used for Outbound Processing: XML to HL7

This section provides an overview of the process flow used for outbound processing: XML to HL7. This process flow (XMLToEDIHL7\_Ebix\_2) is already built and available in the iIT integration project. It is located under the *EDIHL7\_usr\_sample\_proj* node inside the Flows subfolder.



The following image shows the entire outbound process flow, including all of the nodes that are used and their connections.

In this process flow, an HL7 formatted document is read from a validation report file. The XML tags are stripped and the document is written to a directory. Only valid HL7 files are emitted. Error files as well as their input and any error messages can be found in the validation report file.

### Procedure: How to Update Basic Details for Channel Components

1. Select the channel component *inlet:inlet.1* from the channel and click the update channel component icon on the top right side panel, as shown in the following image.



2. Rename or update the inlet details (for example, XMLEDIHL7\_OB\_QS\_pFlow\_Ebix\_Inlet), and click *OK*, as shown in the following image.



After renaming or changing the channel components, your iIT Channel Builder should resemble the following image.

SXMLEDIHL7_OB_QS_pFlow_Ebix_Channel X 🔞 localhost			
Channel Builder			
XMLEDIHL7_0B_QS_pFlow_Ebix_Channel	, ,	XMLEDIHL7_OB_Q5	_pFlow_Ebix_Channel
□       □	* * *	Tick the check-box be Dynamic Routing Manage channel depe	low to enable or disable dynamic routing endencies, such as Ebixes and Registers: Location

# **Configuring Register Sets and Registers**

This section describes how to configure register sets and registers using iWay Integration Tools (iIT).

### Procedure: How to Configure Register Sets and Registers

1. In the EDIHL7\_usr\_sample\_proj project tree, right-click the Registers folder, and select *New*, and then click *Register Set* from the context menu, as shown in the following image.



2. In the Name field, enter a name for the register set and click *Finish*, as shown in the following image.

🔬 New Register Set	Wizard	_ 🗆 ×
General Propert	ies	
Please select a projec	t location and choose a name for the new Registe	r Set
Project Folder	/EDIHL7_usr_sample_proj/Registers	Browse
Name	XMLEDIHL7	
Description		<u></u>
		Ţ
		<u></u>
Target Server Version	7.0.4	•
	Install additional Target Server Version	
	Create in current folder	
?	F	Finish Cancel

The new register set appears under Registers in the Registers folder, as shown in the following image.



3. Click the Add a property icon to add a register to the register set.

J Integration - EDIHL7 usr sample_proj/Registers/XMLEDIHL7.iwr/XMLEDIHL7.iwr - iWay Integration Tools				
File Edit Navigate Search Project Run Wir	File Edit Navigate Search Project Run Window Heb			
🗈 • 🗄 🗠   🗞 🖄 🕍   🛤	\$\$ • <b>○</b> • <b>♀</b> • ] ∦ • ] ≿ •	$\forall \cdot \leftarrow \diamond \cdot \diamond \cdot$		
🔬 Inte 🔀 🦉 iWa 🔿 Libra 🖓 🗖	SMLEDIHL7_OB_QS_pFlow_Ebix_Chan	nel 🚳 localhost 🛅 XMLEDIHL7	liwr 83	
+ + < < < & <				🔶 🖾
E EDINL/_USr_sample_proj	Name	Type	Value	Description
Adapters				
E 🗁 Channels				
🗁 Ebixes				
E Elows				
Registers				
XMLEDIHL7				
Schemas				
Transforms				
- ~~				

4. Enter a name for the new register in the Name field, select a register type from the Type drop-down list (set to string by default), and then enter a value in the Value field, as shown in the following image.

🦼 Register	r Wizard	X
New Reg	jister	
Enter detai	ls regarding the new register.	
Name:	Archive	
Type:	string	
Value:	sreg(EDIHL7_OUTPUT)\OB_Archive	
Description:		
		▼.
?	]	OK Cancel
$\mathbf{\overline{\mathbf{C}}}$	L	

- 5. Click OK.
- 6. Create the following registers under the XMLEDIHL7 registers set you just created, along with the values shown in the table below.

Register Name	Value
Input	Name=Input, Value=sreg(EDIHL7_OUTPUT)
Output	Name=GoodOutput, Value=sreg(EDIHL7_OUTPUT) \OB_TransformGood
Archive	Name=Archive, Value=sreg(EDIHL7_OUTPUT)\OB_Archive
ValidationReport	Name=ValidationReport, Value=sreg(EDIHL7_OUTPUT) \OB_Report
Error	Name=Error, Value=sreg(EDIHL7_OUTPUT)\OB_Error

Register Name	Value
GoodOutput	Name=Output, Value=sreg(EDIHL7_OUTPUT) \OB_TransformGood

XMLEDIHL7_08_QS_pFlow_Ebix_Channel	el 🚳 localhost 🛅 XMLEDIHL	Liwr 🖾 🙀 XMLToEDIHL7_Ebix_2	
			▲□/
Name	Туре	Value	Description
Input	string	sreg(EDIHL7_OUTPUT)	HL7 outbound flow scans this directory for XML files
Output	string	sreg(EDIHL7_OUTPUT)\OB_TransformGood	HL7 outbound flow writes HL7 to this directory
Archive	string	sreg(EDIHL7_OUTPUT)\OB_Archive	Archive of transformed XML files
ValdationReport	string	Sreg(EDIHL7_OUTPUT)\OB_Report	All validation reports are written here (success and fail)
Error	string	sreg(EDIHL7_OUTPUT)\OB_Error	Errors (bad formed xml) are written here
GoodOutput	string	sreg(EDIHL7_Output)\OB_TransformGood	
	Name Input Output Archive YalataroReport Error GoodOutput	WALEDIL7_06_QS_pflow_Ebix_Channel     Image: Channel       Name     Type       Input     sting       Output     sting       Archive     sting       Brow     sting       GoodOutput     sting	MALEDIA 7,00,05,pFlow_Box_Channel         Image: Control of the second seco

### *Procedure:* How to Add a Register Set to an Outbound Channel as a Dependency

1. Click the *channel: XMLEDIHL7\_OB\_QS\_pFlow\_Ebix\_Channel* node and then click the Add dependency icon on the far right side panel of the channel properties pane, as shown in the following image.

SMLEDIHL7_08_QS_pFlow_Ebix_Channel			- [
Channel Builder			
XMLEDIHL7_08_Q5_pFlow_Ebix_Channel	XMLEDIHL7_OB_Q5_	pFlow_Ebix_Channel	i 🖉
Channek IMEDINE 7.08 05 pFlow Ebx: Channel      Set States in the Internet of the Interne	Tick the check-box belo Dynamic Routing Manage channel deper	w to enable or disable dynamic routing for this channel.	
	Туре	Location	÷
			<b>∂</b>

2. From the Registers folder under the integration folder, select *XMLEDIHL7* and click *OK*.

A Resource Selection	
EDIHL7_usr_sample_proj	
? ок	Cancel

The iIT page should resemble the following image.

*XMLEDIHL7_O8_QS_pFlow_Ebix_Channel 🛛	
Channel Builder	
XHLEDIHL7_0B_QS_pFlow_Ebix_channel       B:=     channel: XMLEDIHL7_0B_QS_pFlow_Ebix_channel       B:=     channel: XMLEDIHL7_0B_QS_pFlow_Ebix_channel	XMLEDIHL7_OB_Q5_pFlow_Ebbx_Channel Tick the check-box below to enable or disable dynamic routing for this channel.
Solution: State of the stat	✓ Dynamic Routing Manage channel dependencies, such as Ebixes and Registers:
So outlet: XMLEDIHL7 OB OS pFlow Ebix Outlet	Type Location
emitter: JOALEDIN: 7_08_QS_pFlow_Ebix_Emitter (Pai	register /EDIHL7_usr_sample_proj/Registers//MLEDIHL7

3. Click the Save icon to save your changes. You can also use the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

MILEDIHL7_OB_QS_pFlow_Ebix_Channel			
Channel Builder			
XMLEDIHL7_08_QS_pFlow_Ebix_Channel	* * *	XMLEDIHL7_0B_QS_pFlow_Ebix_Channel	
		Tick the check-box below to enable or disable dynamic routing for this channel. ✓ Dynamic Routing Manage channel dependencies, such as Ebixes and Registers:	
		Туре	Location
		register	/EDIHL7_usr_sample_proj/Registers/XMLEDIHL7

# Importing an Ebix Into the Workspace

This section describes how to import an Ebix into the workspace using iWay Integration Tools (iIT).

### Procedure: How to Import an Ebix Into the Workspace

1. In the integration project EDIHL7\_usr\_sample\_proj, right-click the Ebixes folder and then select *Import* from the context menu, as shown in the following image.



2. Expand the iWay Integration folder, select *Ebix*, and then click *Next*, as shown in the following image.



3. Click the ellipses (...) button to browse and import the ebix from a specific folder location, as shown in the following image.

A Import				
General Proper (2) The name field is a	ties Page required.			
Project Folder	/EDIHL7_usr_sample_proj/Ebixes	Browse		
Import				
Name				
Description				
Target Server Version	7.0.4	-		
Install additional Target Server Version				
?	<back next=""> Enish</back>	Cancel		

4. Select *HL7\_2.6 ebix* from the folder location and click *Open*, as shown in the following image.

🦼 Open			×
() - 05	S (C:)   OFFSHORE   Ebix_Builds   H	L7 🗸 🚺 Search HL7	2
Organize 🔻 New f	older		8II - 🔟 🔞
🚖 Favorites	▲ Name ^	Date modified	Туре
🧮 Desktop	HL7_2.6.ebx	3/25/2015 5:22 PM	EBX File
Downloads			
Recent Places			
Nesktop			
🕞 Libraries			
Documents			
Music			
Subversion	-		
Videos			
🔒 Administrator			
Computer			
Network	<b>v</b> 4		
type control Parter			
	File name: [HL7_2.6.ebx	Ebx File (*.eb	×)
		<u>O</u> pen	Cancel
			- 11.

