# webMethods

# webMethods Chem eStandards Module

## User's Guide

VERSION 6.0.1

# Contents

# About this Book

# Welcome!

This guide describes how to implement Chem eStandards™ processes between you and your trading partners using the webMethods Chem eStandards Module. To use this guide effectively, you should:

- Be familiar with the webMethods Integration Server, the Server Administrator, and webMethods Developer and understand the concepts and procedures described in the *webMethods Integration Server Administrator's Guide* and the *webMethods Developer User's Guide*.

- Be familiar with webMethods Trading Networks Console and understand the concepts and procedures described in the various *webMethods Trading Networks* guides.

- Be familiar with webMethods Modeler and understand the concepts and procedures described in the *webMethods Modeler User's Guide*.

- Have a basic knowledge of the Chemical Industry Data Exchange (CIDX) and Chem eStandards terminology. For more information, see http://www.cidx.org/ or see "Related Documentation" on page 10.

- Have installed the Integration Server, Developer, Trading Networks (server side and console side) software, Modeler (server side and client side) software, and the webMethods Chem eStandards Module software.

# Typographical Conventions

This document uses the following typographical conventions:

| Convention | Example |
| --- | --- |
| Procedures are designated by a blue box in the left column. Procedures are presented as a series of numbered steps. | **To create a flow service**<br><br>1   On the **File** menu, click **New**. |
| Terms that identify elements, options, selections, and commands on the screen are shown in bold. | The **Service** field on the **Properties** tab specifies the name of the requested service. |
| Storage locations for services on the Integration Server are shown in a narrow font using the convention *folder.subfolder:service*. | pub.client:smtp sets a MIME-type e-mail message. |

| Convention | Example |
|---|---|
| Characters that you must type exactly are shown in a typewriter font. | Type: `*Administrators*` |
| Variable information that you must change (based on your specific situation or environment) is shown in italics. | Log on to the proxy server with:<br><br>`USER` *proxy_user*<br>`PASS` *proxy_password* |
| Input and output variables for a service are shown in italics. | A service in the flow takes a document list called *LineItems*. |
| Messages that the system displays on the console are shown in a typewriter font. | The server returns the following error to the user:<br><br>`Server has reached client limit.` |
| Keyboard keys are shown in uppercase. | Press ENTER; then press TAB. |
| Keys that you must press simultaneously are joined with the "+" symbol. | Press CTRL+ALT+M. |
| Directory paths are shown with the "\" directory delimiter unless the subject is UNIX specific. In these cases, the "/" is used. If you are working in a UNIX environment, substitute a "/" for the "\" shown in the procedures in this book. | *webMethods6\IntegrationServer\*config |
| Information that you must read before beginning a procedure or that alerts you to negative consequences of certain actions is presented using this notation. | **Important!** If the folder is not already open in the Navigation Panel, open it before you start the following procedure. |
| Notes that provide related, but non-critical, information are presented using this notation. | **Note:** When you start the product, you are prompted to log on to a webMethods Integration Server. |
| Helpful information (such as shortcuts and alternatives) is presented using this notation. | **Tip!** You can also use CTRL+C to copy an object. |

## Program Code Conventions

For programming code and command syntax, this document uses the following typographical conventions:

| Convention | Example |
|---|---|
| Keywords and values that you must type exactly as printed are shown in typewriter font. | `%CoSymbol%` |
| Variable values or parameters that you must supply are shown in italics. | `%VarName%` |
| Keywords or values that are optional are enclosed in [ ]. Do not type the [ ] symbols in your own code. | `%loop LoopVar [null=NullValue]%` |

# Related Documentation

This section lists the documentation that webMethods provides with the webMethods Chem eStandards Module. In addition, it lists documentation provided with Trading Networks, the Integration Server, Developer, Modeler, and webMethods Monitor that you might find useful.

## webMethods Chem eStandards Module Documentation

The following table lists other documents, in addition to this guide, that webMethods provides with the webMethods Chem eStandards Module.

| Refer to this book… | For… |
|---|---|
| *webMethods Chem eStandards Module Sample Guide* | Information about and instructions for running the sample that is included with the Chem eStandards Module. Location: webMethods6\IntegrationServer\packages\WmChem\pub\doc\wmChem_Sample Guide.pdf |

| Refer to this book… | For… |
|---|---|
| *webMethods Chem eStandards Module Built-In Services Reference Guide* | Information about the built-in services and IS document types that are contained in the Chem eStandards Module.<br><br>Location:<br><br>webMethods6\IntegrationServer\packages\WmChem\pub\doc\wmChem_BuiltIn Services.pdf |

## Trading Networks Documentation

The following table lists manuals that webMethods provides with Trading Networks. Some documents are in PDF format and others are in HTML.

| Refer to this book… | For… |
|---|---|
| *Getting Started with Trading Networks* | An overview of Chem eStandards Module, how to start and end Chem eStandards Module, and information to familiarize you with the Chem eStandards Module user interface.<br><br>Location #1:<br>webMethods6\TNConsole\doc\wMTN_GettingStarted.pdf<br><br>Location #2: webMethods6\IntegrationServer\packages\WmTN\doc\ wMTN_GettingStarted.pd |
| *Building Your Trading Network* | Procedures for building your trading network, including how to define information about your trading partners, add partners to your trading network, and define the processing of business documents that are sent to your trading network.<br><br>Location #1:<br>webMethods6\TNConsole\doc\ wMTN_BuildingYourNetwork.pdf<br><br>Location #2:<br>webMethods6\IntegrationServer\packages\WmTN\doc\ wMTN_BuildingYourNetwork.pdf |

| Refer to this book… | For… |
|---|---|
| *Managing and Analyzing Your Trading Networks* | Procedures for updating information on your corporation and managing partners, and for analyzing the exchange of documents in your trading network, including searching and viewing documents that have flowed through your network and viewing audit logs of events that have occurred in your trading network system. |
| | Location #1: webMethods6\TNConsole\doc\ wMTN_AnalyzingYourNetwork.pdf |
| | Location #2: webMethods6\IntegrationServer\packages\WmTN\doc\ wMTN_AnalyzingYourNetwork.pdf |
| *webMethods Trading Networks Programmer's Reference* | Descriptions of the built-in service that you can use to programmatically access the functions of Trading Networks. This book is for developers. |
| | Location #1: webMethods6\TNConsole\doc\wMTN_ProgrammersRef.pdf |
| | Location #2: webMethods6\IntegrationServer\packages\WmTN\doc\ wMTN_ProgrammersRef.pdf |
| *webMethods Trading Networks API Reference* | Descriptions of the Java classes that you can use to programmatically access the functions of Trading Networks. This reference is for developers who build services. |
| | Location: webMethods6\IntegrationServer\packages\WmTN\doc\api\index.html |

| Refer to this book… | For… |
|---|---|
| *webMethods Trading Networks Web Manager Configuration Guide* | Description of Web Manager and instructions about how to configure Web Manager. This guide is for administrators of Trading Networks that want to learn the capabilities of Web Manager. It is also for developers that want to customize Web Manager.<br><br>Location: webMethods6\IntegrationServer\packages\WmTNWeb\pub\doc\ wMTN_WebMgrGuide.pdf |
| *webMethods Trading Networks Large Document Handling* | Description of how to set up Trading Networks so it is able to process large documents.<br><br>Location #1: webMethods6\TNConsole\doc\wMTN_LargeDocHandling.pdf<br><br>Location #2: webMethods6\IntegrationServer\packages\WmTN\doc\ wMTN_LargeDocHandling.pdf |

## Integration Server and Developer Documentation

The following table lists documentation that webMethods provides with the Integration Server and Developer that you can use as references. Some documents are in PDF format and others are in HTML.

| Refer to this book… | For… |
|---|---|
| *Introduction to Integration with webMethods* | An overview of the webMethods integration platform and how you use it to integrate business processes across applications, information systems, people, and companies.<br><br>Location:<br><br>webMethods6\Developer\doc\IntroToIntegration.pdf |
| *webMethods Integration Server Administrator's Guide* | Information about using the webMethods Server Administrator to configure, monitor, and control the Integration Server. This book is for server administrators.<br><br>Location:<br><br>webMethods6\IntegrationServer\doc\ISAdministratorsGuide.pdf |

| Refer to this book… | For… |
|---|---|
| *webMethods Administrator User's Guide* | Information about using the centralized view provided by webMethods Administrator to access and administer remote servers securely on an enterprise network. This book is for server administrators.<br><br>Location:<br><br>webMethods6\IntegrationServer\packages\WmAdmin\pub\doc\ISWebmAdminUserGuide.pdf |
| *webMethods Developer User's Guide* | Information about using webMethods Developer to create and test services and client applications. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\ISDeveloperGuide.pdf |
| *webMethods Integration Server Clustering Guide* | Information about installing and using the Integration Server clustering feature. It also contains information for administrators who configure and manage an Integration Server system and for application developers who want to create services that interact directly with the Integration Server Repository.<br><br>Location:<br><br>webMethods6\IntegrationServer\doc\ISClusteringGuide.pdf |
| *webMethods Built-In Services Reference Guide* | Descriptions of services that are installed on your Integration Server. This book for is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\ISBuiltInServicesGuide.pdf |
| *DSPs and Output Templates Developer's Guide* | Information about creating Dynamic Server Pages (DSPs) and output templates. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\guides\ISDSPs&Templates.pdf |
| *webMethods Integration Server JAVA API Reference* | Descriptions of the Java classes you use to create services. This reference is for developers who build services using Java.<br><br>Location:<br><br>webMethods6\IntegrationServer\doc\api\Java\index.html |

| Refer to this book… | For… |
| --- | --- |
| *webMethods Integration Server C/C++ API Reference* | Descriptions of the webMethods C/C++ application program interface. This reference is for developers who build services or clients with C or C++.<br><br>Location:<br><br>webMethods6\IntegrationServer\doc\api\C\index.html |
| *SOAP Developer's Guide* | Information about using the Integration Server to exchange SOAP messages. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\guides\ISSoapGuide.pdf |
| *Schema Reference Guide* | Descriptions of the components of an IS schema and how the components relate to the XML Schema or Document Type Definition (DTD) from which it is generated. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\guides\ISSchemaGuide.pdf |
| *Flat File Schema Developer's Guide* | Information about using flat file schemas and dictionaries to parse and validate inbound flat files and produce outbound flat files. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\plugins\FlatFilePlugin\doc\ISFlatFileSchemaGuide.pdf |
| *MIME-S/MIME Developer's Guide* | Information about using the Integration Server to construct MIME and S/MIME messages, secure them, transport them over the Internet, and extract information from them. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\guides\ISMimeSmimeGuide.pdf |
| *Guaranteed Delivery Developer's Guide* | Information about using the guaranteed delivery features in the Integration Server to ensure guaranteed, one-time execution of services. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\guides\GuaranteedDeliveryGuide.pdf |

| Refer to this book… | For… |
|---|---|
| *Web Services Developer's Guide* | Information about using the Integration Server to create Web services and incorporate Web services into the solutions you develop. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\guides\ISWebServicesGuide.pdf |
| *Working with XML Documents* | Information about using the Integration Server to send, receive, and query XML documents. This book is for developers.<br><br>Location:<br><br>webMethods6\Developer\doc\guides\ISXmlGuide.pdf |
| *webMethods Certificate Toolkit User's Guide* | Information about installing and using the webMethods Certificate Toolkit. It also contains information for administrators and developers of webMethods components about creating and managing digital certificates for use with webMethods components.<br><br>Location:<br><br>webMethods6\CertToolkit\doc\ISCertToolkitGuide.pdf |
| *webMethods Integration Server Administrator's Online Help* | Information about the controls in the Server Administrator screens and step-by-step procedures describing how to perform tasks with the Server Administrator.<br><br>You can access the online reference by clicking the **Help** link on a a Server Administrator screen. |
| *webMethods Developer Online Help* | Information about the controls in the Developer application windows and step-by-step procedures describing how to perform tasks with the Developer.<br><br>You can access the online reference by clicking **Help** in an application window or dialog box. |

## Modeler Documentation

The following table lists the manuals that webMethods provides with Modeler.

| Refer to this book… | For… |
|---|---|
| *webMethods Modeler User's Guide* | Information about how to use Modeler to build, generate, and deploy business process models.<br><br>Location:<br>webMethods\Modeler\doc\ModelerUsersGuide.pdf |

## Monitor Documentation

The following table lists the manuals that webMethods provides with Monitor.

| Refer to this book… | For… |
| --- | --- |
| *webMethods Monitor User's Guide* | Information about monitoring business process models, services, and documents.<br><br>Location:<br>webMethods\\*ServerDirectory*\\packages\\WmMonitor\\MonitorUsersGuide.pdf |

## Chem eStandards Documentation

The following table lists documentation that describes the Chem eStandards.

| Refer to this book… | For… |
| --- | --- |
| *webMethods Chem eStandards Phase II* | Provides information about Chemical eStandards, also called ChemXML. Contains specifications for each of the 47 business processes supported by CIDX Chem eStandards.<br><br>Location: http://www.CIDX.org<br><br>This document is downloadable. |

# Viewing this Document

To view this document, which is in PDF format, you must have Acrobat Reader™ 4.0 or later installed on your system. If you have an earlier version of Acrobat Reader, you will receive the following error message when you open this document and Acrobat Reader will not display the images in this document:

```
Could not find the ColorSpace named 'Cs8.'
```

If you do not have this software or you do not have the correct version, you can download a free copy from:

http://www.adobe.com/supportservice/custsupport/download.html.

# Printing this Guide

To produce a hard copy of this guide, print this document from Acrobat Reader.

# Concepts

# What Is the webMethods Chem eStandards Module?

The webMethods Chem eStandards Module is an implementation of the Chem eStandards that use the Transport, Envelope, and Security features of the RosettaNet Implementation Framework (RNIF) version 1.1. Leveraging the webMethods platform process modeling and management capabilities and XML business document (in this guide called business document) recognition capabilities, the webMethods Chem eStandards Module enables you to implement open and common e-business Chem eStandards processes and their implementation guidelines. The webMethods Chem eStandards Module consists of a set of design-time and run-time components, both sets of which are discussed in this chapter. For information about the webMethods Chem eStandards Module design-time components, see "webMethods Chem eStandards Module Design-Time Architecture/Components" on page 24 in this guide. For information about the webMethods Chem eStandards Module run-time components, see "webMethods Chem eStandards Module Run-Time Architecture/Components" on page 26 in this guide.

You implement a Chem eStandards process by importing the necessary Chem eStandards process archive, creating your TN XML document types and Trading Partner Agreements (TPAs), and then customizing one of four process model templates as described in Chapter 9, "Customizing a Process Model Template" in this guide. To customize a process model template, you assign services that you create to specific steps in the template. You also assign TN XML document types to steps that wait for business documents (in this guide called wait steps), so that when the webMethods Chem eStandards Module receives a business document during a conversation, or process, it can recognize the type of business document received. For more information about process models and webMethods Modeler, the tool used to customize the process model templates, see the *webMethods Modeler User's Guide*. For more information about Chem eStandards process archives, see Chapter 5, "Importing Chem eStandards Process Archives" in this guide. For the typical procedure for implementing a Chem eStandards process, see Chapter 3, "Getting Started" in this guide.

The process run time (PRT), a facility of the webMethods Integration Server, manages the execution of Chem eStandards conversations. The PRT is the run-time component that executes process logic, logs process data, and controls process execution order. During run time, the PRT uses the ConversationID associated with a business process to determine if the process is active, or currently running. For more information about the PRT, see "webMethods Chem eStandards Module Run-Time Architecture/Components" on page 26 in this guide and the *webMethods Modeler User's Guide*. For more information about ConversationIDs, see "Running a Conversation and the ConversationID" on page 88 in this guide.

Each trading partner involved in a conversation must be associated with a TPA, either an initiator TPA or a fulfiller TPA, depending on the role that the trading partner plays in the conversation. A TPA is a set of parameters that you can use to govern how documents are exchanged between two trading partners. For example, a TPA for a Chem eStandards conversation contains parameters that control such things as signing, encoding, and validation of Chem eStandards records. For more information about TPAs, see Chapter 7,

"Defining Trading Partner Agreements" in this guide and the *Building Your Trading Network* manual.