- 5. In the Import wizard, click Next.
- 6. Expand the hI7\_2.6 ebix from the left panel and select the 2.6 folder.

7. Select the messages you wish from the right side panel and click *Finish*, as shown in the following image.

- Import		
Ebiy Entrine		
Colect obly version to view	u susible abiy astrias	
Select ebix version to view	vavailable ebix entries.	
Transform ebix are special represents transform cont Select ebix version to view	Ily designed archive files that contain ebix entries. E figuration and dependencies used by iWay transform available abix entries	bix entry n engine.
Import as System Ebia	x	
libix	Ebix Entries	
□ <b>H</b> 7_2.6	ACK	<u> </u>
2.6	ADR_A19	
	ADT_A02	
	ADT_A04	
	ADT ADS	
	ADT_A07	
	ADT_A08	
	ADT_A09	
	ADT A11	
1	ADT A12	-
Description:		
Entry: ADT_A05		<u>^</u>
Run Time Mode: N/A Description:		
		~
	and the second second	G
(T)	< Dack Wexc > Einish	Cancel

👩 Inte... 🔀 XMLEDIHL7\_OB\_QS\_pFlow\_Ebix\_Channel hl7\_2.2.6.ADT\_A01 🔀 🤨 iWa... 🛋 Libra... E- 🔶 EDI - THS EDIHL7\_usr\_sample\_proj 01 [File Field Separator] Adapters 02 [File Encoding Characters] Applications E @ C00069\_03 [File Sending Application] - Channels E @ C00070\_04 [File Sending Facility] Ebix\_C\_\_\_\_\_ E - C00071\_05 [File Receiving Application] 🖲 🗁 Ebixes E \_ O0072\_06 [File Receiving Facility] E Cover 07 [File Creation Date Time] E 2 XMLToEDIHL7\_Ebix\_2 08 [File Security] - C Registers Ø 09 [File Name ID] MLEDIHL7 10 [File Header Comment] Break Schemas 🚸 11 [File Control ID] Transforms 12 [Reference File Control ID] 🗁 XML E @ C02269\_13 [File Sending Network Address] E 2 C02270\_14 [File Receiving Network Address] - Sroup\_Loop 🗄 🛂 BHS 🍤 MSH 🗄 😽 BTS 🗄 😽 FTS 🚸 01 [File Batch Count] 02 [File Trailer Comment]

The iIT page should resemble the following image.

### Procedure: How to Add an Ebix to an Outbound Channel as a Dependency

1. Click the channel: XMLEDIHL7\_OB\_QS\_pFlow\_Ebix\_Channel node and then click the Add dependency icon on the right side panel, as shown in the following image.



2. Expand *EDIHL7\_usr\_sample\_proj*, *Ebixes*, *HL7*, and then select ebix *hI7\_2.6* and click *OK*, as shown in the following image.

A Resource Selection		
EDIHL7_usr_sample_pro	9 <b>1</b>	
•	ОК	Cancel

3. Click the Save button to save your changes or press the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

SXMLEDIHL7_O8_QS_pFlow_Ebix_Channel				
Channel Builder				::
XMLEDIHL7_0B_Q5_pFlow_Ebix_Channel		XMLEDIHL7_0B_Q5_pFlow_Ebio	x_Channel	i 🛛
□       channet: XMEDIA Z, OB, QS, priow. Ebx, Charnel         □       •1% instener: XMEDIA Z, OB, QS, priow. Ebx, Inlet         □       •1% instener: XMEDIA Z, OB, QS, priow. Ebx, State (Finite)         ○       ⊕ troute: XMEDIA Z, OB, QS, priow. Ebx, Route (default)         □       ∰ troute: XMEDIA Z, OB, QS, priow. Ebx, Route (default)         □       ∰ troute: XMEDIA Z, OB, QS, priow. Ebx, Couldet         □       ∰ troute: XMEDIA Z, OB, QS, priow. Ebx, Couldet         □       ∰ troute: XMEDIA Z, OB, QS, priow. Ebx, Couldet         □       ∰ troute: XMEDIA Z, OB, QS, priow. Ebx, Ebx: Emitter (Pax	<b>∲</b>	Tick the check-box below to enable or disable dynamic routing for this channel. V Dynamic Routing Manage channel dependencies, such as Ebixes and Registers:		
	Ŷ	Type register ebix	Location /EDIHL7_usr_sample_proj/Registers/XMLEDIHL7 /EDIHL7_usr_sample_proj/Ebixes/h17_2.6	*
				Ŷ
				Ŷ

# Configuring an iWay Integration Application for Outbound Processing

This section describes how to configure an iWay Integration Application (iIA) for outbound processing using iWay Integration Tools (iIT).
#### Procedure: How to Configure an iWay Integration Application for Outbound Processing

1. Right-click the Applications folder under the EDIHL7\_usr\_sample\_proj integration project, select *New*, and then click *Application* from the context menu, as shown in the following image.



2. Enter a value in Name field (for example, EDIHL7\_usr\_sample\_App), and click *Next*, as shown in the following image.

Application Wizard		_ 🗆 ×
General Properties		
Please select a project locati	on and choose a name for the new application	
Project Folder	/EDIHL7_usr_sample_proj/Applications	Browse
Name	EDIHL7_usr_sample_App	
Description		
		-
Target Server Version	7.0.4	<b>v</b>
	Install additional Target Server Version	
	Create in current folder	
	Use Maven	
2		Cancel
0	- Phish	Cancer

3. Select the XMLEDIHL7\_OB\_QS\_pFlow\_Ebix\_Channel check box from Resource Selection pane and keep clicking Next until you get to the Bindings pane, as shown in the following image.

A New Application Wizard					_ 🗆 ×
Resource Selection					
Add channels, transforms and proces	sses to yo	our applica	ation.		
	-		a contraction of the second		
Name	Туре	Aut	Location	Lesa	Select All
Provide the second seco	nin	yes	/EDIHL/ USF	VM T	Developed all
	proc		/EDIHL/_usr	XMLT	Deselect All
•	-	-		•	
(?)	< Back	Ne	xt> Eini	sh	Cancel
_					

4. In the Bindings pane, add Bindings for Registers you have added in the XMLEDIHL7 Register Set to the iIA EDIHL7\_usr\_sample\_App and click *Finish*, as shown in the following image.

🔏 New Applicatio	on Wizar	d		
Bindings				
List the application resolved at deploy	ment time	, such as Special R e.	egisters and Providers, to be	
Name	Type	Default Value	Description	
1				×
New App	lication	Binding		5
Enter detai	s regardir	ng the new applicat	tion binding	H
Name	: Input			
Туре	: Regist	er		-
Default Value	: sreg(E	DIHL7_OUTPUT)		
Description				<u>^</u>
Description	•			-
• ?			ОК	Cancel
?		< <u>B</u> ack	Next > Einish	Cancel

After adding all register bindings, your iIT screen should resemble the following image.

XMLEDIHL7_OB_QS_pFlow_Ebix_Channel					
Add application bindings, such as Special Registers and Providers, to be resolved at deployment time.					
Name	Туре	Default Value	Description		
Input	Register	sreg(EDIHL7_OUTPUT)			
Output	Register	sreg(EDIHL7_OUTPUT)\OB_Transfor			
GoodOutput	Register	sreg(EDIHL7_OUTPUT)\OB_Transfor			
Archive	Register	sreg(EDIHL7_OUTPUT)\OB_Archive			
Error	Register	sreg(EDIHL7_OUTPUT)\OB_Error			
ValidationReport	Register	sreg(EDIHL7_OUTPUT)\OB_Report			

5. To build the application, right-click *EDIHL7\_usr\_sample\_App*, select *Integration Tools*, and then click *Build* from the context menu, as shown in the following image.



- 6. To publish the iIA, right-click *EDIHL7\_usr\_sample\_App*, select *Integration Tools*, and then click *Publish to* from the context menu.
- 7. Provide the iSM server details in the Server URL text box and the other credentials, then click *Finish*.

A Integration - EDIHL7_usr_sample_proj/Ch	nannels/XMLEDIHL7_08_QS_pFlow_Ebix_Channel.iwichannel/XMLEDIHL7_08_QS_pFlow_Ebix_Channel.iwichannel - iWay Integration Tools
File Edit Navigate Search Project Run Win	vdow Help
] 📬 📲 🐘 🖄 🖄 🗍 🕋 🖄 🖠 🖉 🗍 🛲 ]	弊・Q・Q→   ※・  指・用・〒 Φ・+・
🔬 Inte 🛛 😈 iWa 🛋 Libra 🖓 🗆	SMLEDIHL7_08_QS_pFlow_Ebix_Channel SS
	Channel Builder
EDIHL7_usr_sample_proj	XMLEDIHL7 OB OS pFlow Ebix Channel
B Applications	
Other Jusz sample_App     Other Jusz sample_App     Other Jusz sample_App     Other Jusz sample_App.ia     Other Jusz samp.ia     Other Jusz samp.ia     Other Jusz sample_App.ia     Oth	■ channek XMEDIAL7_00_05_pFlow_Ebix_Charnel         ■ *\$ inlet: XMEDIAL7_00_05_pFlow_Ebix_Inlet         □ + \$\$ inlet: XMEDIAL7_00_05_pFlow_Ebix_Inletener (File)         □ + \$\$ route: XMEDIAL7_00_05_pFlow_Ebix_Lotted (default)         □ + \$\$ voide: XMEDIAL7_00_05_pFlow_Ebix_Couldet         □ + \$\$ voide: XMEDIAL7_00_05_pFlow_Ebix_Couldet         □ + \$\$ voide: XMEDIAL7_00_05_pFlow_Ebix_Couldet         □ + \$\$ voide: XMEDIAL7_00_05_pFlow_Ebix_Entitler (Passtbrough)         □         □         □
UT Marcage Cancele	
[INFO110:19:05 Application 'EDI	HL7 usr sample App' built successfully.
[INFO]10:27:37 Application 'EDI	HL7_usr_sample_App' successfully built and published to http://localhost:9000
P	

The iIT page should resemble the following image.

- 8. Deploy the iIA application on the iSM registry.
- 9. Have the following folder structure created before starting your application in the iSM console, as shown in the following image.



## Setting EDIHL7 Outbound System Registers

This section describes how to set system registers using the iWay Service Manager (iSM) Administration Console.

#### Procedure: How to Set System Registers

- 1. Open the iSM Administration Console and select *EDIHL7\_usr\_sample\_App\_OB* [down] from the Management drop-down list.
- 2. From the console bar, click Server, Register Settings, and then click Add, as shown in the following image.

Properties General Properties Java Properties	Register Settings Special registers are name available to all component Listed below are the regist Special Registers	ed variables that reference values which . s of the system. Any changes to the regi er settings for the HL7_usr_samples_App	are carried throughout the system. Once defined, i ster settings do not take effect until the server is re configuration of this server.	these variables estarted/redepl
Settings	Name	Value	Description	Туре
Console Settings	iwayversion	unavailable	system defined (readonly)	string
Java Settings	iwayhome	unavailable	system defined (readonly)	string
Register Settings	iwaydata	unavailable	system defined (readonly)	string

3. For the deployed EDIHL7\_usr\_sample\_App\_OB application, define the system registers as listed in the following table:

Register Name	Value
EDIHL7_INPUT	sreg(EDIHL7_Installdir)\HL7_in
EDIHL7_installdir	C:\EDIHL7_Accelerator
EDIHL7_OUTPUT	<pre>sreg(EDIHL7_Installdir)\HL7_out</pre>

Register Name	Value
ValidateEDIHL7	true

#### **Register Settings**

Special registers are available to all comp below are the registe	named variables that reference values which are carried throughout the system. Once defined, these variables become onents of the system. Any changes to the register settings do not take effect until the server is restarted/redeployed. Listed ar settings for the EDIHL7_usr_sample_App configuration of this server.
Special Register L	efinition
Name *	Enter the name of the special register to add.
	EDIHL7 Installdir

pe	Select a type for the value of this special register.           string
'alue *	Enter a value for this special register. The value can be a constant or a call to the evaluation functions. C:\EDIHL7_Accelerator
escription	Enter a description for this special register.

After adding the registers, your Register Settings page in the iSM Administration Console should resemble the following image.

iway.serverfullhost	iwhl7.ibi.com	system defined (readonly)	string
iway.pid	1668	system defined (readonly)	string
EDIHL7_INPUT	sreg(EDIHL7_Installdir)\HL7_in		string
EDIHL7_Installdir	C:\EDIHL7_Accelerator		string
EDIHL7_OUTPUT	sreg(EDIHL7_Installdir)\HL7_out		string
ValidateEDIHL7	true		string

4. Start the deployed application in the iSM Administration Console and ensure that the channel is up and running in the Monitoring section of the console.

# Testing the Outbound HL7 Channel Application

This section describes how to test the outbound channel application.

#### Procedure: How to Test the Outbound Channel Application

1. Copy and then paste your input XML file to the output directory that you have configured (for example, EDIHL7\_Accelerator\HL7\_out), as shown in the following image.

EDIHL7_A	EDIHL7_Accelerator   HL7_out				
r					
-	Name *	Date modified	Туре	Size	
	🕌 IB_Archive	2/12/2015 6:36 AM	File folder		
	IB_Error	2/12/2015 6:36 AM	File folder		
	🕌 IB_Output	2/12/2015 6:36 AM	File folder		
	🕌 IB_Report	2/12/2015 6:36 AM	File folder		
	IB_TransformGood	2/12/2015 6:36 AM	File folder		
	🕌 OB_Archive	10/7/2015 11:02 AM	File folder		
1	DB_Error	5/5/2015 7:44 AM	File folder		
	🕌 OB_Output	10/7/2015 11:02 AM	File folder		
	DB_Report	10/7/2015 11:02 AM	File folder		
	B_TransformGood	2/12/2015 6:36 AM	File folder		
	hl7_adta01c01_001_Sample.xml	10/1/2015 11:43 AM	XML File	20 KB	

2. Look for the acknowledgement message in the EDIHL7\_Accelerator\HL7\_out\OB\_Output directory.

\$	Receiver dP Ax						
Profile: Demo Receiver Profile (localhost: 12002) 💌 🔤 Profiles   🔍 Messages					🔀 Stop		
	Started	From	Time				
<b>~</b>	10:13:21	127.0.0.1	10/12/2015 10:50:45	PDA 10000ART000accident          Y	<b>A</b>		
<b>~</b>	10:33:19	127.0.0.1	10/12/2015 10:50:45	Sending acknowledge			
<b>∠</b>	10:48:23	127.0.0.1	10/12/2015 10:50:45	MSHI^~\&  REGADTIMCM 20151012105045  ACK 00000007 P 2.6			
<b>~</b>	10:50:45	127.0.0.1	10/12/2015 10:50:45	MSA AA 0000007			
			10/12/2015 10:50:45	Waiting for message			
	10/12/2015 10:50:45 Connection dosed				-		
	Time: 00:00:00 N: 0						
2	🕑 Validator   🎭 Watch   🇞 Search   🔶 Sender   🌷 Receiver						

3. Look for the validation report to be received in the output directory that you have configured (for example, EDIHL7\_Accelerator\HL7\_out\OB\_report), as shown in the following image.

EDIHL7_	EDIHL7_Accelerator   HL7_out   OB_Report								
are with 🔻 New folder									
-	<b>-</b>	Name ^	Date modified	Туре	Size				
		validation_hl7_adta01c01_001_Sample_New	10/12/2015 10:55 AM	XML File	22 KB				

Validation reports contain valid (good) or invalid (bad) HL7-formatted data, copies of the input files, and any error messages that may have occurred during the transformation. Typically the name of the validation report can inform you if the transformation has passed or failed. You can also configure the outbound processing to write the data to two different file locations, one for valid date and the other for invalid data.

- 4. If the input data contained any errors, you can review this error data in the output directory that you have configured for error handling (for example, EDIHL7\_Accelerator\HL7\_out \OB\_Error).
- 5. After outbound processing has completed, a copy of the input data that was used during the transformation is stored under the archive directory that you specified (for example, EDIHL7\_Accelerator\HL7\_out\OB\_Archive).

# Chapter 8

# Outbound Processing: XML to HL7 (Using MLLP)

This section describes how to configure a basic outbound message flow for the iWay Integration Solution for EDIHL7 using the Minimal Lower Layer Protocol (MLLP). The message flow represents the movement and tasks in the conversion of a message from XML format to HL7 format.

#### In this chapter:

- Configuring a Channel for HL7 Outbound Processing Using MLLP
- Configuring Register Sets and Registers
- Importing an Ebix Into the Workspace
- Configuring an iWay Integration Application for Outbound Processing
- Setting Outbound HL7 System Registers Using MLLP
- Testing the Outbound Channel Application Using MLLP

# Configuring a Channel for HL7 Outbound Processing Using MLLP

The outbound channel creates an HL7 message from XML and a XML-formatted validation report. The documents are routed to designated folders based on the success or failure results of the transformation.

## Procedure: How to Create a Channel for Outbound Processing

1. Start iWay Integration Tools (iIT).

2. Right-click the Integration Explorer pane, click *New*, and then select *Integration Project* from the context menu, as shown in the following image.



The New Integration Project dialog box opens, as shown in the following image.

A New Integration Project	
Integration Project	
Create a new Integration project.	
Project name EDIHL7_usr_sample_proj	
Project location	
Use default	
Directory C:\OFFSHORE\IT704PATCHHL7SEP17201	5\IT-7.0.4\EDIHL7_usr Browse
Additional options	
Create integration folders	
Target Server Version 7.0.4	Install additional Target Server Version
?	Finish Cancel

3. Enter a name for the Integration Project (for example, *EDIHL7\_usr\_sample\_proj*) in the Project name field, and then click *Finish*.

A new Integration Project node called *EDIHL7\_usr\_sample\_proj* is added to the Integration Explorer.

4. Expand the *EDIHL7\_usr\_sample\_proj* Integration Project node, right-click the *Channels* folder, select *New*, and then click *Channel* from the context menu, as shown in the following image.



5. Enter a name for the Channel (for example, XMLEDIHL7\_OB\_QS\_pFlow\_MLLP\_Channel), and then click *Next*, as shown in the following image.

🦼 Channel Object		_ 🗆 🗙
Channel Genera	al Properties	
Please choose a nam	e and location for this new Channel.	
Project Folder	/EDIHL7_usr_sample_proj/Channels	Browse
Name	XMLEDIHL7_OB_QS_pFlow_Ebix_Channel	
Description		×
Targat Server Version	7.0.4	<u>×</u>
rarget server version	Testall additional Testach Conversion	-
	Create in current folder	
?	< Back Next > Finish	Cancel

The Inbound/Outbound Protocols pane appears.

6. Click the *Add* button to add a File Listener in the Inbound section, as shown in the following image.

🛃 Channel Object							
Inbound/Outbound Protocols							
Specify the inbound and outbound protocols to be used in the channel							
Inbound:	nbound:						
Name	Listener Type	Description					
Have an inlet cr	eated for each inbound protoc	ol					
, more anniet of							
Outbound:			÷ ×				
Name	Emitter Type	Description					
<b>_</b>							
Have an outlet of	reated for each outbound pro	otocol					
0	< Back	Next > Enich	Cancel				
Ū	< Dack	Deve >	Cancer				

7. From the Types Filter list, select *File* and then click *Finish*, as shown in the following image.

d .		_ 🗆 🗙
Define listener type		
Select type of listener you want to create. Alterna box to filter types.	itevily, use search	
Types filter		
Use '*' to narrow filter matching: a*, *b, a*c		
AQ AS1 AS2 AS2 [nonblocking] Backup Heartbeat Server ConnectDirect Console CS3 Email Envoy Exchange FI/E FTP[S] Client (Clear text or SSL FTP Clients) FTP[S] Client (Clear text or SSL FTP Clients) FTP[S] Server (Clear text or SSL FTP Server) HL7-MLLP-Listener		
Accepts documents from files in directories		
?	Finish	Cancel

8. Select the *Have an Inlet created for each inbound protocol* check box to create an Inlet for the channel, as shown in the following image.