# webMethods Chem eStandards Module Features

The webMethods Chem eStandards Module runs on top of the Integration Server. The webMethods Chem eStandards Module provides out-of-the-box capability to validate a number of the Chem eStandards processes.

By using the webMethods Chem eStandards Module, you can:

■ **Provide out-of-the-box support for Chem eStandards processes**, which enables you to quickly implement production solutions for automating the many interactions across your entire supply chain.

■ **Create webMethods Chem eStandards-based process models.** Use Modeler to create business process models that reflect your enterprise's business requirements.

■ **Monitor your process models.** Use webMethods Monitor to manage and monitor your business process models.

■ **Incorporate new versions of Chem eStandards processes** quickly after release from CIDX, without waiting for a new release of software.

■ **Capture trading partner-specific rules**, reflecting the unique business practices between your organization and your trading partners.

■ **Leverage existing investments in enterprise solutions** by accepting information from EDI-based systems to populate business documents in Chem eStandards format.

■ **Support the widest range of eStandards**, including cXML, CBL, OAG, FpML, BizTalk, Acord, EDI, and OBI, which gives you the ability to use a preferred approach in-house while supporting the different standards requirements of your customers. In addition, this allows you to use the webMethods Chem eStandards Module as your platform for all Chem eStandards interactions across the Internet.

■ **Maintain transaction logging and audit trails** to ensure the integrity of all trading partner transactions. Automatic archival of transaction messages as well as digital signature support ensures non-repudiation of origin and content.

■ **Support synchronous and asynchronous transmission of Chem eStandards records.** With the webMethods Chem eStandards Module, you can employ either synchronous or asynchronous communications to better meet the time requirements of your trading partner transactions. For those Chem eStandards processes that require immediate responses over the same Internet connections, you can use synchronous communications. For processes with longer duration, you can use asynchronous communications.

■ **Support multi-byte languages,** such as Chinese and Japanese.

# webMethods Chem eStandards Module Packages

The webMethods Chem eStandards Module contains several packages (sets of services and related files) that you install on the Integration Server. The following table lists and describes these packages. For detailed information about the contents of a package, see the *webMethods Chem eStandards Module Built-In Services Reference Guide*.

| Package | Description |
|---------|-------------|
| WmIPRoot | Contains the shared framework components, including implementations of common utilities and abstractions and base classes for frameworks. For example, shared logging facilities are included in this package as an abstract transport framework. Concrete and application-specific implementations are contained in the other packages. |
| WmChem | Contains general application functionality and serves as the main holding area for application user interfaces, typically DSPs, that are not provided by the Trading Networks Console. This package implements: <br><br> ■ the import facility, which allows new webMethods Chem eStandards Module IS document types to be added to the solution dynamically. <br><br> ■ the export facility, which allows you to export available webMethods Chem eStandards Module IS document types, TN XML document types, and TPAs to process archive files. |
| WmRNIF11TRP | Implements the RosettaNet Implementation Framework (RNIF) version 1.1 transport, envelope, and security features, which provide the basic transport mechanisms that are used in the webMethods Chem eStandards Module and Trading Networks for sending and receiving Chem eStandards records. |

| Package | Description |
|---|---|
| WmChemSample | Provides a scenario in which an initiator/buyer and fulfiller/seller exchange business documents, specifically where an initiator sends a purchase order (OrderCreate) to a fulfiller, and, in return, the fulfiller sends a purchase order response (OrderResponse) to the buyer. |
| | This package includes the following: |
| | ■ A non-production buyer script |
| | ■ A non-production seller script |
| | ■ Non-production versions of the necessary TN XML document types for the OrderCreate and OrderResponse business documents |
| WmChemPayloads | Holds the IS document types corresponding to the webMethods Chem eStandards Module messages. These IS document types contain constraint data specified by the standard built into them. This package is not installed when you install the webMethods Chem eStandards Module. However, upon importing a webMethods Chem eStandards Module process archive, the package is dynamically created and populated with the webMethods Chem eStandards Module IS document types found in the process archive file. |

# webMethods Chem eStandards Module Design-Time Architecture/Components

The following figure shows the webMethods Chem eStandards Module design-time architecture and components, and the component relationships. For further explanation, see the table that follows the figure.

**webMethods Chem eStandards Module Design-Time Architecture/Components**
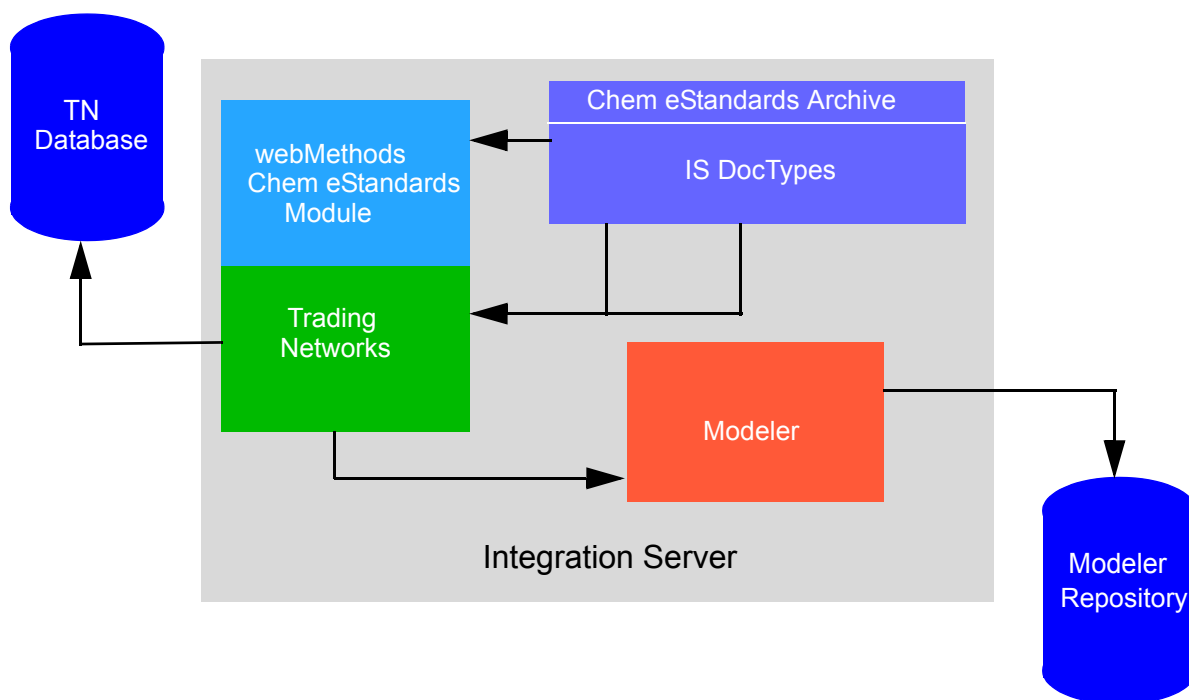


| Component | Description |
|---|---|
| Chem eStandards Archive | A Chem eStandards archive is a compressed file, called a process archive (.par) that contains an IS document type for a specific version of a Chem eStandards process. During design time, you import Chem eStandards archives for the Chem eStandards processes you need into the Integration Server. |
| webMethods Chem eStandards Module | The webMethods Chem eStandards Module contains generic process model templates for the different kinds of Chem eStandards processes you can implement. You can use these process model templates to define your Chem eStandards processes from within Modeler. |

| Component | Description |
|---|---|
| webMethods Trading Networks | Trading Networks enables your enterprise to link with trading partners with whom you want to exchange business documents, thereby forming a business-to-business trading network. During design time, you define your trading partner profiles in the Trading Networks Console. The profiles contain the information that Trading Networks needs to exchange business documents with your trading partners.<br><br>In addition to defining trading partner profiles during design time, you also create TN XML document types and Trading Partner Agreements (TPAs) in the Trading Networks Console.<br><br>For more information about Trading Networks, trading partner profiles, TN XML document types, and TPAs, see the *Building Your Trading Network* manual. You can also find information about trading partner profiles in Chapter 4, "Defining Trading Partner Profiles in Trading Networks" in this guide and information about TPAs in Chapter 7, "Defining Trading Partner Agreements" in this guide. |
| Trading Networks Database | Trading Networks saves trading partner profile, TN XML document type, and TPA information, among other things, to its database and retrieves this information when needed. |
| Modeler | Modeler is a Java GUI. You use Modeler to customize the process model templates provided by webMethods, thereby creating your own process models. You customize a process model template by specifying how the process model is to interact with your back-end systems and editing the services that are invoked by the steps of the process model, among other things. When you generate a process model, you generate the run-time elements (services, triggers, and process run-time scripts, or fragments) that execute at run time. For more information about Modeler, see the *webMethods Modeler User's Guide*. For information about customizing a process model template, see Chapter 9, "Customizing a Process Model Template" in this guide. |
| Modeler Repository | The Modeler Repository is a storage area that Modeler uses to save process model information. For more information about Modeler, see the *webMethods Modeler User's Guide*. |
| Integration Server | The Integration Server contains the documents, services, and records that you will want to access when creating your process models. |

# webMethods Chem eStandards Module Run-Time Architecture/Components

The following figure shows the webMethods Chem eStandards Module run-time architecture and components, and the component relationships. For further explanation, see the table that follows the figure.

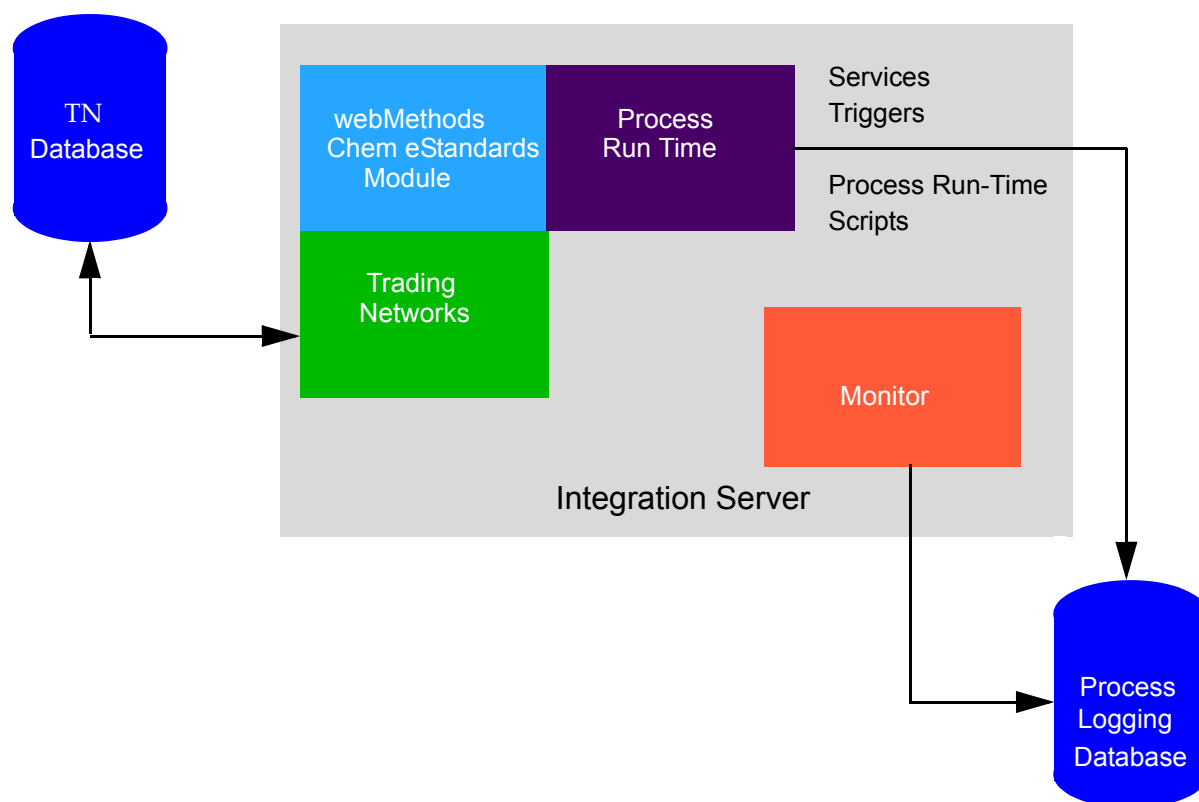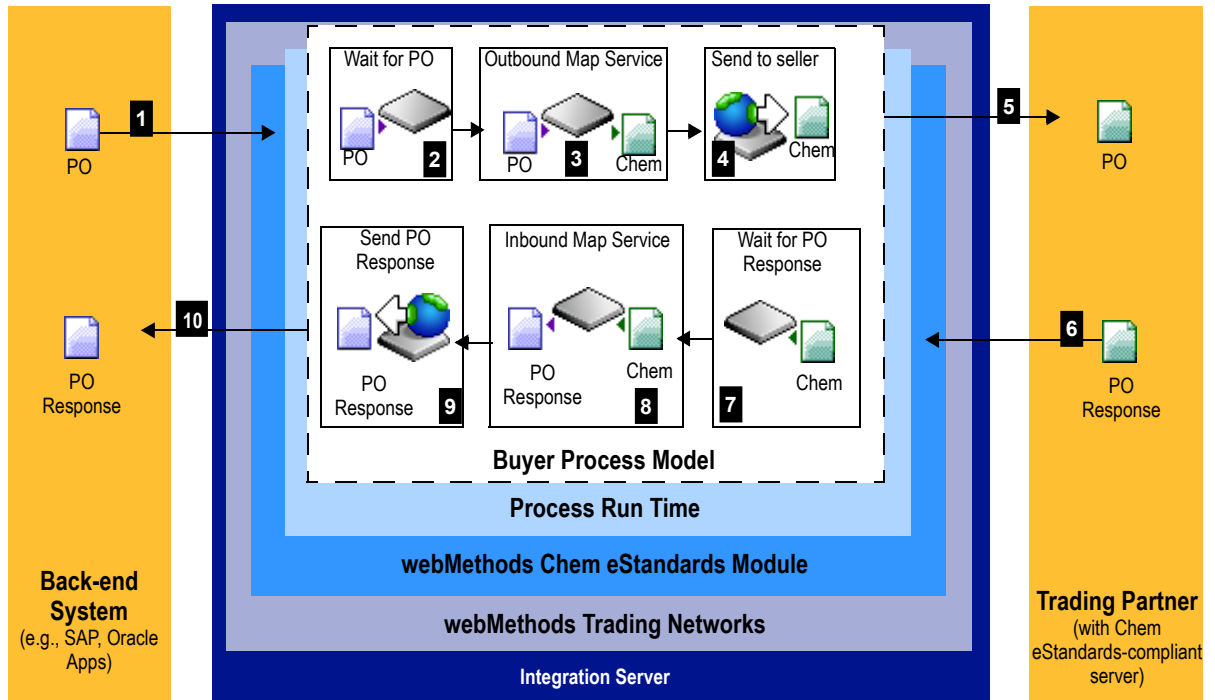**webMethods Chem eStandards Module Run-Time Architecture/Components**

| Component | Description |
|---|---|
| webMethods Chem eStandards Module | During run time, the webMethods Chem eStandards Module receives a business document from a back-end system or trading partner. It invokes a Trading Networks service to recognize the business document, create a BizDocEnvelope and ConversationID, and save the BizDocEnvelope to the Trading Networks database. The webMethods Chem eStandards Module passes the business document to the PRT. For more information about the PRT, see the PRT row in this table and the *webMethods Modeler User's Guide*. |
| Trading Networks | Trading Networks enables your enterprise to link with trading partners with whom you want to exchange business documents, thereby forming a business-to-business trading network.<br><br>During run time, the webMethods Chem eStandards Module uses Trading Networks services and TN XML document types to recognize documents it receives, create BizDocEnvelopes, and save BizDocEnvelopes to the Trading Networks database. The webMethods Chem eStandards Module uses the trading partner profiles in Trading Networks to know, for example, the methods by which to send business documents to its trading partners. The webMethods Chem eStandards Module uses TPAs in Trading Networks to know information such as whether the outbound business document for a process should be signed, whether the inbound business document for a process should contain your trading partner's signature information, and so on.<br><br>For more information about Trading Networks, trading partner profiles, TN XML document types, and TPAs, see the *Building Your Trading Network* manual. You can also find information about trading partner profiles in Chapter 4, "Defining Trading Partner Profiles in Trading Networks" in this guide and information about TPAs in Chapter 7, "Defining Trading Partner Agreements" in this guide. |
| Trading Networks Database | The Trading Networks database stores TN XML document type, TPA, and trading partner profile information, among other things. |

| Component | Description |
|---|---|
| PRT | The process run time (PRT) is a facility of the Integration Server that manages the execution of Chem eStandards conversations. The PRT ensures the integrity, traceability, observability, and controllability of Chem eStandards conversations by performing the following functions:<br><br>■ Accepts business documents from Trading Networks.<br><br>■ Determines what process model to use for a given Chem eStandards process.<br><br>■ Processes a business document, based on the type of business document received and who sent it.<br><br>■ Records the status of the business document to the Process Logging Database.<br><br>During processing of a business document, the PRT uses the ConversationID associated with the business document to see if the document belongs to a new or existing conversation. The PRT looks for a matching ConversationID among the enabled process models. If it does not find a matching ConversationID, it starts a new conversation (process). If the PRT finds an enabled process model with a matching ConversationID, the business document rejoins the conversation.<br><br>For more information about the PRT, see the *webMethods Modeler User's Guide*. For more information about ConversationIDs, see "Running a Conversation and the ConversationID" on page 88 in this guide. |
| Monitor | You use Monitor to manage and monitor conversations. Monitor displays information about a conversation by retrieving information from the Process Logging Database.<br><br>Using Monitor, you can manage a conversation by performing such commands as suspend, resume, restart, and terminate. |
| Process Logging Database | Monitor and the PRT log audit data about running conversations to the Process Logging Database. |
| Integration Server | The Integration Server contains the run-time elements that are generated from the automated controlled steps within the process model. The run-time elements are services, triggers, and process run-time scripts, or fragments. |