A Channel Object				
Inbound/Outbou	ind Protocols			
Specify the inbound	and outbound protocols to be	used in the channel		
				-
Inbound:			÷ 🗙	
Name	Listener Type	Description		
listener.1	File	Accepts documents	from files in	
		_		
Have an inlet cre	sated for each inbound proto	0		
When uncheck	ed, a single inlet will be create	ed containing all the specifie	ed listeners; otherwise,	a separate inlet will be created for each listener.
Outbound:			* ×	
Name	Emitter Type	Description		
Have an outlet of	reated for each outbound pro	otocol		
?	< <u>B</u> ack	Next > Einish	Cancel	

9. Click the *Add* button in the Outbound section to define an emitter.

The Define emitter type pane appears.

10. In the Types filter list, select *Passthrough* to define the passthrough emitter, and then click *Finish*, as shown in the following image.

2		
Define emitter type		
Select type of emitter you want to create. Alternatev box to filter types.	ily, use search	
Types filter		
1		
Use '*' to narrow filter matching: a*, *b, a*c		
FTP[S] Client (Deprecated FTP Clients) HTTP 1.0 [deprecated] HTTP 1.1 [nonblocking] (nhttp) Internal Queue Java Message Service (jmsq) MLLP		-
MQ MQJMS Ordered Queue Passthrough		
print SFTP Client (Secure Shell version FTP Client) Sonic TCP Tibrv		
		<u> </u>
Discards messages		
?	Einish	Cancel

**Note:** Passthrough does not emit data from the listener. Instead, it just passes the control here and does nothing.

11. Select the *Have an outlet created for each outbound protocol* check box to create an outlet for the channel, and then click *Finish*, as shown in the following image.

1	Channel Object						
1 m	Inbound/Outbound Protocols						
	Specify the inbound and out						
	Inbound:	÷ ×					
1	Name	Listener Type	Description				
	listener.1	File	Accepts documents from	n files in			
I.	Have an inlet created fr	ar each inhound protocol					
L	nave an met d'eateu it	a carrino dana protocor					
	Outbound:						
	Name	Emitter Type	Description				
	emitter.1	Passthrough	Discards messages				
Г	Z have an outlet created	for each outbound protocol					
L							
				1			
		< <u>B</u> ack №e	Einish	Cancel			

The Channel Builder pane appears.



12. Under *inlet: inlet.1*, click *listener: listener.1* (*File*) and then expand the *Main* (*Missing 2 required fields*) configuration parameter on the right pane, as shown in the following image.

🔕 koolhost 📑 WALEDRA 7.06_0S_pFlow_MALP_Channel 🔝 📅 🗖							
Channel Builder <u>3 errors detected</u>							
Channel Builder <u>letros detected</u> XHLEDIHL7_OB_QS_pFlow_HLLP_Channel  C channel MAEDUHL7_OB_QS_pFlow_MLLP_Channel  C channel MAEDUHL7_Channel  C	* * *	Istener.1       i         Accepts documents from files in directories       Type: File thange type:         Filter (enter string to filter properties)       Clear					
		Do not unzip ZIP files 🕦					

13. Enter a valid Input Path, Destination, and Removal Destination (optional) path and then select *hI7* in the Suffix Out drop-down list, as shown in the following image.

koshost      StyleDblt7_06_Q5_pFlow_MLP_channel      Channel Builder Process process In Process Incomess Violational				5	2
XHLEDIHL7_OB_QS_pflow_HLLP_Channel         Image: state and	* X 0	Istener.1         Accepts documents from files in directories         Type: File: chance Lype:         Filter (enter string to filter properties) <ul> <li>Clear</li> <li>Traput Path ()</li> <li>sregOMAEDDH4.7.Input)</li> <li>Destination ()</li> <li>sregOMAEDH4.7.GoodOutput)(sreg(basename)_*.H7</li> <li>Removal Destination ()</li> <li>sregOMAEDH4.7.Ardive)</li> <li>Suffix In Filter ()</li> <li>xmi</li> <li>Scan subdirectories ()</li> <li>fisice</li> <li>Do not unzip ZIP files ()</li> <li>fisice</li> <li>Suffix Out ()</li> </ul>	i 2		

14. Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.



- 15. Import the *XMLToEDIHL7\_Ebix\_2\_MLLP* process flow from the local drive or create the one in the EDIHL7\_usr\_sample\_proj directory in the Flows folder.
- 16. Under the *route: route.1(default)* node, select *process: process.1* and click the process icon on the right panel to reference the process flow into your channel, as shown in the following image.



17. Select a process flow from the integration project and then click OK.

A Resource Selection	
EDIHL7_usr_sample_proj EDIHL7_usr_sample_proj E→ Flows MILTOEDIHL7_Ebix_2_MLLP	
Ок	Cancel

**Note:** Process flows should already be built and available in the iIT integration project. They can be found in the EDIHL7\_usr\_sample\_proj directory inside the Flows folder.

For more information, see *Process Flow Used for Outbound Processing: XML to HL7 (MLLP)* on page 204.

Your screen should resemble the following image.

localhost 🗧 *XMLEDIHL7_OB_QS_pFlow_MLLP_Channel 🔀	- 0
S Channel Builder Process: process. 1: Process process. 1' not defined	
XHLEDIH.7_08_QS_pFlow_HILP_Channel         Image: State in the st	process Select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by this channel component  for the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you want to be referenced by the select process from workspace that you wa

 Click the Save icon near the File menu to save your edited listener configuration. You can also press the shortcut key (Ctrl+S) to save your work if you are using a Windows environment.



#### Reference: Process Flow Used for Outbound Processing: XML to HL7 (MLLP)

This section provides an overview of the process flow used for outbound processing: XML to HL7 (MLLP). This process flow (XMLToEDIHL7\_Ebix\_2\_MLLP) is already built and available in the iIT integration project. It is located under the *EDIHL7\_usr\_sample\_proj* node inside the Flows subfolder.

The following image shows the entire outbound process flow, including all of the nodes that are used and their connections.



In this process flow, an HL7 formatted document is read from a validation report file. The XML tags are stripped and the document is written to a directory. Only valid HL7 files are emitted. Error files as well as their input and any error messages can be found in the validation report file.

#### *Reference:* \_hl7ack(): Parse HL7 Acknowledgement Message

The \_hI7ack() iFL function is used in the outbound MLLP process flow. This function parses a HL7 acknowledgement message to return a specific element value. It uses the following format:

_hl7ack(query[,	defaultValue])
-----------------	----------------

Property	Description		
query (Required)	The query property is the name of a specific element in the HL7 acknowledgement message. Specify one of the following allowed values:		
	<b>ackcode</b> . Acknowledgment Code (#01 in segment MSA).		
	<b>ctrlid</b> . Control ID (#02 in segment MSA).		
	<b>sndapp</b> . Sending Application (#03 in segment MSH).		
	<b>sndfacil</b> . Sending Facility (#04 in segment MSH).		
	<b>recapp</b> . Receiving Application (#05 in segment MSH).		
	<b>recfacil</b> . Receiving Facility (#06 in segment MSH).		
defaultValue	The default value for the queried element.		
(Optional)			

#### Procedure: How to Update Basic Details for Channel Components

1. Select the channel component *inlet:inlet.1* from the channel and click the update channel component icon on the top right side panel, as shown in the following image.



2. Rename or update the inlet details (for example, XMLEDIHL7\_OB\_QS\_pFlow\_MLLP\_Inlet), and click *OK*, as shown in the following image.



After renaming or changing the channel components, your iIT Channel Builder should resemble the following image.

localhost AMLEDIHL7_OB_QS_pFlow_MLLP_Channel				- 0
Channel Builder				
XMLEDIHL7_08_QS_pFlow_MLLP_Channel		XMLEDIHL7_OB_Q	S_pFlow_MLLP_Channel	i
	*	Tick the check-box b	elow to enable or disable dynamic routing for this channel pendencies, such as Ebixes and Registers:	
<sup> </sup>	€ ₹	Type	Location	

## **Configuring Register Sets and Registers**

This section describes how to configure register sets and registers using iWay Integration Tools (iIT).

#### Procedure: How to Configure Register Sets and Registers

1. In the EDIHL7\_usr\_sample\_proj project tree, right-click the Registers folder, and select *New*, and then click *Register Set* from the context menu, as shown in the following image.



2. In the Name field, enter a name for the register set and click *Finish*, as shown in the following image.

🔬 New Register Set	Wizard	_ 🗆 ×
General Propert	ies	
Please select a projec	t location and choose a name for the new Register Set	
Project Folder	/EDIHL7_usr_sample_proj/Registers	Browse
Name	XMLEDIHL7	
Description		
Target Server Version	7.0.4 Install additional Target Server Version	•
	Create in current folder	
?	Finish	Cancel

The new register set appears under Registers in the Registers folder, as shown in the following image.

🔏 Inte 🕮 😈 Wa 🛋 Libra 💆 🗆	Violation State Number of the second state of
(> &   = 😫 🏹	
EDIHL7_usr_sample_proj	
🗁 Adapters	Name
- 🧀 Applications	
🖻 🗁 Channels	
Image: State St	
- 🗁 Ebixes	
🗄 🗁 Flows	
D 20 XMLToEDIHL7_Ebix_2_MLLP	
E 🗁 Registers	
XMLEDIHL 7	
- 🧀 Schemas	
🧀 Transforms	
🗁 XML	

3. Click the Add a property icon to add a register to the register set.

🕯 Inte 🛛 🤨 iWa 🛋 Libra 🖓 🗆	Violation State in the second state in the second state in the second state is a secon	pFlow_MLLP_Channel	7.iwr 🕄		
는 🔶 😧 😑 😵 🏹					
EDIHL7_usr_sample_proj					
- 🗁 Adapters	Name	Туре	Value	Description	
- 🗁 Applications					
🖻 🗁 Channels					
E = XMLEDIHL7_OB_QS_pFlow_MLLP_					
- 🗁 Ebixes					
E 🗁 Flows					
Image: Strategy American St					
Ceres					
XMLEDIHL7					
- 🗁 Schemas					
- 🗁 Transforms					
- 🗁 XML					

4. Enter a name for the new register in the Name field, select a register type from the Type drop-down list (set to string by default), and then enter a value in the Value field, as shown in the following image.