# Process Overview

The following figure illustrates the use of the Chem eStandards *OrderCreate* process to send a purchase order (PO) to a trading partner using the webMethods Chem eStandards Module. Only the conversation for the buyer role is displayed. For further explanation, see the table that follows the figure.

**Buyer Process Overview of the Chem eStandards *OrderCreate* Process**

| Step | Description |
|------|-------------|
| **1** | Your enterprise (acting as the initiator/ buyer) submits an internal PO from your back-end system to the webMethods Chem eStandards Module. Upon receiving the internal PO, the webMethods Chem eStandards Module invokes a Trading Networks service to recognize the TN XML document type. Trading Networks wraps the PO in a BizDocEnvelope, assigns the PO a ConversationID, and saves the BizDocEnvelope to the Trading Networks database. |
| **2** | The webMethods Chem eStandards Module passes the BizDocEnvelope to the PRT. The PRT looks for a matching ConversationID among the enabled process models. If the PRT does not find a matching ConversationID, it starts the conversation (process). At the **Wait for PO** step, a service retrieves the Trading Networks profiles for the sender and receiver and the TPAs. |
| **3** | An outbound mapping service maps the internal PO into a Chem eStandards record. |
| **4** | At the **Send to Seller** step, a service sends the PO (as a Chem eStandards record) to the trading partner (the fulfiller/seller). |
| **5** | The trading partner receives the PO, validates it, and sends a Receipt Acknowledgment (not illustrated). |
| **6** | The trading partner (Seller) sends a PO Response in the form of a Chem eStandards record to your enterprise by way of the webMethods Chem eStandards Module. The webMethods Chem eStandards Module receives the PO Response and extracts the ConversationID. The webMethods Chem eStandards Module passes the PO Response to the PRT. |
| **7** | The PRT matches the ConversationID from the PO Response with the ConversationID associated with a specific process model. The conversation resumes, validation of the PO Response occurs, and your enterprise sends an Acceptance Acknowledgment (not illustrated) to the trading partner. |
| **8** | An inbound mapping service maps the PO Response (as a Chem eStandards record) to an internal format recognizable by your back-end system. |
| **9** | At the **Send PO Response** step, a service sends the PO Response to your back-end system. |
| **10** | Your back-end system receives the PO Response. |

# Getting Started

# Typical Procedure for Implementing a Chem eStandards Process

The following procedure outlines the steps you take to implement a Chem eStandards process.

**Important!** The following procedure assumes that you already have installed the webMethods Integration Server, webMethods Trading Networks, webMethods Modeler, and the webMethods Chem eStandards Module.

## Step 1: Install the webMethods Chem eStandards Module Packages

You install the webMethods Chem eStandards Module packages using the webMethods installer. The packages you install are:

- WmIPRoot

- WmChem

- WmRNIF11TRP

- WmChemSample

For information about installing the webMethods Chem eStandards Module packages, see the *webMethods Chem eStandards Module Installation Guide*.

## Step 2: Define Trading Partner Profiles

In the webMethods Trading Networks Console, you define the profiles for yourself and all trading partners with whom you want to exchange business documents. A trading partner profile includes parameters such as the D-U-N-S® Number and server URL. For more information about defining trading partner profiles, see Chapter 4, "Defining Trading Partner Profiles in Trading Networks" in this guide and the *Building Your Trading Network* manual.

## Step 3: Define TN XML Document Types

You define internal TN XML document types for each of the back-end business documents you want to use in your Chem eStandards processes and external TN XML document types for each of the Chem eStandards records you exchange with your trading partners. For information about defining TN XML document types, see Chapter 6, "Defining TN XML Document Types" in this guide and the *Building Your Trading Network* manual.

### Step 4: Import the Chem eStandards Process Archives You Are Going To Use

You download the compressed Chem eStandards process archives that represent the various versions of the Chem eStandards processes that you want to implement. You install these archives on the Integration Server using the import functionality in the webMethods Chem eStandards Module. For more information about importing process archives, see Chapter 5, "Importing Chem eStandards Process Archives" in this guide.

### Step 5: Define Your Trading Partner Agreements

You define your trading partner agreements (TPAs), if necessary. A TPA is a set of parameters that you can use to govern how documents are exchanged between two trading partners. You define a TPA in the Trading Networks Console. Each participant in a Chem eStandards conversation must define a TPA. For more information about TPAs, see Chapter 7, "Defining Trading Partner Agreements" in this guide and the *Building Your Trading Network* manual.

### Step 6: Write Outbound and Inbound Mapping Services

You create mapping services, which consist of two types: outbound and inbound. You create an outbound mapping service to map an internal business document received from the back-end to a Chem eStandards record that you want to send to a trading partner. You create an inbound mapping service to map a Chem eStandards record received from a trading partner to an internal business document that is then sent to a back-end system. For more information about mapping business documents, see Chapter 8, "Mapping a Chem eStandards Record" in this guide.

### Step 7: Write Error Handler Services

You write error handler services, if necessary. The webMethods Chem eStandards Module provides handler services for various tasks, such as handling a Chem eStandards record that does not conform to any TN XML document type. You can customize some of these handler services or create your own. For more information about error handling services, see Chapter 11, "Handling Errors and Exceptions" in this guide.

### Step 8: Customize the Process Model Template You Are Going To Use

You determine the webMethods-provided process model template you need to use and then customize the template based on the Chem eStandards process you are implementing. Customizing a process model template consists of such tasks as specifying the service that a step invokes, assigning a TN XML document type to a wait step, and so on. For more information about customizing a process model template, see Chapter 9,

"Customizing a Process Model Template" in this guide. For more information about process models in general, see the *webMethods Modeler User's Guide*.

## Step 9: Generate and Update Your Process Model

You generate and update your process model in Modeler. When you generate a process model, Modeler generates the run-time elements (that is, services, triggers, and process run-time scripts, or fragments) that actually execute at run-time. For information about generating and updating your process model, see the *webMethods Modeler User's Guide*.

## Step 10: Start and Run the Conversation

To start a conversation on the initiator's side, the initiator's back-end system calls wm.ip.cm:process Document. To start a conversation on the fulfiller's side, a service on the initiator's side sends a business document (as a Chem eStandards record) to wm.ip.chem:receive. After you start a service, you can monitor its progress using webMethods Monitor.

For more information about wm.ip.cm:process Document and wm.ip.chem:receive, see the *webMethods Chem eStandards Module Built-In Services Reference Guide*. For more information about starting a conversation, see "Starting a Conversation" on page 90 in this guide. For more information about Monitor, see "About Monitor" on page 92 in this guide, and the *webMethods Monitor User's Guide*.

# Defining Trading Partner Profiles in Trading Networks

# What Is a Trading Partner?

A *trading partner* is any person or organization with whom you want to conduct business electronically. In the webMethods Chem eStandards Module, a trading partner is defined by several criteria that you specify in a trading partner profile, including company name and identifying information, contact information, and preferred delivery methods.

In addition to specifying trading partner profiles for all of your trading partners, you must specify a profile for your own organization.

# Why Are Trading Partner Profiles Important?

Your trading partner profiles, used in conjunction with trading partner agreements (TPAs) and process models, define how the various trading partners transact business with you. The different roles, for example, initiator/buyer or fulfiller/seller, in a process model define what actions your company can take in certain transactions, as well as the actions you expect your trading partners to perform during those transactions. In fact, the concise definition of profiles, the customization of process model templates, the application of TPAs, and the implementation of Chem eStandards processes are what enables you to successfully interact with your trading partners.

You are likely to need to act in different roles with different trading partners. For example, you might need to act as a buyer toward your suppliers and as a seller toward your distributors. With the webMethods Chem eStandards Module, you define a single trading partner profile for yourself (**My Enterprise**) and then customize the pertinent process model templates to define each role you will perform in conducting transactions with your trading partners. You also must define a trading partner profile for each trading partner with whom you will conduct transactions, and again customize the pertinent process model templates for each role those partners might play.

A trading partner's role figures prominently when you customize the process model templates for Chem eStandards processes. When you customize a process model template, you will use the various trading partner profiles to help define the sender of the business document, the receiver of the business document, target server's URL, and other criteria that depend heavily on the role of the trading partner. For information about customizing process model templates, see Chapter 9, "Customizing a Process Model Template" in this guide. For information about process models in general, see the *webMethods Modeler User's Guide*.

# Defining Your Enterprise Profile Using the Trading Networks Console

Before defining your trading partner profiles in Trading Networks and exchanging business documents with your trading partners, you must first define your enterprise (**My Enterprise**) profile. You define your enterprise profile by completing the fields in the Profile Assistant in the Trading Networks Console. For procedural information about defining your enterprise profile as well as descriptions of the fields you must complete when defining your enterprise profile, see the *Building Your Trading Network* manual. For an example of how to complete your enterprise profile, see the *webMethods Chem eStandards Module Sample Guide*.

The following sections specify the required fields you must complete to define your enterprise profile in the Trading Networks Console and the Chem eStandards-specific notes on other information you must complete, such as items you should know when completing delivery method information.

## Required Profile Fields

Profile information is displayed on the **Corporate** tab. The following table lists and describes the required profile fields you must complete when defining your enterprise profile.

| Required Profile Field for My Enterprise | Description |
| --- | --- |
| Corporation Name | The name of your enterprise. |
| External ID Type Value | Your enterprise's D-U-N-S$^{®}$ Number. |

For descriptions of other fields you can complete when you define your enterprise profile, see the *Building Your Trading Network* manual.

## Delivery Method Information

Delivery method information is displayed on the **Delivery Method** tab. To complete information on the **Delivery Method** tab, you need to know the following:

■   You must specify at least one delivery method as your preferred method by selecting the **Use as preferred protocol** check box. If you do not specify a preferred delivery method in Trading Networks, the webMethods Chem eStandards Module will not be able to determine how to send and receive Chem eStandards records and therefore will not be able to execute any conversations.

■   If you select one of the HTTP protocols as the delivery method, you must specify invoke/wm.ip.chem/receive as the location.

## Security Information

Security information is displayed on the **Security** tab. The webMethods Chem eStandards Module uses the same certificate information for signing and decrypting.

## Activating Your Enterprise Profile

After you define your enterprise profile, you activate it by clicking **Enable**  💡  .

# Defining Your Trading Partners' Profiles

Each trading partner with whom you want to exchange business documents must have a trading partner profile in Trading Networks. After you have defined your enterprise profile, you are ready to define your trading partners' profiles.

You define a trading partner profile by completing the fields in the Profile Assistant in the Trading Networks Console. For procedural information about defining a trading partner profile as well as descriptions of the fields you must complete when defining a trading partner profile, see the *Building Your Trading Network* manual.

The following sections specify the required fields you must complete to define a trading partner profile and the Chem eStandards-specific notes on other information you must complete, such as items you should know when completing delivery method information.

## Required Profile Fields for Your Trading Partner

Profile information is displayed on the **Corporate** tab. The following table lists and describes the required fields you must complete when defining a trading partner profile.

| Required Profile Field for Trading Partner | Description |
| --- | --- |
| **Corporation Name** | The name of the trading partner. |
| **Partner Type** | The type of partner. You should select **webMethods Trading Networks**. |
| **External ID Type Value** | Your trading partner's D-U-N-S Number. |

For descriptions of other fields you can complete when you define a trading partner profile, see the *Building Your Trading Network* manual.

## Delivery Method Information

Delivery method information is displayed on the **Delivery Method** tab. To complete information on the **Delivery Method** tab, you need to know the following:

■ You must specify at least one delivery method as your preferred method by selecting the **Use as preferred protocol** check box. If you do not specify a preferred delivery method, the webMethods Chem eStandards Module will not be able to determine how to send and receive business documents and therefore will not be able to execute any conversations.

■ If HTTP is the preferred protocol, then specify the URL provided by your trading partner to receive Chem eStandards records sent by you.

■ If SMTP is the preferred protocol, then enter the e-mail address provided by your trading partner to receive Chem eStandards records sent by you.

## Security Information

The Profile Assistant for defining a trading partner profile contains four security information screens. In these screens, you specify the security information you want to use when sending business documents to your trading partner.

The certificate names you enter in the fields on these screens are used for signing Chem eStandards records you send to your trading partner and for decrypting encrypted Chem eStandards records you receive from your trading partner. If you do not provide any certificate information, then these business documents cannot be signed, encrypted, or decrypted.

Security information is displayed on the **Security** tab.

**Note:** The webMethods Chem eStandards Module uses the same certificate information for signing and decrypting.

## Activating Your Trading Partners' Profiles

After you define a trading partner profile, you activate it by clicking Enable     .

# Importing Chem eStandards Process Archives

# What Is a Chem eStandards Process Archive?

A *Chem eStandards process archive* is a compressed file (.par) that contains the IS document types for a specific version of a specific Chem eStandards process.

> **Note:** IS document types define the namespace for the Chem eStandards process and also define the structure of the Chem eStandards records that are passed back and forth during a Chem eStandards conversation.

Chem eStandards process archives reside on the webMethods software download site at the URL provided to you by webMethods. Each Chem eStandards process archive has a file name that reflects the Chem eStandards process name and version. For example, the file name for the Chem eStandards *OrderCreate* process, version 2.0, is OrderCreatev20.par.

You must create the Trading Partner Agreements (TPAs) and TN XML document types that complete the implementation of the Chem eStandards processes. However, the Chem eStandards Module sample provides sample process archives that contain IS document types *as well as* TPAs and TN XML document types. You can use the TPAs and TN XML document types in these sample process archives as a guideline for building your own TPAs and TN XML document types.

For information about TN XML document types, see Chapter 6, "Defining TN XML Document Types" in this guide and the *Building Your Trading Network* manual. For information about TPAs, see Chapter 7, "Defining Trading Partner Agreements" in this guide and the *Building Your Trading Network* manual.