🦼 Register	r Wizard	X
New Reg	jister	
Enter detai	ls regarding the new register.	
Name:	Archive	
Type:	string	
Value:	sreg(EDIHL7_OUTPUT)\OB_Archive	
		A
Description:		
		▼.
?	]	OK Cancel
$\mathbf{\overline{\mathbf{C}}}$	L	

- 5. Click OK.
- 6. Create the following registers under the XMLEDIHL7 registers set you just created, along with the values shown in the table below:

Register Name	Value	
Input	Name=Input, Value=sreg(EDIHL7_OUTPUT)	
Output	Name=GoodOutput, Value=sreg(EDIHL7_OUTPUT) \OB_TransformGood	
Archive	Name=Archive, Value=sreg(EDIHL7_OUTPUT)\OB_Archive	
ValidationReport	Name=ValidationReport, Value=sreg(EDIHL7_OUTPUT) \OB_Report	
Error	Name=Error, Value=sreg(EDIHL7_OUTPUT)\OB_Error	
GoodOutput	Name=Output, Value=sreg(EDIHL7_OUTPUT) \OB_TransformGood	

Register Name	Value
Ack	Name=Ack, Value=sreg(EDIHL7_OUTPUT)\OB_OUTPUT

#### *Procedure:* How to Add a Register Set to an Outbound Channel as a Dependency

1. Click the *channel: XMLEDIHL7\_OB\_QS\_pFlow\_MLLP\_Channel* node and then click the Add dependency icon on the far right side panel of the channel properties pane, as shown in the following image.

localhost = XMLEDIHL7_OB_QS_pFlow_MLLP_Channel 23			- 0
Channel Builder			•
XMLEDIHL7_OB_QS_pFlow_MLLP_Channel		XMLEDIHL7_08_QS_pFlow_MLLP_Channel	i 💈
□       channek: NMEDINE7_COB_CGS_priow_MLIP_Inter         □       □         □	•	Tick the check-box below to enable or disable dynamic routing for this channel.	↓ ◆ ※ ↔

2. From the Registers folder under the integration folder, select XMLEDIHL7 and click OK.

A Resource Selection	
EDIHL7_usr_sample_proj	
Э ок	Cancel

The iIT page should resemble the following image.

localhost				- 0
Channel Builder				::
XMLEDIHL7_0B_QS_pFlow_MLLP_Channel		XMLEDIHL7_OB_QS_pFlor	w_MLLP_Channel	i 🛛
Channel: MAEDINE.7_CB_CS_PFlow_MLP_Channel     Similar MAEDINE.7_CB_CS_PFlow_MLP_Inter	<ul> <li>↓</li> <li>↓</li> <li>↓</li> <li>↓</li> </ul>	Tick the check-box below to	enable or disable dynamic routing for this channel. les, such as Ebixes and Registers: Location /EDIH:17_usr_sample_proj/Registers/MILEDIH:7	

3. Click the Save icon to save your changes. You can also use the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

à locahost StMLEDDLT_08_QS_pFlow_MLIP_Channel ⊠					
Channel Builder					
XMLEDIHL7_08_QS_pFlow_MLLP_Channel		XMLEDIHL7_OB_QS	_pFlow_MLLP_Channel	i 💋	
□     channek. XMLEDIKIZ, OB_QS_prilow_MLP_Charnel       □     •%       intel: XMLEDIKIZ, OB_QS_prilow_MLP_Linter       □     •%       intel: XMLEDIKIZ, OB_QS_prilow_MLP_Linter       □     •%       intel: XMLEDIKIZ, OB_QS_prilow_MLP_Linter       □     •%       intel: XMLEDIKIZ, OB_QS_prilow_MLP_Linter	]	Tick the check-box below to enable or disable dynamic routing for this channel. Dynamic Routing Manage channel dependencies, such as Ebixes and Registers:			
E the outlet: XMLF0EUTL7_EDIX_2_MLP	1	Туре	Location		
Software Matter Metallik 2.08 (25 prim Juli 2000) Software Metallik 2.08 (25 prim MLP Emilter (Passtbrough)	]	register	/EDIHL7_usr_sample_proj/Registers/MILEDIHL7		

# Importing an Ebix Into the Workspace

This section describes how to import an Ebix into the workspace using iWay Integration Tools (iIT).

#### Procedure: How to Import an Ebix Into the Workspace

1. In the integration project EDIHL7\_usr\_sample\_proj, right-click the Ebixes folder and then select *Import* from the context menu, as shown in the following image.



2. Expand the iWay Integration folder, select *Ebix*, and then click *Next*, as shown in the following image.



3. Click the ellipses (...) button to browse and import the ebix from a specific folder location, as shown in the following image.

nport 📝		
General Propert	ies Page equired.	
Project Folder	/EDIHL7_usr_sample_proj/Ebixes	Browse
Import	[	]
Name		
Description		
Target Server Version	704	
	Install additional Target Server Version Create in current folder	
?	< Back Next > Einish	Cancel

4. Select *HL7\_2.6 ebix* from the folder location and click *Open*, as shown in the following image.



- 5. In the Import wizard, click Next.
- 6. Expand the hI7\_2.6 ebix from the left panel and select the 2.6 folder.
7. Select the messages you wish from the right side panel and click *Finish*, as shown in the following image.

at Import		
Ebix Entrice		
Colort chineses	e colleble e bio e state e	
Select ebix version to view	available ebix entries.	
Transform ebix are special	y designed archive files that contain ebix entries. E	bix entry
Select ebix version to view	available ebix entries.	n engine.
Import as System Ebiy		
i inport as System Loix		
libix	Ebix Entries	
□ 🖶 hl7_2.6	ACK	<b>_</b>
2.6	ADR_A19	
	ADT ADT	
	ADT_A03	
	ADT A04	
	ADT ADS	
	ADT_A07	
	ADT_A08	
	ADT_A09	
	ADT_AIO	
1	HA ADT A12	-
Description:		
Entry: ADT A05		Â
Run Time Mode: N/A		
Description:		<b>*</b>
		1



The iIT page should resemble the following image.

## Procedure: How to Add an Ebix to an Outbound Channel as a Dependency

1. Click the channel: XMLEDIHL7\_OB\_QS\_pFlow\_MLLP\_Channel node and then click the Add dependency icon on the right side panel, as shown in the following image.



2. Expand *EDIHL7\_usr\_sample\_proj*, *Ebixes*, *HL7*, and then select ebix *hl7\_2.6* and click *OK*, as shown in the following image.

A Resource Selection		
EDIHL7_usr_sample_pro EDIHL7_usr_sample_pro Ebixes HL7 H7_2.6 Registers	oj	
•	ОК	Cancel

3. Click the Save button to save your changes or press the keyboard shortcut (Ctrl+S) if you are using a Windows environment.

Või localhost				- 0
Channel Builder				•••
XMLEDIHL7_OB_QS_pFlow_MLLP_Channel		XMLEDIHL7_OB_QS_p	Flow_MLLP_Channel	i 🖉
B = Schannet: JMEDIHE7_CB_QS_pFlow_MLIP_Channel ⇒ 4 <sup>™</sup> <sub>a</sub> inlet: JMEDIHE7_CB_QS_pFlow_MLIP_Inlet ↓ 4 <sup>™</sup> <sub>a</sub> is listener: JMEDINE7_CB_QS_pFlow_MLIP_Interner (File) B = Fronte: JMEDINE7_CB_QS_pFlow_MLIP_Route (default) ↓ 4 <sup>™</sup> <sub>a</sub> management MLIP_ROUE (default)	÷ ×	Tick the check-box below Dynamic Routing Manage channel depend	to enable or disable dynamic routing for this channel. encies, such as Ebixes and Registers:	_
□· 🖫 outlet: XMLEDIHL7_OB_QS_pFlow_MLIP_Outlet	Ŷ	Туре	Location	
emitter: XMLEDIHL7_OB_QS_pFlow_MLLP_Emitter (Passthrough)	n	register	/EDIHL7_usr_sample_proj/Registers/XMLEDIHL7 /EDIHL7_usr_sample_proj/Ebixes/hl7_2.6	×
				Ŷ
				÷

# Configuring an iWay Integration Application for Outbound Processing

This section describes how to configure an iWay Integration Application (iIA) for outbound processing using iWay Integration Tools (iIT).

# *Procedure:* How to Configure an iWay Integration Application for Outbound Processing

1. Right-click the Applications folder under the EDIHL7\_usr\_sample\_proj integration project, select *New*, and then click *Application* from the context menu, as shown in the following image.



2. Enter a value in Name field (for example, EDIHL7\_usr\_sample\_App), and click *Next*, as shown in the following image.

Application Wizard		_ 🗆 ×
General Properties		
Please select a project location	n and choose a name for the new application	
Project Folder	/EDIHL7_usr_sample_proj/Applications	Browse
Name	EDIHL7_usr_sample_App	
Description		<b>^</b>
		Ψ.
Target Server Version	7.0.4	<b>v</b>
	Install additional Target Server Version	_
	Create in current folder	
	Use Maven	
		a
0	< Back Next > Finish	Cancel

3. Select the *XMLEDIHL7\_OB\_QS\_pFlow\_MLLP\_Channel* check box from Resource Selection pane and keep clicking *Next* until you get to the Bindings pane, as shown in the following image.

A New App	lication Wizard						_ 🗆 🗵
Resource	e Selection						
Add channe	els, transforms and pr	rocesses t	o vour ac	plication.			
			. , ,				
Name			Type	Aut	Location		Select All
	EDIHL7_OB_QS_pFk	W_MLLP_	Channel	ves	/EDIHL7 usr .		
	ToEDIHL7_Ebix_2_M	LLP	proc		/EDIHL7_usr	••	Deselect All
R				1		ЪL	
?		< Ba	dk 📗	Next >	Finish		Cancel

4. In the Bindings pane, add Bindings for Registers you have added in the XMLEDIHL7 Register Set to the iIA EDIHL7\_usr\_sample\_App and click *Finish*, as shown in the following image.

🔏 New Applicatio	on Wizar	d						_ 🗆 🗙
Bindings								
List the application resolved at deploy	bindings ment tim	, such as Specia e.	al Reg	gisters and F	Providers	, to be		
Name	Туре	Default Value		Description				
- 2								
New App	licatio	n Binding						•
Enter detail	s regardir	ng the new app	icatio	on binding				5
								P
Name	: Input							_
Туре	: Regist	er						<u> </u>
Default Value	: sreg(E	DIHL7_OUTPU	т)					
Description								*
								<u> </u>
• ?					0	ĸ	Cano	8
?		< <u>B</u> ac	k	Next >		Einish		Cancel

After adding all register bindings, your iIT screen should resemble the following image.

XMLEDIHL7_OB_QS_pFlow_N	4LLP_Channel 👸 XM	LTOEDIHL7_Ebix_2_MLLP	KMLEDIHL7.iwr 🔅 EDIHL7_usr_sample_App.iab 🛛
Add application bindings, such	as Special Registers and Pr	oviders, to be resolved at deployme	ent time.
Name	Type	Default Value	Description
Input	Register	sreg(EDIHL7_OUTPUT)	
Output	Register	sreg(EDIHL7_OUTPUT)\OB_T	Transfor
GoodOutput	Register	sreg(EDIHL7_OUTPUT)\OB_T	Transfor
Archive	Register	sreg(EDIHL7_OUTPUT)\OB_A	Archive
ValidationReport	Register	sreg(EDIHL7_OUTPUT)\OB_R	Report
Error	Register	sreg(EDIHL7_OUTPUT)\OB_E	Error
Adk	Register	sreg(EDIHL7_OUTPUT)\OB_C	Dutput

5. To build the application, right-click *EDIHL7\_usr\_sample\_App*, select *Integration Tools*, and then click *Build* from the context menu, as shown in the following image.

្ទ Integration - EDIHL7_UST_Sample_proj/App	pilcations/EDIHL7_USF_sample_App.ia	D/EDIHL/_USP_Sam	прие_Аррлав - гіча	y integration Loois	
We Edit Navigate Search Project Run Wind	dow Help				
😂 • 🛛 🕾 🛯 🕙 🖉 🖉 🖉 🛤 🗍 🕸	🏂 • 🗿 • 🎭 • ] 🔗 • ] 🖉 • i	l	Ψ		
🗄 Outine 🔏 Integration Explorer 🐰 🗖 🗖	SMLEDIHL7_08_QS_pFlow_MLLP_Channel	el 🎯 EDIHL 7_u	sr_sample_App.iab	22	
	Add channels, transforms and processes	o your application. Au	utostart column lets y	ou control channel start behavior on serve	er startup.
EDIHL7_usr_sample_proj	Name	Type	Auto Start	Location	Description
Adapters	SMLEDIHL 7_08_Q5_pFlow_MLLP_Cha	inlineChannel	yes	/EDIHL7_usr_sample_proj/Channels/	
EDDHL7 usr sample App					
🗄 🥭 Components					
E Bindings					
一张) build.xml					
EDIHL7_usr_sample_App.ia					
B B VM EDDI 7 OB OS pElow MUR					
E- Ebixes					
🖻 🇁 HL7					
🕀 🏭 H7_2.6					
E- Constant Flows					
. XMLTOEDHL7_Ebix_2_MLLP					
Kegsters					
Schemas					
Transforms					
- 🗁 XML					
Console 23					
T Message Console			_		
[INFO]09:30:22 Application 'EDIH	HL7_usr_sample_App' built st	ccessfully.			

- 6. To publish the iIA, right-click *EDIHL7\_usr\_sample\_App*, select *Integration Tools*, and then click *Publish to* from the context menu.
- 7. Provide the iSM server details in the Server URL text box and the other credentials, then click *Finish*.

The iIT page should resemble the following image.

d Integration - EDIHL7 use sample proj/App	lications/FDIHL7 usr sample Applia	h/FDTHL7 usr	sample Appliab - il	Way Integration Tools	
File Edit Navigate Search Project Run Windo	ow Help				
📫 • 🖩 🗞 🛆 ] 😤 🌌 😫 ] 🛲 ] 🦻	» • Ø • <b>%</b> • ] ∦ • ] ∦ • }	a - +- <b>↓</b> -	$\rightarrow$ -		
📴 Outline 🔬 Integration Explorer 🕺 🗖 🗍	XMLEDIHL7_OB_QS_pFlow_MLLP_Channel	el 🔅 EDIHL	7_usr_sample_App.ia	ЬX	
← → @   🖻 🕵 🏹	Add channels, transforms and processes	to your application	. Autostart column let	s you control channel start behavior on serv	er startup.
EDIHL7_usr_sample_proj	Les.	-	1	I	In the
Adapters	Name	Туре	Auto Start	Location	Description
Applications	SWILEDTHE / OB_QS_DHOW_MLLP_CH	inineChannel	yes	/EDIHL/_usr_sample_proj/Channels/	
General EDIHL7_usr_sample_App					
E 🔁 Components					
Image Bindings					
······································					
EDIHL7_usr_sample_App.ia					
Channels					
C S Things					
Elixes					
H					
E Bows					
E 20 XMLToEDIHL7 Ebix 2 MLLP					
E 🗁 Registers					
XMLEDIHL7					
Company C					
XML					
E Concola S?					
IT Message Console					
[INFO]09:30:22 Application 'EDIH	L7 usr sample App' built s	accessfully			
[INFO]09:32:49 Application 'EDIH	L7_usr_sample_App' success:	fully built	and published	d to http://localhost:9000	

- 8. Deploy the iIA application on the iSM registry.
- 9. Have the following folder structure created before starting your application in the iSM console, as shown in the following image.

HL7_out				
G O v Computer ▼ OS (C:) ▼ EDIHL7_Accelerator ▼ HL7_out ▼				
Organize 💌 🎧 Open 🛛 Include in library 💌	Share with 🔻 New folder			
EDIHL7_Accelerator	▲ Name ^	Date modified		
鷆 HL7_in	IB Archive	2/12/2015 6:36 AM		
HL7_out	IB Error	2/12/2015 6:36 AM		
IB_Archive	IB_Output	2/12/2015 6:36 AM		
IB_Cutout	IB_Report	2/12/2015 6:36 AM		
IB Report	IB_TransformGood	2/12/2015 6:36 AM		
IB_TransformGood	B_Archive	10/7/2015 11:02 AM		
B_Archive	OB_Error	5/5/2015 7:44 AM		
OB_Error	0B_Output	10/7/2015 11:02 AM		
) OB_Output	B OB Report	10/7/2015 11:02 AM		
) OB_Report	B OB TransformGood	2/12/2015 6:36 AM		
OB_TransformGood		2, 12, 2010 0100 / 0 /		

# Setting Outbound HL7 System Registers Using MLLP

This section describes how to set system registers using the iWay Service Manager (iSM) Administration Console.

#### Procedure: How to Set System Registers

- 1. Open the iSM Administration Console and select *EDIHL7\_usr\_sample\_App\_MLLP\_OB* [down] from the Management drop-down list.
- 2. From the console bar, click Server, Register Settings, and then click Add, as shown in the following image.

Properties General Properties Java Properties	Register Settings Special registers are name available to all components Listed below are the register Special Registers	d variables that reference values which , of the system. Any changes to the regi er settings for the HL7_usr_samples_App	are carried throughout the system. Once defined, ster settings do not take effect until the server is re configuration of this server.	hese variables estarted/redep
Settings	Name	Value	Description	Туре
Console Settings	iwayversion	unavailable	system defined (readonly)	string
Java Settings	iwayhome	unavailable	system defined (readonly)	string
Register Settings	iwaydata	unavailable	system defined (readonly)	string

3. For the deployed EDIHL7\_usr\_sample\_App\_MLLP\_OB application, define the system registers as listed in the following table:

Register Name	Value
EDIHL7_INPUT	sreg(EDIHL7_Installdir)\HL7_in
EDIHL7_installdir	C:\EDIHL7_Accelerator
EDIHL7_OUTPUT	<pre>sreg(EDIHL7_Installdir)\HL7_out</pre>

Register Name	Value
ValidateEDIHL7	true

#### **Register Settings**

Special Register De	finition
Name *	Enter the name of the special register to add.
	EDIHL7_Installdir
Туре	Select a type for the value of this special register.
	string
Value *	Enter a value for this special register. The value can be a constant or a call to the evaluation functions.
	C:\EDIHL7_Accelerator
Description	Enter a description for this appoint conjector
Description	Enter a description for this special register.

After adding the registers, your Register Settings page in the iSM Administration Console should resemble the following image.

iway.serverfullhost	iwhl7.ibi.com	system defined (readonly)	string
iway.pid	1668	system defined (readonly)	string
EDIHL7_INPUT	sreg(EDIHL7_Installdir)\HL7_in		string
EDIHL7_Installdir	C:\EDIHL7_Accelerator		string
EDIHL7_OUTPUT	sreg(EDIHL7_Installdir)\HL7_out		string
ValidateEDIHL7	true		string

4. Start the deployed application in the iSM Administration Console and ensure that the channel is up and running in the Monitoring section of the console.

# Testing the Outbound Channel Application Using MLLP

This section describes how to test the outbound channel application using the Minimal Lower Layer Protocol (MLLP).