# Why Do I Need To Import a Chem eStandards Process Archive?

You need to import a Chem eStandards process archive to implement a Chem eStandards conversation in the webMethods Chem eStandards Module. Because you download Chem eStandards process archives from the webMethods software download site, you can add new Chem eStandards process archives to your webMethods Chem eStandards Module installation as they are released by webMethods, without having to upgrade your installation.

You can implement different versions of the same Chem eStandards process to support transactions between you and your trading partners when not all of your trading partners are using the same version of the same process. However, in any specific Chem eStandards conversation, you and your trading partner must be using the same version of the Chem eStandards process.

You cannot copy over the Chem eStandards process archives for previous releases of the webMethods Chem eStandards Module from one upgrade to the next. If you do not have customized Chem eStandards processes and are upgrading to release 6.0.1, you should download and import the process archive. For customized Chem eStandards processes, release 6.0.1 upgrade functionality will be added at a later date.

# Chem eStandards Processes and Process Models

Each Chem eStandards process archive enables you to implement a Chem eStandards process. Each of these processes can be described as following one of a few very basic process models. The differences between the specific Chem eStandards processes are mostly in the business documents that the webMethods Chem eStandards Module passes back and forth and how the Module handles those business documents.

In Chapter 2, "Concepts", the basic overview of Chem eStandards processes was presented using the example of the Chem eStandards *OrderCreate* process. This process is an example of a two-action asynchronous process model. In general, Chem eStandards processes follow one of the three types of process models:

■ **One-action asynchronous.** The initiating trading partner sends an asynchronous request, and the receiving trading partner sends a receipt acknowledgment. The webMethods Chem eStandards Module contains two process model templates for the one-action asynchronous model: one template for the initiator/sender and one template for the fulfiller/receiver. For information about these process model templates, see Chapter 9, "Customizing a Process Model Template" in this guide.

■ **Two-action asynchronous.** The initiating trading partner sends an asynchronous request, and the receiving trading partner sends a receipt acknowledgment. Next, the receiving partner sends an asynchronous response document, and the initiating partner sends a receipt acknowledgment. The webMethods Chem eStandards Module contains two process model templates for the two-action asynchronous model: one template for the initiator and one template for the fulfiller. For information about these process model templates, see Chapter 9, "Customizing a Process Model Template" in this guide.

■ **Two-action synchronous.** The initiating trading partner sends a synchronous request, and the receiving trading partner does not need to send a receipt acknowledgment. Next, the receiving partner sends a synchronous response document, and the initiating partner does not need to send a receipt acknowledgment. This model is currently not used in the Chem eStandards environment.

# Importing Chem eStandards Process Archives

All Chem eStandards process archives are located on the webMethods software download site at the URL provided to you by webMethods. You import a Chem eStandards process archive by using the Chem eStandards import facility in the webMethods Server Administrator.

**To import a Chem eStandards process archive**

1   Navigate via a Web browser to the webMethods software download site at the URL provided to you by webMethods.

2   Enter your user name and password.

3   Scroll to the **webMethods Chem eStandards Module** section on the page.

4   Click the Chem eStandards process archive that you want to download. For example, if you want to download the *OrderCreate*, version 2.0, process archive, click **OrderCreatev20.par**.

5   Save the file to the *import* subdirectory of your Chem eStandards Module installation, for example:

\webMethods6\IntegrationServer\packages\WmChem\import

where \webMethods6\IntegrationServer is the directory in which your webMethods Integration Server is installed.

6   After saving the file to your file system, start the Integration Server and Server Administrator, if they are not already running.

> **Note:** If you need procedures for this step, see the *webMethods Integration Server Administrator's Guide*.

7   On the Server Administrator navigation panel, under the **Adapters** heading, click **Chem**.

8   On the Server Administrator navigation panel, under the **Chem** heading, click **Import** to display the following page.

**Transaction Import Files Page**

Process archive files that you saved in the *import* subdirectory of your webMethods Chem eStandards Module installation are displayed here.

**Transaction Import Files**

☐ OrderCreatev20.par

Please select the files for which you would like to see the contents and then press the 'View' button:

[View]

9   Select the checkboxes next to the Chem eStandards process archives you want to import.

**10** Click **View**.

> **Important!** If an exclamation point appears to the left of the entries in the **Archive Contents** section of the **Transaction Import File Details** page, you have already installed this version of this Chem eStandards process archive. If you want to use the version of the process archive that you are importing, continue this procedure.
>
> If you have custom changes to the Chem eStandards process IS document types that you want to preserve, **stop** this procedure and follow the procedure in "Exporting a Chem eStandards Process Archive" on page 46 in this guide. After exporting the customized Chem eStandards process, continue with the next step in this procedure.

**11** Scroll to the bottom of the page, and select **Import Archives.**

# Customizing Chem eStandards Process Archives

When implementing your webMethods Chem eStandards Module installation, you might make numerous changes to the process archives that you import. For example, you might make changes to the IS document types against which you validate Chem eStandards records. The next two sections discuss two different approaches for customizing a process archive.

> **Note:** For the advantages and disadvantages of customizing and exporting a Chem eStandards process, see "Advantages and Disadvantages of Customizing and Exporting Chem eStandards Process Archives" on page 49 in this guide.

## Customizing IS Document Types for Existing Standard Chem eStandards Process Archives

There are two ways to customize an IS document type for an existing standard Chem eStandards process archive:

■ If your trading partners are unable to populate and/or have no need for certain required fields in a Chem eStandards record, you can change the constraints in those fields and make them optional.

**Advantage:** This type of customization enables your Chem eStandards record for that Chem eStandards process to pass validation.

**Disadvantage:** This type of customization affects every trading partner that will be exchanging this record with you.

■ If you intend to isolate changes to an IS document type for a specific trading partner, follow these steps:

**a** Create another folder in the WmChemPayloads package under wm.b2b.chem.rec.payloads and name it *<your customized record structure>*.

**b** Enter the new location as the value for the *NSFolder* parameter in the TPA for that Chem eStandards process and the specific trading partner.

**Advantage:** This type of customization localizes changes for one trading partner and for one particular Chem eStandards process only.

**Disadvantage:** This type of customization provides you with one additional IS document type to maintain.

## Implementing a Custom Chem eStandards Process

If you would like to implement a custom Chem eStandards process other than the standard Chem eStandards processes published by the Chemical Industry Data Exchange (CIDX™), follow the steps below.

**To implement a custom Chem eStandards process**

**1** In the Trading Networks Console, create a new TN XML document type and specify a root tag for the TN XML document type that matches the root tag of the customized Chem eStandards record.

**2** Add a pipeline matching variable to this new TN XML document type called *processVersion* and specify a value for this variable that matches the version in the Service Header of the customized Chem eStandards record.

**3** In the Server Administrator, create a new IS document type for the customized Chem eStandards record.

**4** In the Trading Networks Console, create a new Trading Partner Agreement (TPA). Specify the new TN XML document type name as the Agreement ID for the TPA, and specify the new IS document type as the value for the *NSFolder* parameter in the TPA.

**5** In webMethods Modeler, in the process model template for the Chem eStandards process you are implementing, specify the new TN XML document type as input to the appropriate wait step.

# Exporting a Chem eStandards Process Archive

When you create TPAs or TN XML document types, or modify or create IS document types in a process archive, you should package your changes together into an exported process archive file that you can then manage as a production archive, complete with version control, and can share with your trading partners.

You have the option of exporting any TPAs, TN XML document types, and IS document types located in the Integration Server.

**To export a Chem eStandards process archive**

1　Start the Integration Server and Server Administrator, if they are not already running.

> **Note:** If you need procedures for this step, see the *webMethods Integration Server Administrator's Guide*.

2　On the Server Administrator navigation panel, under the **Adapters** heading, click **Chem**.

3　On the Server Administrator navigation panel, under the **Chem** heading, click **Export**.

4　Enter a file name for the new process archive file; for example, `MyOrderCreatev20.par` for an archive based on the provided webMethods Chem eStandards Module sample OrderCreatev20.par archive.

The file extension must be .par to indicate that this is a process archive file. You should include the full path name as part of the file name to specify exactly where you want the archive file located on your file system. The default location is the following directory:

webMethods6\IntegrationServer\packages\WmChem\export

where webMethods6\IntegrationServer is the directory in which your Integration Server is installed.

5　Enter a name for the new archived Chem eStandards process; for example, `CustomPurchaseOrder` for a process based on the provided sample *PurchaseOrder* process in the *E41 OrderCreate* version 2.0 process archive.

6　Enter a number for the new archived Chem eStandards process; for example, `Z41` for a process based on the provided sample *E41 OrderCreate* version 2.0 process archive.

7　Enter a version number for the new archived Chem eStandards process; for example, `2.0` for a process based on the provided *E41 OrderCreate* version 2.0 process archive.

> **Note:** The version number that you enter here is meant for display purposes only. That is, when you import this archive, you see the version number that you specified while exporting. The version number that you enter here has no bearing on the contents of the archive.

8　Select the checkboxes for all of the Chem eStandards process IS document types that you want to export as part of the new process archive.

You should typically select all of the IS document types that were included in the standard process archive upon which you based your custom process archive, as well as any new IS document types you created for the custom archive.

**9**    Select the checkboxes for all of the TN XML document types that you want to export as part of the new process archive.

You should typically select all of the TN XML document types that were included in the standard process archive upon which you based your custom process archive, as well as any new TN XML document types you created for the custom archive.

**10**   Select the checkboxes for all of the TPAs that you want to export as part of the new process archive.

You should typically select all of the TPAs that were included in the standard process archive upon which you based your custom process archive, as well as any new TPAs you created for the custom archive.

**11**   Click **Create Archive**.

# Advantages and Disadvantages of Customizing and Exporting Chem eStandards Process Archives

If you decide to customize and export Chem eStandards process archives, you should be aware of the advantages and disadvantages, which are listed in the following table.

| Advantages of Customizing and Exporting | Disadvantages of Customizing and Exporting |
|---|---|
| Enables the webMethods Chem eStandards Module to manage your custom Chem eStandards process as distinct from the standard Chem eStandards process on which you based your customizations. | After you create a custom Chem eStandards process based on a standard Chem eStandards process, you must, from that point on, manually keep the custom process in parallel with the standard process; for example, when new features are added to the process. |
| Enables you to send the custom Chem eStandards process to your trading partners (if they are also using the webMethods Chem eStandards Module). | Customizing and exporting a Chem eStandards process does not save any services or process models associated with that process. Exporting a Chem eStandards process only saves the TPAs, IS document types, and TN XML document types associated with the process (and only those TPAs, IS document types, and TN XML document types that are contained in the wm.b2b.chem.rec.payloads folder). |
| | To save the services, you must create your own package on the Integration Server and place all of your services within that package. You can then manage that package, as required by your installation, including sending that package to your trading partners. To save the process model for the customized Chem eStandards process, you must use Modeler. |
| Enables you to do your own version management on your customized Chem eStandards processes. | |

# Defining TN XML Document Types

# What are TN XML Document Types?

TN XML document types are definitions that tell webMethods Trading Networks how to identify a type of business document and specify the attributes that Trading Networks is to extract from the business document.

When the webMethods Chem eStandards Module receives a business document, it invokes a Trading Networks service to recognize the type of business document by using the TN XML document types that you set up. When Trading Networks recognizes the TN XML document type of the business document, Trading Networks extracts specific pieces of information from the business document based on the document attributes you specify in the TN XML document type.

For more information about TN XML document types, see the *Building Your Trading Network* manual.

# TN XML Document Types Provided by WebMethods

When you install the webMethods Chem eStandards Module on the webMethods Integration Server, the TN XML document types listed in the following table are defined in Trading Networks:

| TN XML document type | Description |
|---|---|
| AcceptanceAcknowledgement | XMLDocType for RNIF 1.1 AcceptanceAcknowledgement |
| AcceptanceAcknowledgementException | XMLDocType for RNIF 1.1 AcceptanceAcknowledgementException |
| Chem Dummy Document Type | Dummy document type used in the Chem eStandards process model templates |
| RN11BizDocType | RN11BizDocType for version 1.1 |
| Exception(RNIF1.1) | XML document type for RNIF 1.1 Exception |
| ReceiptAcknowledgementException | XML document type for RNIF 1.1 ReceiptAcknowledgementException |
| ReceiptAcknowledgment(RNIF1.1) | XML document type for RNIF 1.1 ReceiptAcknowledgement |

You can view these TN XML document types in the Trading Networks Console.

# Information You Supply To Define TN XML Document Types

To define a TN XML document type, you specify the following identification information:

■ **The root tag of a business document.** To identify a document that has a declared root tag value of cXML, you would specify "cXML". For example, the document might contain the following document type declaration containing the declared root tag:

    <!DOCTYPE cXML SYSTEM "cXML.dtd">

■ **The *processVersion* pipeline variable name and its value.** You must specify the **Name** and the **Value** of the pipeline matching variable so that Trading Networks checks to see that both exist in the pipeline.You need to specify the *processVersion* pipeline matching variable and its value for an external TN XML document type only.

For more information about specifying the root tag and the pipeline matching variable to define a TN XML document type, see the *Building Your Trading Network* manual.

# Defining TN XML Document Types in Trading Networks

You must define the TN XML document types that you need to use for your Chem eStandards conversations. For example, for each Chem eStandards process, such as *OrderCreate* or *OrderResponse*, involved in the conversation, you need to define:

■ The private, or internal, TN XML document type representing your back-end system business document, if you are the conversation initiator.

■ The public, or external, TN XML document type representing a Chem eStandards record, if you are the conversation fulfiller.

**Note:** If you change the name of a TN XML document type, you must also change the name of it in the process model and in any associated trading partner agreements (TPAs); that is, where you have used the TN XML document type name as the agreement ID. For more information about process models, see Chapter 9, "Customizing a Process Model Template" in this guide and the *webMethods Modeler User's Guide*.

You create TN XML document types in the **Document Type Details** screen in the Trading Networks Console.

## Defining an Internal TN XML Document Type

When you define an internal TN XML document type, you specify the root tag from within the business document that the TN XML document type is to match and the XML queries that Trading Networks uses to extract the SenderID and ReceiverID attributes from the business document.

**To define an internal TN XML document type in Trading Networks**

**1** Start the Trading Networks Console. If you need procedures for this step, see the *Getting Started with Trading Networks* manual.

**2** In the Selector Panel, click **My Enterprise**.

**3** Select **View** ▸ **Document Types.**

**4** Select **Types** ▸ **New.**

**5** In the **Create New DocType** dialog box, select the **XML** document type category from the list, and click **OK**.

**6** In the **Document Type Details** screen, the **Name** field, type the name you want to give the internal TN XML document type.

**7** In the **Description** field, type a description for the internal TN XML document type.

**8** In the **Root Tag** field, type the value of the root tag from within the business documents that this internal TN XML document type is to match.

The following figure illustrates the **Document Type Details** screen with the aforementioned three fields completed.

**Document Type Details Screen for an Internal TN XML Document Type**

> **Note:** You do not need to specify the *processVersion* pipeline matching variable for internal back-end business documents. You specify this variable for external business documents only.

**9** Specify SenderID as one of the attributes to extract.

   **a** Click the **Extract** tab.

   **b** Click **Add an Attribute** .

   **c** In the **Add an Attribute** dialog box, the **Name** list, select **Sender ID**.

   **d** Select the **Required** check box.

   **e** In the **Description** field, enter a description for the attribute.

   **f** In the **Query** field, enter the query to extract the SenderID attribute. For the query to extract the SenderID attribute, see "SenderID XML Queries" on page 57 in this guide.

   **g** In the **Detail** list, select **DUNS**.

   **h** Click **OK**.

**10** Repeat step 9 for the ReceiverID attribute. For the query to extract the ReceiverID attribute, see "ReceiverID XML Queries" on page 57 in this guide.

## Defining an External TN XML Document Type

When you define an external TN XML document type, you specify the root tag from within the business document that the TN XML document type is to match, the pipeline matching variable of *processVersion*, and the value for *processVersion*. You do not need to specify XML queries to extract the SenderID and ReceiverID attributes because Trading Networks determines the SenderID and the ReceiverID from the external business document's Service Header.