# Procedure: How to Test the Outbound Channel Application Using MLLP

1. Copy and then paste your input XML file to the output directory that you have configured (for example, EDIHL7\_Accelerator\HL7\_out), as shown in the following image.

EDIHL	EDIHL7_Accelerator + HL7_out +					
r						
	≜	Name *	Date modified	Туре	Size	
		🕌 IB_Archive	2/12/2015 6:36 AM	File folder		
		)) IB_Error	2/12/2015 6:36 AM	File folder		
		鷆 IB_Output	2/12/2015 6:36 AM	File folder		
		鷆 IB_Report	2/12/2015 6:36 AM	File folder		
		鷆 IB_TransformGood	2/12/2015 6:36 AM	File folder		
		DB_Archive	10/7/2015 11:02 AM	File folder		
	1	DB_Error	5/5/2015 7:44 AM	File folder		
		DB_Output	10/7/2015 11:02 AM	File folder		
		DB_Report	10/7/2015 11:02 AM	File folder		
		B_TransformGood	2/12/2015 6:36 AM	File folder		
		hl7_adta01c01_001_Sample.xml	10/1/2015 11:43 AM	XML File	20 KB	

2. Look for the acknowledgement message in the EDIHL7\_Accelerator\HL7\_out\OB\_Output directory.

\$	Receiver				9 th X
Pro	file: Demo Re	ceiver Profile (localhost: 12002)	💌 📗 Profiles   🔍	Messages	📕 Stop
	Started	From	Time		
<b>~</b>	10:13:21	127.0.0.1	10/12/2015 10:50:45	PDA 10000ART000accident             Y	-
<b>~</b>	10:33:19	127.0.0.1	10/12/2015 10:50:45	Sending acknowledge	
<b>~</b>	10:48:23	127.0.0.1	10/12/2015 10:50:45	MSH ^~\&  REGADT MCM 20151012105045  ACK 00000007 P 2.6	
~	10:50:45	127.0.0.1	10/12/2015 10:50:45	MSA AA 0000007	
			10/12/2015 10:50:45	Waiting for message	
			10/12/2015 10:50:45	Connection closed	-
			Time: 00:00:00 N:	0	
2	Validator 🛛 🤻	• Watch   🕸 Search   🔶 Sender	😕 Receiver		

The \_hI7ack() iFL function is used in the outbound MLLP process flow. This function parses a HL7 acknowledgement message to return a specific element value. It uses the following format:

\_hl7ack(query[,defaultValue])

For more information on using this function, see \_hI7ack(): Parse HL7 Acknowledgement Message on page 205.

The \_hl7ack() iFL function will return "AA", "AE", "AR", "CA", "CE", "CR" acknowledgement codes. Based on this code the acknowledgement will be stored in different locations.

 Look for the validation report to be received in the output directory that you have configured (for example, EDIHL7\_Accelerator\HL7\_out\OB\_report), as shown in the following image.



Validation reports contain valid (good) or invalid (bad) HL7-formatted data, copies of the input files, and any error messages that may have occurred during the transformation. Typically the name of the validation report can inform you if the transformation has passed or failed. You can also configure the outbound processing to write the data to two different file locations, one for valid date and the other for invalid data.

- 4. If the input data contained any errors, you can review this error data in the output directory that you have configured for error handling (for example, EDIHL7\_Accelerator\HL7\_out \OB\_Error).
- 5. After outbound processing has completed, a copy of the input data that was used during the transformation is stored under the archive directory that you specified (for example, EDIHL7\_Accelerator\HL7\_out\OB\_Archive).



# **Supported HL7 Versions**

This section summarizes the HL7 versions that are currently supported by the iWay Integration Solution for EDIHL7.

#### In this appendix:

Supported HL7 Versions and Messages

# Supported HL7 Versions and Messages

The following table lists the HL7 versions and messages that are supported by the iWay Integration Solution for EDIHL7.

HL7 Version	Supported Messages
2.3	ALL
2.3.1	ALL
2.4	ALL
2.5	ALL
2.5.1	ALL
2.6	ALL
2.7	ALL
2.7.1	ALL



# **Using HL7 Separators and Terminators**

All HL7 documents are embedded with tokens that are separated by special characters called separators and terminators. Specifically, these special characters are used to identify:

- element separators
- □ sub-element separators
- segment terminators

This appendix provides a list of the separators and terminators that are allowed during the configuration of preparsers and preemitters in iWay Service Manager (iSM) and iWay Integration Tools (iIT).

#### In this appendix:

HL7 Separators and Terminators

Hex	Char	Hex	Char	Hex	Char
01	SOH	16	SYN	2F	/
02	STX	17	ETB	ЗА	:
03	ETX	18	CAN	3B	;
04	EOT	19	EM	ЗC	<
05	ENQ	1A	SUB	ЗD	=
06	АСК	1B	ESC	3E	>
07	BEL	1C	FS	ЗF	?
08	BS	1D	GS	40	@
09	ТАВ	1E	RS	5B	[

# **HL7** Separators and Terminators

Hex	Char	Hex	Char	Hex	Char
OA	LF	1F	US	5C	١
OB	VT	21	!	5D	]
OC	FF	23	#	5E	٨
OD	CR	24	\$	5F	_
OE	SO	25	%	60	
OF	SI	26	&	7B	{
10	DLE	27	,	7C	
11	DC1	28	(	7D	}
12	DC2	29	)	7E	~
13	DC3	2A	*	7F	DEL
14	DC4	2B	+		
15	NAK	2D	-		



# Using iWay Integration Tools to Configure an Ebix for HL7

This section describes how to use iWay Integration Tools (iIT) to configure an Ebix for HL7.

#### In this appendix:

- Using iIT to Configure an Ebix File for HL7Overview
- Using iIT to Configure an Ebix File for HL7 Prerequisites
- Loading an Ebix
- Working With iWay Integration Tools (iIT)

# Using iIT to Configure an Ebix File for HL7Overview

You can use iWay Integration Tools (iIT) to import, edit, export, and work with Ebixes for HL7. The topics in this appendix describe how to:

Import an HL7 version 2.6 Ebix into iIT.

- Add a ZMS segment at the end of an ADT\_A01 message to the HL7 version 2.6 Ebix.
- □ Export the edited Ebix to a physical location.

The edited Ebix can be returned and then tested with the appropriate HL7 version 2.6 ADT\_A01 message.

#### Using iIT to Configure an Ebix File for HL7 Prerequisites

The following list shows the prerequisites for using iWay Integration Tools (iIT) to configure an Ebix for HL7:

A working knowledge of iWay Service Manager (iSM), iWay Integration Tools (iIT), and HL7.

- iSM Version 8.0.
- □ iWay Integration Solution for EDIHL7 (Patch).
- iIT Version 8.0.

# Loading an Ebix

This section describes how to load an Ebix.

# *Procedure:* How to Load an Ebix

To load or import an Ebix:

- 1. Download the HL7\_ebixes.zip file from http://techsupport.informationbuilders.com.
- 2. Unzip the downloaded HL7\_ebixes.zip and save *HL7\_2.6.ebx* into any physical location on your local drive.

For example, this Ebix contains ADT\_A01 in it. Make sure all folders used for HL7\_2.6.ebx do not have any blank spaces in the folder name.

S S S S S S S S S S S S S S S S S S S							
Organize   Include in library   Share with	Organize   Include in library   Share with   New folder						
☆ Favorites	Name ^	Date modified	Туре	Size			
📃 Desktop	HL7_2.6.ebx	3/25/2015 5:22 PM	EBX File	20,151 KB			
🐞 Downloads							
🔢 Recent Places							

# Working With iWay Integration Tools (iIT)

This section describes how to import, edit, and export an Ebix using iWay Integration Tools (iIT).

## *Procedure:* How to Import an Ebix

1. Start iWay Integration Tools (iIT).

2. Right-click the Integration Explorer pane, click *New*, and select *Integration Project* from the context menu.



3. Enter a new Integration Project name, for example, *HL7\_Ebix\_edit\_sample\_proj*, in the Project name field, and click *Finish*.

🦼 New Integration Project 📃 🗆 🗙
Integration Project
Create a new Integration project.
Project name   HL7_Ebix_edit_sample_proj
Project location
I Use default
Directory C:\OFFSHORE\iIT_Builds\7.0.4_nightly_30-mar-2015\iIT-7.0.4\wo Browse
Create integration folders
Target Server Version 7.0.4-SNAPSHOT
Install additional Target Server Version
Finish Cancel

4. Right-click the Integration Explorer pane and select *Import* from the context menu, as shown in the following image.



5. In the Import wizard, expand *iWay Integration*, select *Ebix*, and then click *Next*, as shown in the following image.



6. Click the *ellipsis button* (...) and select the downloaded *HIPAA\_5010X299.ebx* file from the physical drive location.

al Import		_ 🗆 🗵
General Prope	rties Page	
🙆 The name field is	required.	
Project Folder	/HL7_Ebix_edit_sample_proj/Ebixes	Browse
Import	[	
Name	1	
Description		-
		*
Target Server Versio	D. C. 4-SNAPSHOT	-
	Install additional Target Server Version	
	Create in current folder	
?	< Back Next > Einish	Cancel
-		

7. Click Open, as shown in the following image.

🧖 Open							×
🕥 🖟 • os	(C:) * OFFSHORE * Ebix_B	uilds - HL7	÷ 🙀	Search HL7			2
Organize 👻 New fo	lder				8≡ -		0
🔆 Favorites	▲ Name ^		Date r	nodified	Туре		
Desktop Downloads Downloads Elbraries Documents Music Pictures Subversion	HL7_2.6.ebx		3/25/	2015 5:22 PM	EBX File		
💻 Computer							
🏭 OS (C:)							
C on THIRU							
🖵 D on THIRU	<b>▼ ∢</b>						Þ
	File name:		•	Ebix File (*.ebx		incel	• 

8. Click *Next*, as shown in the following image.

A Import						
General Properties Page						
Please enter a name	Please enter a name and description for this imported ebix.					
Project Folder	/HL7_Ebix_edit_sample_proj/Ebixes	Browse				
Import	C:\OFFSHORE\Ebix_Builds\HL7\HL7_2.6.ebx	<u></u>				
Name	HL7_2.6					
Description		<u></u>				
		No.				
		-				
Target Server Versio	7.0.4-SNAPSHOT	-				
	Install additional Target Server Version					
	Create in current folder					
?	< Back Next > Enish	Cancel				

9. Expand the Ebix showed in the Ebix pane, select *ADT\_A01* from the Ebix Entries pane, and then click *Finish*.

对 Import		X		
Ebix Entries Select ebix version to view available ebix entries.				
Transform ebix are specially designed archive files that contain ebix entries. Ebix entry represents transform configuration and dependencies used by iWay transform engine. Select ebix version to view available ebix entries.				
Ebix 	Ebix Entries			
Description: Entry: ADT_A01 Run Time Mode: N/A Description:		]		
?	< Back			



Your iWay Integration Tool interface should resemble the following image:

#### *Procedure:* How to Edit an Ebix

1. Click the *HL7\_ADT\_A01\_2.6* tab to open the HL7 Ebix structure.

2. Right-click the ADT\_A01 [Admit/Visit Notification (Event A01)] element, select Add, and then click Segment, as shown in the following image.