**To define an external TN XML document type in Trading Networks**

**1** Start the Trading Networks Console. If you need procedures for this step, see the *Getting Started with Trading Networks* manual.

**2** In the Selector Panel, click **My Enterprise**.

**3** Select **View ▶ Document Types.**

**4** Select **Types ▶ New.**

**5** In the **Create New DocType** dialog box, select the **XML** document type category from the list, and click **OK**.

**6**   In the **Document Type Details** screen, the **Name** field, type the name you want to give the external TN XML document type.

**7**   In the **Description** field, type a description for the external TN XML document type.

**8**   In the **Root Tag** field, type the value of the root tag from within the business documents that this external TN XML document type is to match.

**9**   Click **Add New Pipeline Match** .

**10**   In the **Add Pipeline Matching** dialog box, the **Name** field, type `processVersion`, the name of the pipeline matching variable.

> **Important!** The name of the pipeline matching variable is **always** processVersion.

**11**   In the **Value** field, type the version of the Chem eStandards record you are creating.

> **Important!** The value specified is **always** the version of the Chem eStandards record you are creating.

The following figure illustrates the **Document Type Details** screen with the aforementioned fields completed.

**Document Type Details Screen for an External TN XML Document Type**
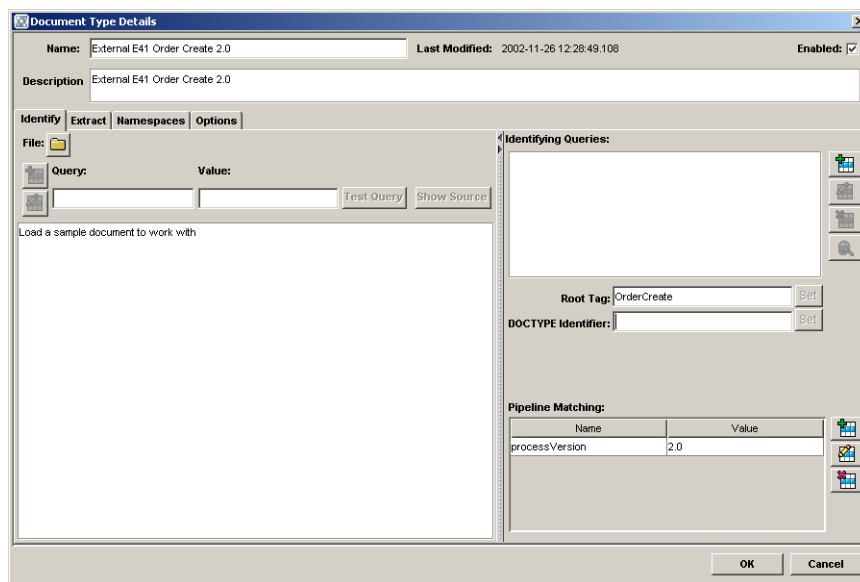
# XML Queries Used to Extract Attributes From Business Documents

Trading Networks uses XML queries to extract the SenderID and ReceiverID attributes from internal business documents sent to the webMethods Chem eStandards Module from a back-end system. Trading Networks uses the SenderID and ReceiverID to determine who is sending the business document and whom to send the business document to.

**Note:** If a business document is an external business document received from a trading partner, Trading Networks uses the SenderID and ReceiverID from the business document's Service Header.

The following sections provide the XML queries used to extract the SenderID and Receiver ID attributes.

## SenderID XML Queries

The SenderID is always the D-U-N-S$^{®}$ Number of the sender. The query to extract the SenderID for business documents (for example, Order Create, Order Response, and so on) is the following:

```
/ServiceHeader [0]/ProcessControl [0]/TransactionControl
[0]/ActionControl/PartnerRoute [0] /fromPartner [0]/PartnerDescription
[0]/BusinessDescription [0]/GlobalBusinessIdentifier [0]
```

The query to extract the SenderID for signal documents (for example, Receipt ACK, Exception, and so on) is as follows:

```
/ServiceHeader [0]/ProcessControl [0]/TransactionControl
[0]/SignalControl [0]/PartnerRoute [0]/fromPartner [0]/PartnerDescription
[0]/BusinessDescription [0]/GlobalBusinessIdentifier [0]
```

## ReceiverID XML Queries

The ReceiverID is always the D-U-N-S Number of the receiver. The query to extract the ReceiverID for business documents (for example, OrderCreate, OrderResponse, and so on) is as follows:

```
/ServiceHeader [0]/ProcessControl [0]/TransactionControl
[0]/ActionControl [0]/PartnerRoute [0]/toPartner [0]/PartnerDescription
[0]/BusinessDescription [0]/GlobalBusinessIdentifier [0]
```

The query to extract the ReceiverID for signal documents (for example, Receipt ACK, Exception, and so on) is as follows:

```
/ServiceHeader [0]/ProcessControl [0]/TransactionControl
[0]/SignalControl [0]/PartnerRoute [0]/toPartner [0]/PartnerDescription
[0]/BusinessDescription[0]/GlobalBusinessIdentifier [0]
```

# Defining Trading Partner Agreements

# What is a Trading Partner Agreement?

A *Trading Partner Agreement (TPA)* is a set of parameters that you can use to govern how business documents are exchanged between two trading partners. For example, a TPA might contain parameters that specify whether the outbound business document for a process model should be signed, whether the inbound business document for a process model should contain your trading partner's signature information, and so on.

Each party in a Chem eStandards conversation, the initiator/sender and the fulfiller/receiver, must be associated with a TPA. You create and view TPAs in the Trading Networks Console **Agreement Details** screen. The TPAs that you create in Trading Networks contain a set of parameters that map to some (but not all) elements in the Service Header of a business message.

For more information about TPAs in general, see the *Building Your Trading Network* manual.

# How Does the webMethods Chem eStandards Module Identify a TPA?

The webMethods Chem eStandards Module identifies a TPA by a unique Sender, Receiver, and Agreement ID. During a conversation between trading partners, the webMethods Chem eStandards Module uses this information to retrieve the TPAs for the initiator and fulfiller in the conversation and to process the business documents exchanged.

For the initiator, the webMethods Chem eStandards Module locates the Sender, Receiver, and Agreement ID from the starting TN XML document type. For the fulfiller, the webMethods Chem eStandards Module locates the Sender, Receiver, and Agreement ID from the Service Header in the Chem eStandards record.

# Defining a TPA

Before you define a TPA, you must already have defined your enterprise profile and the trading partner profiles of the trading partners with whom you plan to exchange business documents.

When you define a TPA, you supply the following information: sender, receiver, agreement ID, IS document type, data status, and optionally, a description. You also define the values for TPA parameters. The following table lists and describes the TPA information at the top of the **Agreement Details** screen that you need to understand and modify. For step-by-step instructions about how to define a TPA, see the *Building Your Trading Network* manual.

| TPA Information | Description |
|---|---|
| Sender | The name of the trading partner that has the sender role in the TPA. You select the sender from the profiles defined on your Trading Networks system, including your own profile (**My Enterprise**). The Sender is always a D-U-N-S® Number. |
| Receiver | The name of the trading partner that has the receiver role in the TPA. You select the receiver from the profiles defined on your Trading Networks system, including your own profile (**My Enterprise**). The Receiver is always a D-U-N-S Number. |
| Agreement ID | An application-specific field that uniquely identifies the type of agreement between two partners. If you are the initiator in the Chem eStandards conversation, the Agreement ID for the TPA is the starting, or internal, TN XML document type name. If you are the fulfiller for the Chem eStandards conversation, the Agreement ID for the TPA is the TN XML document type name that specifies the type of Chem eStandards record that you expect to receive from the initiator. |
| IS Document Type | The wm.ip.chem.rec:UserParameters IS document type that specifies the data, or the parameters, that you define in the TPA. You can find the wm.ip.chem.rec:UserParameters IS document type in the WmChem package. To define a TPA, you need to know the fully qualified name of this IS document type. |
| Data Status | This field applies only when the **Agreement Status** is Agreed. The data status indicates whether you can update the values for the TPA parameters defined in the IS document type (located in the panel on the lower portion of the **Agreement Details** dialog box). The data status can be one of the following:

■ **Modifiable** - the TPA data can be changed

■ **Non-modifiable** - the TPA data cannot be changed |

**Important!** The Sender, Receiver, and Agreement ID **must** be unique for each TPA. After you create a TPA, you cannot change these values.

> ⚠️ **Important!** The webMethods Chem eStandards Module does **not** support a TPA in which either the Sender is known (for example *Company1*) and the Receiver is *Unknown* or in which the Sender is *Unknown* and the Receiver is known (for example, *Company2*). The webMethods Chem eStandards Module supports only those TPAs in which either both the Sender and Receiver are known (for example, *Company1* and *Company2*, respectively) or in which the Sender and Receiver are *Unknown*.

The following figure illustrates the **Agreement Details** screen that currently displays a TPA. For further explanation, see the text that follows this figure.

**Agreement Details Screen**



This TPA is an initiator's TPA because the Agreement ID is InternalOrderCreate, indicating that the Agreement ID is a starting, or internal, TN XML document type. In the lower half of the screen are the TPA parameters with their current values displayed. You define or modify TPA parameter values in the **Input for 'wm.ip.cm.rec:UserParameters'** screen.

For a list and descriptions of the TPA parameters, see the next section, "Parameter Settings".

# Parameter Settings

TPA parameters are divided into two sections:

- **ProcessInfo.** Consists of process-level information, such as the Transport, the TransportVersion, and so on.

- **PipInfo.** Consists of information about the Chem eStandards process, such as the DTD, ActionCode, and so on. Do **not** edit Chem eStandards process information unless you are implementing a customized process.

The following table lists and describes the TPA parameters.

| TPA Section | Parameter | Description |
| --- | --- | --- |
| ProcessInfo | Transport | The transport protocol type; for example, Chem. |
| | TransportVersion | The version of the transport protocol; that is, 1.1. |
| | Sign | Whether the outbound Chem eStandards record for the process model should be signed. Valid values are *yes* or *no*. |
| | SignatureRequired | Whether the inbound Chem eStandards record for the process model should contain your trading partner's signature information. Valid values are *yes* or *no*. |
| | HashAlgorithm | The algorithm to use to generate the Chem eStandards Message Digest; that is, MD5 or SHA-1. |
| | UseProfileCerts | For HTTPS connections, whether to use the certificates from the **Sign** tab in the **Security** section of the receiving trading partner profile. The default value for this parameter is *No*, indicating that the webMethods Chem eStandards Module will use the Integration Server default certificate. *Yes* indicates that the webMethods Chem eStandards Module will use the certificates from the receiving trading partner profile. |
| | ValidationService | This parameter does not apply to the webMethods Chem eStandards Module. |
| | ValidateOutput | Whether validation of the Chem eStandards record will be performed before you send the document to your trading partner. Valid values are *yes* or *no*. |
| PIPInfo | ProcessCode | The code for, or name of, the Chem eStandards process, such as E41. The value for this parameter is derived from the Chemical Industry Data Exchange (CIDX™) Specification. The value for this parameter applies to a particular Chem eStandards process and a particular version of a Chem eStandards process. You should not modify this parameter. |
| | ProcessVersion | The version of the Chem eStandards process, such as 2.0. The value for this parameter is derived from the CIDX Specification. The value for this parameter applies to a particular Chem eStandards process and a particular version of a Chem eStandards process. You should not modify this parameter. |

| TPA Section | Parameter | Description |
|---|---|---|
| PIPInfo (cont.) | ProcessDescription | The description of the Chem eStandards process; for example, the description of the *OrderCreate* process is Order Create. The value for this parameter is derived from the CIDX Specification. The value for this parameter applies to a particular Chem eStandards process and a particular version of a Chem eStandards process. You should not modify this parameter. |
| | TransactionCode | The transaction associated with each Chem eStandards record. The value for this parameter is derived from the CIDX Specification. The value for this parameter applies to a particular Chem eStandards process and a particular version of a Chem eStandards process. You should not modify this parameter. |
| | ActionCode | The action associated with each Chem eStandards record. The value for this parameter is derived from the CIDX Specification. The value for this parameter applies to a particular Chem eStandards process and a particular version of a Chem eStandards process. You should not modify this parameter. |
| | DTD | The file name of the Data Type Definition for the Chem eStandards record. The value for this parameter is derived from the CIDX Specification. The value for this parameter applies to a particular Chem eStandards process and a particular version of a Chem eStandards process. You should not modify this parameter. |
| | NSFolder | The folder that contains the IS document types against which the Chem eStandards records will be validated. The value for this parameter applies to a particular Chem eStandards process and a particular version of a Chem eStandards process. You should not modify this parameter. |
| | UsageCode | Whether the process model is in Test or Production mode. Valid values are *Test* or *Production*. |
| | SenderClassification | The business classification of the sender; for example, Manufacturer, Distributor, Customers Broker. |
| | ReceiverClassification | The business classification of the receiver. |

# Mapping a Chem eStandards Record

# What Is "Mapping" a Business Document?

"Mapping" a business document refers to the process of assigning the structure, values, or content of one or more business documents to a new business document; that is, mapping the values, data, and information from various sources into a single, new business document.You need to map business documents because, typically, your back-end systems have different document formats than that of the Chem eStandards records.

## Why Do You Create an Outbound Mapping Service?

You create an outbound mapping service to translate an outbound back-end proprietary business document (for example, an IDOC from an SAP R/3 system) to a Chem eStandards record. Elements of the proprietary business document need to be mapped to corresponding elements in a specific version of a Chem eStandards record. Extra elements in the back-end business document are ignored, as long as values are mapped to **all** elements in the Chem eStandards record. Examples of outbound mapping services are located in the WmChemSample package.

## Why Do You Create an Inbound Mapping Service?

You create an inbound mapping service to map each element of the inbound Chem eStandards record to a corresponding element in your back-end proprietary business document format.

For example, if you use SAP R/3 as your back-end ERP system, and you want to receive a Chem eStandards record via the webMethods Chem eStandards Module, you create an inbound mapping service that maps each element of the Chem eStandards record to a corresponding element in the IDOC format. Examples of inbound mapping services are located in the WmChemSample package.

# Example of Mapping a Business Document

The following figure illustrates mapping a business document for a two-action Chem eStandards process. For further explanation, see the table that follows the figure.

**Example of Mapping a Business Document**
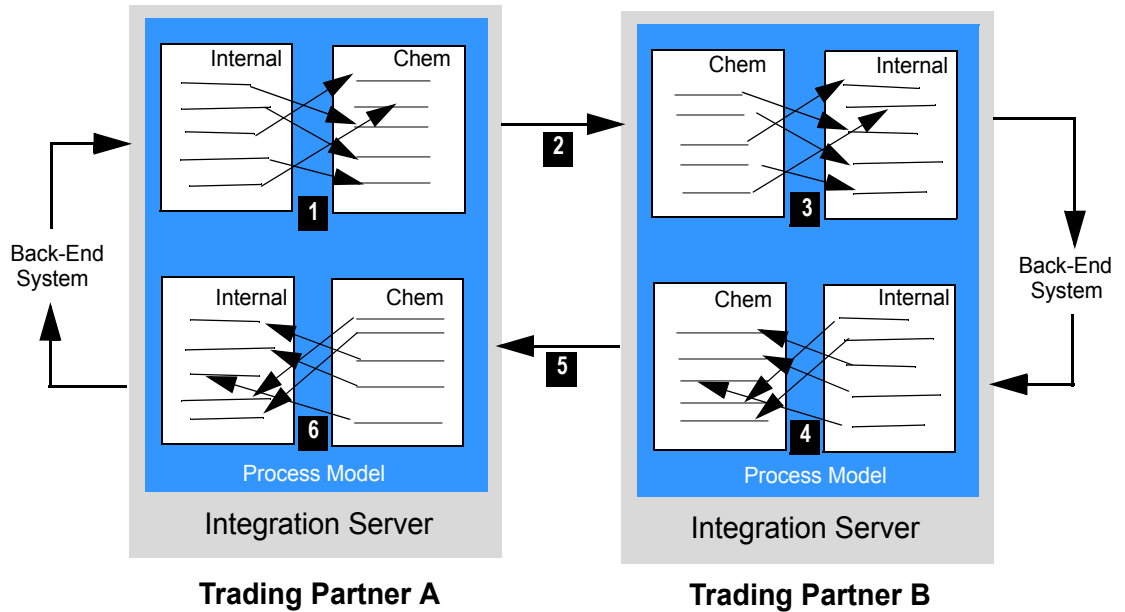


| Step | Description |
|---|---|
| 1 | Trading Partner A uses an outbound mapping service to map an internal business document from a back-end system to a Chem eStandards record. |
| 2 | Trading Partner A sends the Chem eStandards record to Trading Partner B. |
| 3 | Trading Partner B receives the Chem eStandards record and uses an inbound mapping service to map the Chem eStandards record to an internal business document. After the business document is mapped, it is in a format that Trading Partner B's back-end system can process. |
| 4 | Trading Partner B uses an outbound mapping service to map the internal response document from the back-end system to a Chem eStandards record. |

 

| Step | Description |
|------|-------------|
| 5 | Trading Partner B sends the Chem eStandards record to Trading Partner A. |
| 6 | Trading Partner A receives the Chem eStandards record and uses an inbound mapping service to map the Chem eStandards record to an internal business document. After the business document is mapped, it is in a format that Trading Partner A's back-end system can process. |

# Creating an Outbound Mapping Service

In webMethods Developer, you create an outbound mapping service by creating a new flow service. This flow service contains one or more MAP entities, which do the actual mapping from your back-end business document, through any desired intermediate steps, to the IS document type for the appropriate outbound Chem eStandards record. For information about mapping data in a flow service, see the *webMethods Developer User's Guide*. For examples of how to create an outbound mapping service, see the sample in the WmChemSample package.

## Input/Output to Use

The input to the outbound mapping service is, indirectly, your back-end business document, represented as an IData object.

**Note:** You typically invoke wm.ip.util:getBizDocFromEvent to retrieve the BizDocEnvelope, which contains your back-end business document, and then invoke wm.ip.util:recordFromBizDoc to extract the business document content from the BizDocEnvelope. This process returns your back-end business document as an IData object.

The output from the outbound mapping service is the Chem eStandards record in the **documents\Payload** IData object in the pipeline. For instance, if you are mapping your back-end documents to the *OrderCreate* Chem eStandards record, the output of the mapping service would be *OrderCreate(OrderCreate)* in the **documents\Payload** IData object. See the figure in "Example of an Outbound Mapping Service: Initiator/Sender" on page 70 in this guide.

The output from the outbound mapping service **must** be placed in the **documents\Payload** IData object. The webMethods Chem eStandards Module has to convert a Chem eStandards IData object into an XML string before sending the Chem eStandards record to the trading partner. Therefore, the webMethods Chem eStandards Module must know the precise location of the Chem eStandards IData object.

**Note:** If the documents for your back-end system have DTDs, you can automatically import an external DTD in Developer to provide a starting point for mapping. Simply create a new external record and specify the source as *XML, DTD,* or *XML Schema.*

## Flow Operations to Use

In the flow service, you insert a MAP operation and use the service pipeline to map elements of the IS document type for your back-end business document to all elements of the IS document type for the appropriate Chem eStandards record. Any unmapped elements in your back-end business document are dropped. Built-in IS document types for all versions of the Chem eStandards processes that you imported are located in the WmChemPayloads package in the wm.b2b.chem.rec.payload folder, which is displayed in the following figure.

**Imported IS Document Types**
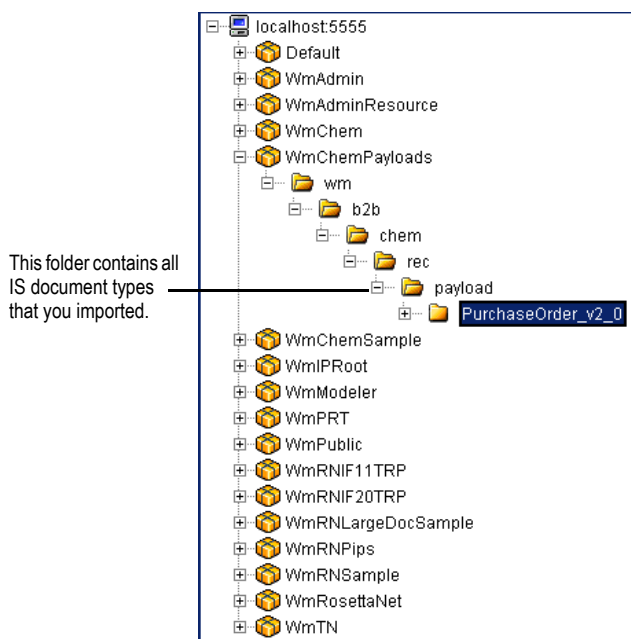
This folder contains all IS document types that you imported.

# Example of an Outbound Mapping Service: Initiator/Sender

If you are the initiator/sender and are implementing a version of the *OrderCreate* process, the outbound mapping service might look like the following figure. For further explanation, see the text that follows the figure.

**Example of an Outbound Mapping Service: Initiator/Sender**
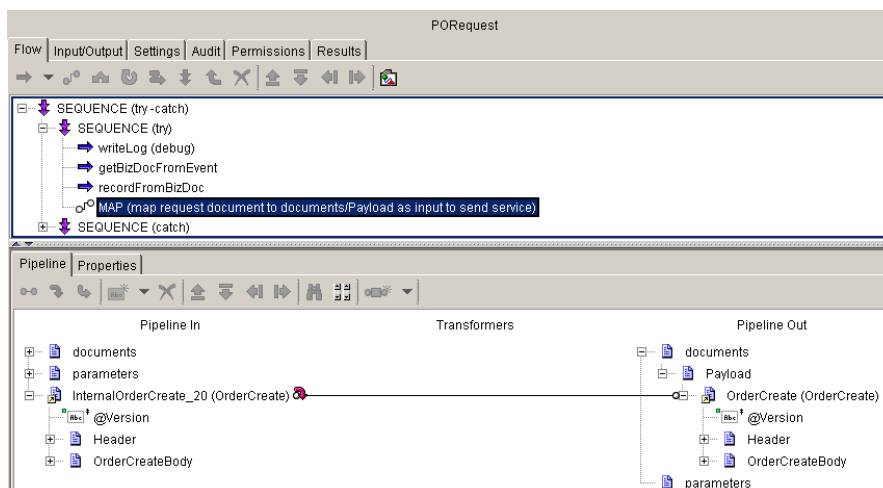


Under the **Pipeline Out** heading, in the **documents\Payload** IData object, the **OrderCreate** not in parentheses is the *IData object*, or Chem eStandards record, being sent to the trading partner and the **OrderCreate** in parentheses is the *IS document type* that defines the Chem eStandards record. The name of the IData object is case sensitive.

**Important!** As a general convention, the name of the IData object should be the same (including the case) as the name of the IS document type. If not, the validation on the receiver's end will fail. Using the above figure as an example, the receiver's Chem eStandards-compliant server will expect the name of the incoming Chem eStandards record to have a Chem eStandards process name of **OrderCreate**.

## Example of an Outbound Mapping Service:Fulfiller/Receiver

If you are the fulfiller/receiver and are implementing a version of the *OrderCreate* process, the outbound mapping service might look like the following figure. For further explanation, see the text that follows the figure.

**Example of an Outbound Mapping Service: Fulfiller/Receiver**
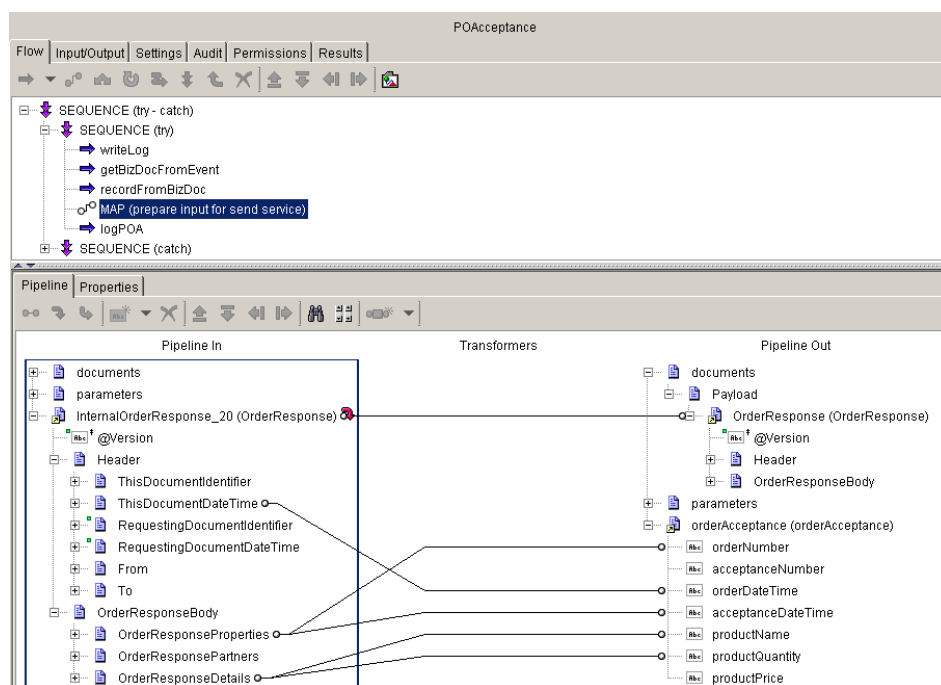


The fulfiller returns a Purchase Order (PO) Response to the initiator. Note that the **InternalOrderResponse** IData object that contains information from the fulfiller's back-end system is mapped to the **OrderResponse** IData object, or the Chem eStandards record in the **documents\Payload** IData object.

# Creating an Inbound Mapping Service

You create an inbound mapping service in Developer by creating a new flow service. This flow service contains one or more MAP entities, which do the actual mapping from the received Chem eStandards record, through any intermediate steps, to the format of your back-end business document.

## Input/Output to Use

Unlike an outbound mapping service, one input to an inbound mapping service is the incoming Chem eStandards record, represented as an IData object in the *userDocument*

---

IData object in the pipeline. This IData object **must** be named *userDocument*, which is the IData object in which the webMethods Chem eStandards Module places the inbound Chem eStandards record after validating it and recording it to the Integration Server Repository. The inbound document is placed in this pipeline variable so that the information is immediately available without requiring an additional access call to data storage (thus improving performance). See the figure in "Example of an Inbound Mapping Service: Initiator/Sender" on page 73 in this guide.

For RNIF version 1.1, the inputs to the inbound mapping service are *userDocument*, the Preamble Header, the Service Header, and attachments (optional).

**Note:** If the documents for your back-end system have DTDs, you can automatically import an external DTD in Developer to provide a starting point for mapping. Simply create a new external record and specify the source as *XML*, *DTD*, or *XML Schema*.

## Flow Operations to Use

In the flow service, you insert a MAP operation and use the pipeline to map elements of the IS document type for the Chem eStandards record to elements in your back-end business document.

## Example of an Inbound Mapping Service: Initiator/Sender

If you are the initiator/sender and are implementing a version of the *OrderCreate* process, the inbound mapping service might look like the following figure. For further explanation, see the text that follows the figure.

**Example of an Inbound Mapping Service: Initiator/Sender**
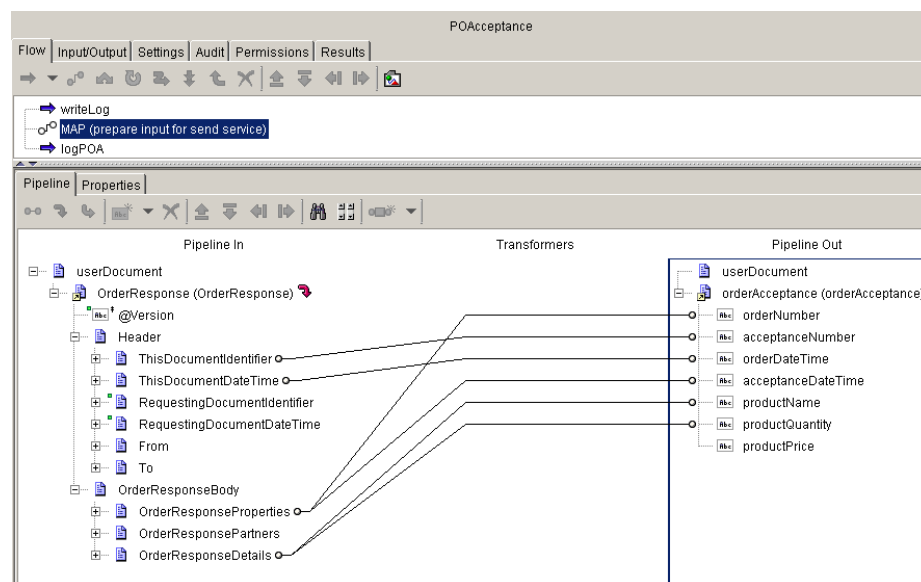


When the webMethods Chem eStandards Module for the initiator receives the inbound PO Response from the fulfiller, it places the **OrderResponse** IData object in the **userDocument** IData object in the pipeline. The **OrderResponse** IData object is mapped to the **orderAcceptance** I Data object, a back-end business document.

### Example of an Inbound Mapping Service: Fulfiller/Receiver

If you are the fulfiller/receiver and are implementing a version of the *OrderCreate* process, the inbound mapping service might look like the following figure. For further explanation, see the text that follows the figure.

**Example of an Inbound Mapping Service: Fulfiller/Receiver**
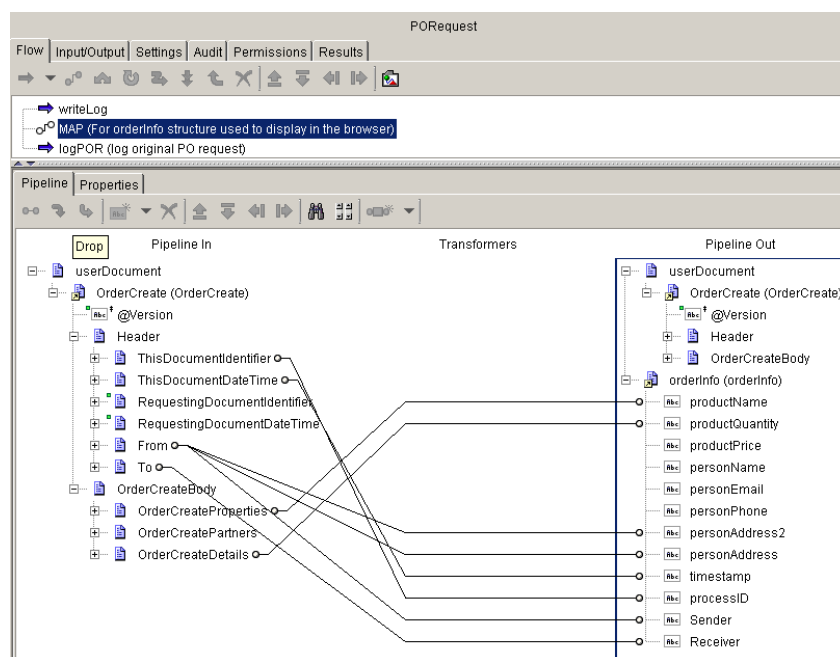


When the webMethods Chem eStandards Module for the fulfiller receives the inbound PO from the initiator, it places the **OrderCreate** IData object in the **userDocument** IData object in the pipeline. The **OrderCreate** IData object is mapped to the **orderinfo** IData object, a back-end business document.

## Reusing Mapping Services

In the webMethods Chem eStandards Module, you can reuse mapping services for trading partners that submit the same document format. For example, you can use the same mapping services for Trading Partner A and Trading Partner B if they use the same version of the same Chem eStandards process and they both always submit a business document in the same document format to the webMethods Chem eStandards Module. As the receiver of those business documents, you only need to define one inbound mapping service for both trading partners, because the document format and Chem eStandards process versions are the same.