You screen should resemble the following image.



 Provide valid segment name (for example, ZMS) and description (for example, Miscellaneous patient data) by double-clicking on the value field of its property, as shown in the following image.

🗖 Properties 🔀 🧐 Error Log 📮 Console 🐕 Problems					
[Segment]					
General	Property	Value			
General	Name	ZMS			
	Description	Miscellaneous patient data			
	Туре				
	Reg	0			
	MaxUse	1			
	Notes				

You screen should resemble the following image.



4. Right-click ZMS [Miscellaneous patient data] to add Data Elements in it, select Add, and then click Data Element, as shown in the following image.

PV1 [Patient Visit Segment]     PV2 [Patient Visit - Additional	Add 🔸	🚸 Data Element
ARV [Access Restrictions Se	Export to Library	A Composite Element
E Segment] 	X Delete	
OBX [Observation/Result Se	칩 Move Up	
🕀 💁 AL1 [Patient Allergy Informa	Move Down	
DG1 [Diagnosis Segment]     DRC [Diagnosis Related Cross	Validate	
	Preview Dictionary Structure	
🗄 🛂 GT1 [Guarantor Segment]	Validate Structure	
ACC [Accident Segment]	Expand	
🕒 🋂 UB1 [Uniform Billing 1 Segme	Expand All	
🗉 🤷 UB2 [UB92 Data Segment]	Collapse	
🗉 🛂 PDA [Patient Death And Aut		
ZMS [Miscellaneous patient of	Properties	J
HL7Header Structure:HL7_ADT_A01	L_2.6 Schema:HL7_ADT_A01_2.6 F	Rule:HL7_ADT_A01_2.6_Rules

5. Right-click the newly created Data Element and select Properties.

	행 ACC [Accident Segment] 행 UB1 [Uniform Billing 1 Segment] 행 UB2 [UB92 Data Segment] 행 PDA [Patient Death And Autopsy	Segment]
	2MS [Miscellaneous patient data] Data Element	
Header:HL7	Header Structure:HL7_ADT_A01_2.6	Schema:HL7_ADT_A01_2.6 Rule:HL7_ADT_A01_2.6_Rules HL7_ADT_A01_2.6toXML
🗆 Propertie	s 🔀 🥺 Error Log 📮 Console 🖇	2 Problems
[Data Ele	ment]	
Ceneral	Property	Value
General	Name	Data Element
	Description	
	Reg	0
	MinLength	1
	MaxLength	
	Pad	True
	PadChar	

6. Change the Name field value to 01, Description value to *Next Visit Date*, MaxLength value to 8, and the Type value to *DT*, as shown in the following image.

Propertie	s 🔀 🔮 Error Log 📮 Console 😡 Problems	
[01]		
Ceneral	Property	Value
General	Name	01
	Description	Next Visit Date
	Req	0
	MinLength	
	MaxLength	8
	Pad	
	PadChar	
	Align	
	Туре	рт
	Notes	
	Domain	

Note: Use Backspace to delete existing values in the value field.

- 7. Repeat step 4 to create the following two data elements:
  - Reason Code
  - Number Of Days Left
- 8. Input the following values into their respective parameters in the second data element (Reason Code) under the ZMS segment.

Parameter	Value
Name	02
Description	Reason Code
Req	0
MaxLength	3
Туре	ST

Parameter	Value
Domain	GEN, RVW, CRD, FOL

🗆 Properties 🕴 💇 Error Log 📮 Console 🙀 Problems				
[02]				
General	Property	(	Value	
General	Name		02	
	Description		Reason Code	
	Req		0	
	MinLength			
	MaxLength		3	
	Pad			
	PadChar			
	Align			
	Туре		ST	
	Notes			
	Domain		GEN,RVW,CRD,FOL	

9. Input the following values into their respective parameters in the third data element (Number Of Days Left) under the ZMS segment.

Parameter	Value
Name	03
Description	Number Of Days Left
Req	0
MaxLength	3
Туре	NM

Properti	ies 🔀 🔮 Error Log 🚍 Console 🙀 Problems	
[03]		
General	Property	Value
General	Name	03
	Description	Number Of Days Left
	Req	0
	MinLength	
	MaxLength	3
	Pad	
	PadChar	
	Align	
	Туре	NM
	Notes	
	Domain	

Your iWay Integration Tool screen should resemble the following image.

tt hl7_2.2.6	.ADT_A01 💥			
	B DB1 [Disability Segment]			
÷.	🗄 📲 OBX [Observation/Result Segment]			
	🗄 🦓 AL1 [Patient Allergy Information Segment]			
÷.	A DG1 [Diagnosis Segment]			
	A DRG [Diagnosis Related Group Segment]			
<b>.</b>	9 PROCEDURE			
₽	A GT1 [Guarantor Segment]			
<b>.</b>	9 INSURANCE			
₽	ACC [Accident Segment]			
÷	UB1 [Uniform Billing 1 Segment]			
	UB2 [UB92 Data Segment]			
<b>+</b>	PDA [Patient Death And Autopsy Segment]			
8	ZMS [Miscellaneous patient data]			
	01 [Next Visit Date]			
	02 [Reason Code]			
	03 [Number Of Days Left]			
Use devil 10 70				
header:hL/r	leader [Structure:hL7_AD1_A01_2.6] Schema:hL7_AD1_A01_2.6   Kule:hL7_AD1_A01_2.6_Kules	HL7_AD1_A01_2.6t0XML		
Propertie	s 🕅 🤨 Error Log 📮 Console 😡 Problems			
[03]				
General	Property	Value		
	Name	03		
	Description	Number Of Days Left		
Req O		0		
MinLength				
	MaxLength 3			
Pad				
PadChar				
Align				
	Туре	NM		
	Notes			
## *Procedure:* How to Export an Ebix

To export an Ebix:

1. Right-click the *h*17\_2.6 Ebix from the Integration Explorer window and select the *Export* option from the context menu, as shown in the following image.



2. Expand the *iWay Integration* folder, select *Ebix*, and then click *Next*, as shown in the following image.



3. Expand the iIT project (for example, *HL7\_Ebix\_edit\_sample\_proj*) from the Export wizard and expand *Ebixes*, *HL7*, *HL7\_2.6*, select the 2.6 folder from the left pane of Export wizard, and then select the *ADT\_A01* check box on the right pane, as shown in the following image.

A Export				_ 🗆 🗙
Export Please enter a destination	n directory.			
•	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,			
HL7_Ebix_edit_san	nple_proj	■ # ADT_A0:	1	
			Select All	Deselect All
To directory:				Browse
?	< <u>B</u> ack	Next >	Einish	Cancel

4. Click *Browse* and choose a folder location to store the Ebix, and then click *Next*, as shown in the following image.

🚀 Export								
Export								
Export Ebix resources to the local file system.								
HL7_Ebix_edit_sar	mple_proj	AD 😫 AD	T_A01					
Channels								
Ebixes								
⊡ <mark>⊞</mark> hl7_2.	6							
2.	6							
Schemas								
Transforms								
			Select All	Deselect All				
To directory: C: Users Administrator Desktop								
?	< <u>B</u> ack	Next >	Einish	Cancel				

5. Provide a valid name for the Ebix in the Name field, select *Pipeline* from the Runtime Mode drop-down list, add a description (optional), and then click *Finish*, as shown in the following image.

n Export				
Export Ebix				
Create a new Ebix				
Create a new transform eb Transform ebix are special and dependencies used by a new ebix and ebix entry	ix by first spec y designed ard iWay transform for specified ty	ifying a name ar hive files that co nation engine. T pe.	nd description of a Intain transform of This wizard will allo	new ebix. onfiguration w you to create
Name:				
Custom_Name_V26_ADT_	A01			
Ebix Type:				
HL7				
Runtime Mode:				
Pipeline				•
Standard				
	_			
Your Description here				
				Ψ.
0	< Pack	Martin	Einich	Cancel
	< DBCK	INEXC >	Finish	Caricel

Your exported Ebix is available in the specified location.

## Feedback

Customer success is our top priority. Connect with us today!

Information Builders Technical Content Management team is comprised of many talented individuals who work together to design and deliver quality technical documentation products. Your feedback supports our ongoing efforts!

You can also preview new innovations to get an early look at new content products and services. Your participation helps us create great experiences for every customer.

To send us feedback or make a connection, contact Sarah Buccellato, Technical Editor, Technical Content Management at Sarah\_Buccellato@ibi.com.

To request permission to repurpose copyrighted material, please contact Frances Gambino, Vice President, Technical Content Management at *Frances\_Gambino@ibi.com*.

Inf%rmation Builders

## iWay

iWay Integration Solution for EDIHL7 User's Guide

Version 7.0.x and Higher

DN3502093.0418

Information Builders, Inc. Two Penn Plaza New York, NY 10121-2898