You generally cannot reuse the same mapping service for different versions of the same Chem eStandards process because the mapping service includes references to IS document types within a specific version of a Chem eStandards process.

# Customizing a Process Model Template

# What is a Process Model?

A process model is a figure that represents a business process. The following figure illustrates a sample process model. For further explanation, see the text that follows the figure.

**Sample Process Model**
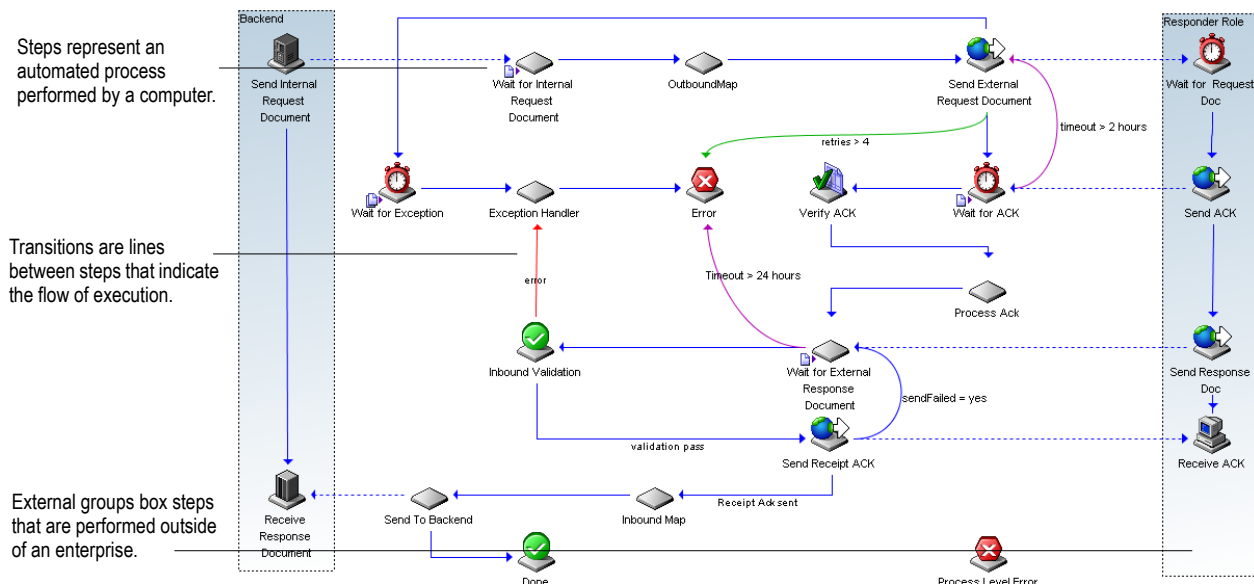


A process model consists of:

■ **Steps.** The basic unit of work in a process model.

■ **Transitions.** The lines between steps that indicate the execution order of steps within the process model.

■ **Groups.** Clustering steps to represent different organizational boundaries within a process model.

■ **Annotations (notes or text).** The labels, notes, and explanatory text in a process model.

For more information about steps, transitions, groups, and annotations, see the *webMethods Modeler User's Guide*.

The steps in a process model determine how the process run time conducts a conversation, which includes how the process run time processes a business document. For information about the process run time, see "webMethods Chem eStandards Module Run-Time Architecture/Components" on page 26 in this guide and the *webMethods Modeler User's Guide*.

The webMethods Chem eStandards Module provides process model templates that you can customize to build your own process models. You customize a process model

template by specifying how the process model is to interact with your back-end system. You do this by editing the services that are invoked by the steps of the process model, replacing the Chem eStandards Dummy Doc Types with the TN XML document types that you have created for steps that are waiting for action documents, specifying inbound and outbound mapping services, and enhancing the provided error and exception handling services, if necessary. When customizing a process model template, you also assign TN roles to TN XML document types and a focal role to the process model.

You use webMethods Modeler to customize the process model templates based on the type of Chem eStandards process you are implementing. For more information about process models and Modeler, see the *webMethods Modeler User's Guide*.

## TN Roles and Focal Roles

When you customize a process model template, you assign both TN roles and a focal role.

### TN Roles

In Modeler, you assign TN roles to TN XML document types that are associated with steps that wait to receive action documents, such as a purchase order (PO) or a PO acceptance (POA). A *TN role* specifies the role of a user for a specific business document. The TN roles that you can assign to a TN XML document type for an action document depend on the Chem eStandard process you are implementing.

To find the TN roles to assign to TN XML document types, refer to the specific process description section for the process you are implementing in the Chemical Industry Data Exchange (CIDX) *Chem eStandards*, which are available on the CIDX Web site at http://www.cidx.org.

For example, the TN roles you would assign to TN XML document types associated with steps that are waiting for a PO in an OrderCreate process model and a POA in an OrderResponse process model are "Seller" and "Buyer," respectively, as indicated in the OrderCreate and OrderResponse sections of the CIDX *Chem eStandards*. If you send a PO to your trading partner, you, acting as a Buyer, are the initiator/sender for the PO and your trading partner, acting as the Seller, is the fulfiller/receiver of the PO. When you receive a POA from your trading partner, you, the Buyer, are the fulfiller/receiver of the POA, and your trading partner, the Seller, is the initiator/sender of the POA. In other words, you or your trading partner may be either the initiator/sender or the fulfiller/receiver for a business document depending on the document being sent, regardless of the role that you or your trading partner play in the conversation as a whole, as indicated in the following illustration.

**TN Roles**



When you replace a Chem eStandards Dummy Doc Type with a TN XML document type that you created for a particular wait step, Modeler prompts you to enter the TN roles for the **Sender Role** and the **Receiver Role** in the **Set TN Roles** dialog box. For example, if you are replacing a Chem eStandards Dummy Doc Type for a particular wait step with the name of a PO that you are sending to a trading partner, you would enter "Buyer" as the **Sender Role** and "Seller" as the **Receiver Role**.

### Focal Role

When you customize a process model template, you also assign a focal role to the process model. A *focal role* specifies the role of the user for a particular conversation, or process. As with TN roles, you determine the focal role you need to use from the specific process description section for the process you are implementing in the Chemical Industry Data Exchange (CIDX™) *Chem eStandards.* You assign a focal role for your enterprise in the **Properties** dialog box in Modeler.

# Process Model Templates Provided by webMethods

webMethods provides the following process model templates to assist you in creating your process models:

- ◼ One Action Initiator Model.xml

- ◼ One Action Responder Model.xml

- ◼ Two Action Initiator Model.xml

- ◼ Two Action Responder Model.xml

You can find the process model templates in the following location:

webMethods6\IntegrationServer\packages\WmChem

where webMethods6\IntegrationServer is the directory in which the Integration Server is installed.

# Steps in the Process Model Templates That You Must Modify

In the process model templates, steps that you must modify are denoted with a plain, diamond-shaped icon ⬦ . The following table lists the steps in the process model templates that you must modify, as well as descriptions of the modifications and whether a step applies to the initiator, fulfiller, or both.

| Process Model Template Step | Description of Modification | Initiator | Fulfiller |
|---|---|---|---|
| **Wait for Internal Request Document** | Add a subscription to an internal TN XML document type. | ✔ | |
| **Wait for External Request Document** | Add a subscription to an external TN XML document type. | | ✔ |
| **Outbound Map** | Add an outbound mapping service to map your back-end document to a Chem eStandards record. | ✔ | ✔ |
| **Inbound Map** | Add an inbound mapping service to map your Chem eStandards record to a back-end document. | ✔ | ✔ |
| **Send to Back-End** | Add a service to send the business document to the back-end system. | ✔ | ✔ |
| **Wait for External Response Document** | Add a subscription to an external TN XML document type. | ✔ | |
| **Wait for Internal Response Document** | Add a subscription to an internal TN XML document type. | | ✔ |
| **Process ACK** | Add a service to handle the acknowledgement; for example send the ACK to another department in your enterprise, trigger another service, and so on. | ✔ | ✔ |
| **Exception Handler** | Add a service to handle the exception condition in your desired way; for example, to send an e-mail message. | ✔ | ✔ |

**Note:** The process model templates contain one step that you can modify if you want to. This step is the **Error** step. You can modify this step only if you will be invoking wm.ip.cm.handlers:error in the service that you are going to specify for this step.

# Prerequisites for Customizing a Process Model Template

The following table lists the prerequisites you should complete before you customize a process model template.

| Prerequisite | For more information, see … |
|---|---|
| Define the trading partner profiles for the trading partners with whom you will be exchanging business documents. | Chapter 4, "Defining Trading Partner Profiles in Trading Networks" in this guide and the *Building Your Trading Network* manual. |
| Define your TN XML document types. | Chapter 6, "Defining TN XML Document Types" in this guide and the *Building Your Trading Network* manual. |
| Import the Chem eStandards process archives you need to the webMethods Integration Server. | Chapter 5, "Importing Chem eStandards Process Archives" in this guide. |
| Create your inbound and outbound mapping services. | Chapter 8, "Mapping a Chem eStandards Record" in this guide. |
| Created your send-to-back-end service. | |
| Create error and exception handlers, if necessary. | Chapter 11, "Handling Errors and Exceptions" in this guide. This chapter provides information about the types of errors that might occur. |

# Customizing a Process Model Template

To customize a process model template, you must first determine the appropriate process model template you need to use. The process model template that you use depends on two criteria:

- Your role in the Chem eStandards conversation (that is, initiator/sender or fulfiller/receiver).

- The number of transactions in the Chem eStandards conversation (that is, one-action or two-action).

For example, the *OrderCreate* process would require either the initiator or fulfiller version of the two-action asynchronous process model template. To determine whether a Chem eStandards process is a one-action or two-action model, go to the CIDX Web site (http://www.cidx.org), and look up the specification for the type of Chem eStandards process archive that you imported to the Integration Server.

After you determine the appropriate process model template to use, you are ready to customize the template.

**To customize a process model template**

1    Start Modeler. If you need help with this step, see the *webMethods Modeler User's Guide*.

> **Note:** To start Modeler, your Integration Server must be running.

2    Select **File ▶ Import**.

3    Go to:

webMethods6\IntegrationServer\packages\WmChem\ProcessModels

where webMethods6\IntegrationServer is the installation directory in which the Integration Server is installed

4    Choose the process model template you want to work from, and click **Open**.

5    Select **File ▶ Save as New Version**.

6    Save the file under a different name than the template name.

7    For each step that waits for an action document, replace the Chem eStandards Dummy Doc Type with the TN XML document type that you created for the step. For example, if you are using the Two Action Initiator.xml process model template, you would replace the Chem eStandards Dummy Doc Type for the **Wait for Internal Request Document** and **Wait for External Response Document** steps with the TN XML document types you created for these steps. When you replace a Chem eStandards Dummy Doc Type with a TN XML document type, Modeler prompts you to assign the TN roles for the **Sender Role** and the **Receiver Role** to the TN XML document type. For information about TN roles, see "TN Roles" on page 79 in this guide. For information about how to assign the TN XML document types that wait steps subscribe to, see the *webMethods Modeler User's Guide*.

> **Note:** If you later change the name of an internal or external TN XML document type, you must also change the name of the TN XML document type in the process model to avoid an error.

8    If a wait step has a join type of "Complex," replace the Chem eStandards Dummy Doc Type for that join with the TN XML document type ID of the input business document assigned to that step.

   a    In the **Properties** dialog box, click the down arrow to the right of the **Complex** join type.

   b    In the **Join Type** list, click **Complex**.

    **c**    In the **Complex Join Editor** dialog box, click **Chem eStandards Dummy Doc Type**, and select the TN XML document type ID of the input business document assigned to the step.

> **Important!** If you do not replace a Chem eStandards Dummy Doc Type with the correct TN XML document type ID for a wait step with a join type of "Complex," the incoming business document will not join the conversation.

**9**    Assign services; that is:

- Assign outbound mapping and inbound mapping services to the **Outbound Mapping** and **Inbound Mapping** steps.

- Assign error and exception handlers, if necessary.

- Assign the send-to-back-end service.

For information about how to assign the service that a step invokes, see the next section, "Assigning the Service that a Step Invokes".

**10**    Enter a value for the **Focal Role** property. For information about focal roles, see "Focal Role" on page 80 in this guide. For information about setting this property, see the *webMethods Modeler User's Guide*.

> **Important!** If a user's focal role for a process is different than either of the TN roles for an incoming business document, the business document will not join the conversation.

**11**    Enter a value for the **Description** property. The description should briefly explain the use of the process model. For information about setting this property, see the *webMethods Modeler User's Guide*.

**12**    Set timeout and retry count values if you want these values to be different than the Chem specification. For information about how to set timeout and retry values using Modeler, see "Setting Timeout and Retry Count Values" on page 85 in this guide.

**13**    Modify step labels and icons as appropriate. For information about how to modify step labels and icons, see the *webMethods Modeler User's Guide*.

**14**    Save the process model.

**15**    Generate the process model; then update the process model for monitoring. For instructions about how to generate and update a process model, see the *webMethods Modeler User's Guide*.

When you generate a process model, a process model package (containing services and triggers) and a process run-time script (or fragment) are created on the Integration Server.

**16** In webMethods Monitor, enable the process model. For instructions about how to enable a process model, see the *webMethods Modeler User's Guide*.

## Assigning the Service that a Step Invokes

When you customize a process model template, you need to assign the services that the **Inbound Mapping** and **Outbound Mapping** steps invoke and the service that the **Send to Back-End** step invokes. If you create your own error and exception handlers, you will also need to set these services as the ones that your **Error** and **Exception Handler** steps invoke.

**To assign the service that a step invokes**

**1** Start Modeler. If you need help with this step, see the *webMethods Modeler User's Guide*.

**2** Open the process model template that you want to customize.

> **Note:** Remember to save the process model template under a different name.

**3** Right-click on the step for which you want to set the service to invoke.

**4** Click **Select service to invoke**.

**5** In the **Select service to invoke from server: Design Server** dialog box, browse the package folders, and select the service you want to invoke.

**6** Save your changes.

For information about other process model properties that you can change in Modeler, see the *webMethods Modeler User's Guide*.

## Setting Timeout and Retry Count Values

You can set both timeout and retry count values in Modeler. *Timeout* refers to the amount of time to wait to receive a Receipt Acknowledgment (ACK) or to receive a Response document, such as a POA. *Retry Count* refers to the number of times to attempt to send a Receipt ACK or to send a business document, such as a PO or POA, including the initial attempt to send the document.

The retry count value you set for a particular step determines the behavior of the process model at that step. The following table lists the possible retry count values and the effect of the value on a process model step.

| Retry Count Value | Effect of Retry Count Value on the Process Model Step |
|---|---|
| 0 | This value provides the step with an unlimited number of retries. |
| 1 | This value provides the step with one try only. On the second retry, the process fails. |
| >1 | For any value greater than 1, a "retry count exceeded" can take place. For example, if you set the retry count value to 2, and the process transitions to the step for a third attempt, the retry count is exceeded. For any value greater than 1, you must draw a "retry exceeded" transition from the step in question. |

You can set the retry count value for each step independently of other steps. You should be concerned most with setting the retry count value for steps where a process model is sending an action message to a trading partner. Setting the value of the **Retry Count** property in Modeler changes the value for your Chem eStandards conversation but does not change the value in your trading partner's Chem eStandards conversation.

**To set timeout and retry count values**

**1**   Start Modeler. If you need help with this step, see the *webMethods Modeler User's Guide*.

**2**   Open the process model template that you want to customize.

**Note:** Remember to save the process model template under a different name

**3**   Right-click on the step for which you want to set the timeout or retry count values.

**4**   Click **Properties**.

**5**   In the **Retry Count** field, enter the retry count.

**6**   In the **Timeout Value** field, enter the timeout (in milliseconds).

**7**   Save your changes.

For information about other process model properties that you can change in Modeler, see the *webMethods Modeler User's Guide*.

# Running and Monitoring a Conversation

# Running a Conversation and the ConversationID

When a conversation starts, the process run time selects the appropriate process model to use for the conversation. The process run time (PRT) can run multiple instances of the same process model at one time. Each instance is a different conversation. To keep the instances of the conversation distinct, each instance must have its own unique ConversationID. webMethods assigns each Chem eStandards-transaction conversation a unique ConversationID.

This ConversationID is the concatenation of the initiator's Global Business Identifier (a D-U-N-S® Number), a process instance identifier, and the focal role of the process model, separated with hyphens. For example, if you are sending a Purchase Order (PO) Request to initiate an Order Create conversation, the webMethods Chem eStandards Module creates a ConversationID for that conversation in the form of your D-U-N-S number, a Process Instance Identifier, and "Buyer" (because "Buyer" is the focal role of the initial process model for the *OrderCreate* process). When the PO Request arrives at the fulfiller's webMethods Chem eStandards Module site, the webMethods Chem eStandards Module creates a ConversationID for that conversation in the form of your D-U-N-S Number (because you were the initiator) the same process instance identifier (which is sent as part of the Chem eStandards business message), and "Seller" (because "Seller" is the focal role of the process model for the *OrderCreate* process on the fulfiller side when a PORequest is received. The following figure illustrates this concept.

**ConversationID**
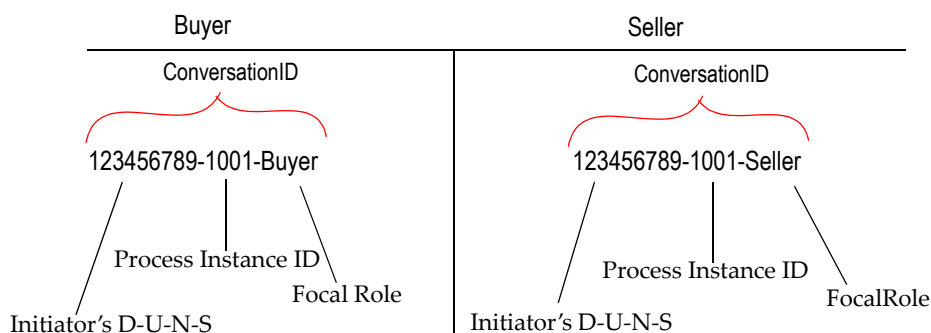


For the PRT to associate documents in a single conversation with the correct running instance of a process model, all documents entering the conversation must specify the same ConversationID. In a process model, the Send to Back-End step (the step that sends the document to the back-end) transitions to the back-end. When this occurs, the back-end system is passed information, including the ConversationID. For the conversation to continue properly when the back-end system returns the response, the documents from the back-end system must include the correct ConversationID. This means that, at design time, when you set up your back-end systems to participate in a conversation, you must ensure that the back-end system maintains the ConversationID when it receives a document.

You must maintain the ConversationID on your back-end system when you receive a document. You then must include the ConversationID in any Response document you send back so that the webMethods Chem eStandards Module can identify the proper conversation to which the Response document belongs and make sure the Response document rejoins the correct existing conversation.

## XML Queries Used To Extract and Generate the ConversationID

When the webMethods Chem eStandards Module receives a business document, it invokes the Trading Networks wm.tn.doc:recognize service. If the business document is from a back-end system, the wm.ip.cm:process Document service extracts the initiating trading partner's D-U-N-S Number, a process instance ID number, and the focal role of the transaction from the business document and uses these elements to generate the ConversationID, as explained in the previous section. The following sections illustrate the XML queries used to retrieve the elements that make up a ConversationID.

### Initiator's D-U-N-S Number

The query to extract the initiator's D-U-N-S Number is:

```
/ServiceHeader[0]/ProcessControl[0]/ProcessIdentity[0]/initiatingPartner[
0]/ GlobalBusinessIdentifier[0]
```

### Process Instance Identifier

The query to extract the Process Instance Identifier is:

```
/ServiceHeader[0]/ProcessControl[0]/ProcessIdentity[0]/InstanceIdentifier
[0]
```

### Focal Role

The query to extract the Focal Role is:

```
/ServiceHeader[0]/ProcessControl[0]/TransactionControl[0]/PartnerRoleRout
e[0]/toRole[0]/PartnerRoleDescription[0]/GlobalPartnerRoleClassificationC
ode[0]
```

# Running a Conversation Example

The following figure illustrates two distinct conversations: a conversation on the initiator's/buyer's side and a conversation on the fulfiller's/seller's side. The figure shows that an initiator and a fulfiller in a conversation both send Chem eStandards records to each other by sending the document to wm.ip.chem:receive. To send an internal business document from a back-end system to the webMethods Chem eStandards Module, the initiator and the fulfiller use wm.ip.cm:processDocument. To send an internal business document from the webMethods Chem eStandards Module to a back-end system, the initiator and the fulfiller use a site-written service. The figure uses the *OrderCreate* process as an example. For further explanation, see the sections that follow the figure.

**Running a PIP 3A4 Conversation**
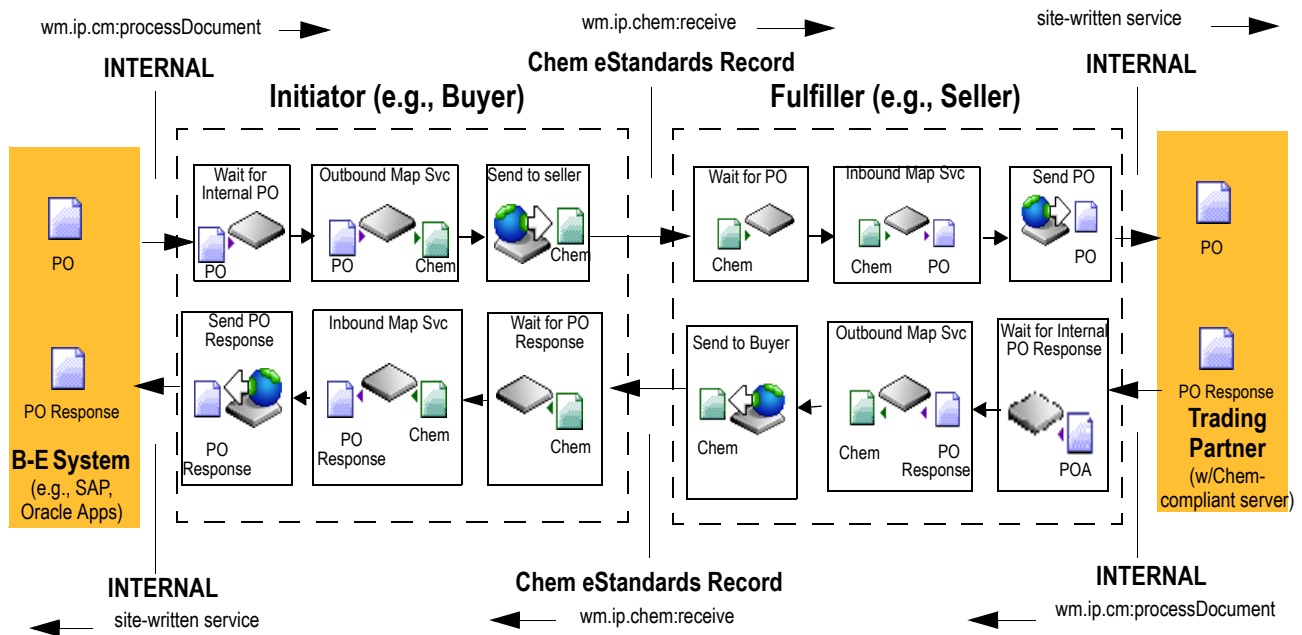


## Starting a Conversation

To start a conversation on the initiator's side, the initiator's back-end system calls wm.ip.cm:processDocument. To start a conversation on the fulfiller's side, a service invoked by the initiator's **Send to Seller** step sends the PO (as a Chem eStandards record) to wm.ip.chem:receive. Both wm.ip.cm:processDocument and wm.ip.chem:receive invoke wm.tn.doc:recognize to recognize the TN XML document type of the document received. On each side, a BizDocEnvelope is created. Both wm.ip.cm:processDocument and wm.ip.chem:receive send the BizDocEnvelope to the PRT.

On each side, the PRT checks to see if a ConversationID exists for the conversation. If no ConversationID exists, the webMethods Chem eStandards Module creates a

ConversationID. Then the PRT creates a new instance of the process model and assigns to the process model the ConversationID. The PRT executes the conversation based on the steps defined in the process model.

### Rejoining an Existing Conversation on the Fulfiller's Side

When the process model on the fulfiller's side has completed processing the PO, a service invoked by the fulfiller's **Send PO** step sends the PO to a back-end system. The back-end system must maintain the ConversationID associated with the PO. The back-end system processes the PO and sends an internal PO Response to rejoin the existing conversation by invoking wm.ip.cm:processDocument. Pass the internal PO Response and the stored ConversationID as inputs to this service. The PRT searches for a running conversation (or process instance) where a **Wait for PO Response** step is waiting for a PO Response with a matching ConversationID. When the PRT finds the correct conversation (or process), it sends the PO Response to rejoin the existing conversation.

### Rejoining an Existing Conversation on the Initiator's Side

To send the PO Response to the initiator, the fulfiller sends the business document (as a Chem eStandards record) to wm.ip.chem:receive. The wm.ip.chem:receive service invokes wm.tn.doc:recognize, which recognizes the business document and creates a BizDocEnvelope. The wm.ip.chem:receive service sends the BizDocEnvelope to the PRT. The PRT searches for a running conversation (or process instance) where a **Wait for PO Response** step is waiting for a PO Response with a matching ConversationID. When the PRT finds the correct conversation (or process), it sends the PO Response to rejoin the existing conversation.

If the PRT does not find a running conversation where a **Wait for PO Response** step is waiting for a PO Response with a matching ConversationID, the PRT ignores the PO Response.

## Why Monitor a Conversation?

You monitor a Chem eStandards conversation to track the state of a particular Chem eStandards process. For example, suppose that you initiate an *OrderCreate* process by sending a PO for 30 PCs to a trading partner. The next day, the trading partner calls, indicating that they did not receive your PO. In the webMethods Chem eStandards Module, you can view the activity of the conversation to determine what the current status is and what activity has occurred during the progress of the conversation. Following the same example, a reason why the trading partner did not receive the PO could be because the transaction encountered an error, such as a timeout error. By monitoring the conversation, you can determine if this was the cause for non-receipt of the PO. You can see whether your PO Request was resent, and how many times, and with what success. By viewing the current status of and the progress of a Chem eStandards

process, you can take appropriate action, which might include retrying the conversation, editing your trading partner profiles, or editing your process models.

# Sources of Conversation Status and Information

To monitor conversations, you have a number of sources of information from which you can draw to determine the status of your conversations and the activity of the various involved software entities. The following table lists and describes the primary sources from which you can draw information.

| Source | Description |
|---|---|
| webMethods Monitor | Use this tool to monitor and manage conversations. For more information about Monitor, see the next section, "About Monitor", and the *webMethods Monitor User's Guide*. |
| webMethods Trading Networks **Transaction Analysis** screen | Use this log to query and analyze results of documents that are sent or received by Trading Networks. For more information about the webMethods Trading Networks Transaction Analysis screen, see the *Building Your Trading Network* manual. |
| webMethods Integration Server error log | Use this log to retrieve server-related error and exception messages that occur during an invocation of a service directly or indirectly from a conversation. |

## About Monitor

Monitor is a web-based user interface that you can use to examine instances of your process models. Monitor displays information about instances of your process models by accessing data from the Process Logging Database. For information about the Process Logging Database, see "webMethods Chem eStandards Module Run-Time Architecture/Components" on page 26 in this guide.

Using Monitor, you can:

■ Search for a conversation by name, status, or date range.

■ Search specifically for a conversation that ended in error.

■ See a graphical overview of a conversation.

■ Examine information about a conversation and its execution; for example status of the conversation, status and iteration of process model steps, and so on.

■ View services used in the conversation.

■ Perform control tasks to affect the state of a conversation:

- Suspend and resume a conversation.

- Enable and disable a conversation.

- Start and stop a conversation.

- Edit and resubmit process model steps.

For more information about Monitor and for procedures about how to perform the aforementioned tasks, see the *webMethods Monitor User's Guide*.

# Archiving Chem eStandards Records to the File System

As a default, the webMethods Chem eStandards Module archives all business documents that it sends and receives to the Trading Networks database. In addition, you can configure the webMethods Chem eStandards Module to archive all business documents it receives to the file system. For example, you can configure the RNIF 1.1 transport protocol to archive either inbound Chem eStandards records, outbound Chem eStandards records, or both. You can configure the transport protocol to store the archived Chem eStandards records where you specify. You can retrieve these archived Chem eStandards records by going to the archive directory and opening the archived records in an editor of your choice.

Archiving Chem Standards records to the file system is an added level of archiving provided for debugging purposes. If performance is a concern, however, you should turn this level of archiving off.

**To archive inbound or outbound Chem eStandards records to the file system**

**1** Open the following file in the text editor of your choice:

webMethods6\IntegrationServer/packages/WmRNIF11TRP/config/chem11.cnf

where webMethods6\IntegrationServer is the directory in which your webMethods Integration Server is installed.

**2** Change the value of the *archiveInboundRNO?* parameter to **yes**.

Note that the default value is already set to `yes` so that the Chem eStandards records will be archived.

**3** Change the value of the *InboundFolder* parameter to the directory you want to contain your archived inbound Chem eStandards records.

Note that the default value is already set to `packages/WmRNIF11TRP/pub/archive`.

**4** Change the value of the *archiveOutboundRNO?* parameter to **yes**.

Note that the default value is already set to `yes` so that the Chem eStandards records will be archived.

**5**   Change the value of the *OutboundFolder* parameter to the directory you want to contain your archived outbound Chem eStandards records.

Note that the default value is already set to `packages/WmRNIF11TRP/pub/archive`.

> **Note:** Archive file names are composed of a series of parts, separated by an underscore. The first part of the name is the D-U-N-S$^{®}$ number of the trading partner who sent the archived Chem eStandards record to you or to whom you sent the record. The second part of the name is the Process Instance ID of the conversation to which the archived Chem eStandards record belonged. The third part of the name is either an *S* or an *R* to indicate whether the Chem eStandards record has been sent or received. The fourth part of the name is the Chem eStandards process number and version that was implemented by the conversation associated with the archived Chem eStandards record. The fifth part of the name is the Global Business Signal Code for the signal documents and Global Business Action Code for business documents. The last part of the name is the file extension *.rno*.
>
> For example, an archived message might have a file name like the following:
>
> *987654321_0a013218f70dc0a6000000e5_R_E41v2.0_ReceiptAcknowledgment.rno*

**6**   Save and close the configuration file.

From this point on, your Chem eStandards records will be archived according to what you specified in the configuration file. If you want at any time to turn off the archival feature, use this same procedure to set the *archiveInboundRNO?* and *archiveOutboundRNO?* values to *no*.

**7**   On the Server Administrator navigation panel, under the **Packages** heading, click **Management**.

**8**   Click **Reload** for the *WmRNIF11TRP* package.

# Handling Errors and Exceptions

# Handling Errors and Exceptions

The webMethods Chem eStandards Module provides handler services for various tasks; for example, handling errors during inbound validation of a Chem eStandards record or handling a Chem eStandards record that does not conform to any TN XML document type. Optionally, you can customize the provided handler services or create your own handler services to enhance the messages written to the various logs, generate an e-mail message upon an exception, write messages to a database, and so on.

## What Is a Handler Service?

A handler service is a service that a particular step invokes when an event in the Chem eStandards process occurs. For example, in the Two Action Initiator Model.xml process model template, the **Verify ACK** step invokes the wm.ip.cm.handlers:verifyAcknowledgement service when the initiator receives a Receipt Acknowledgement (ACK) from the fulfiller.

In webMethods Developer, you create handler services by creating a new service for each event that you want to handle in the Chem eStandards process.

## Handler Services Provided by the webMethods Chem eStandards Module

The following table lists and describes the handler services that the webMethods Chem eStandards Module provides in the WmIPRoot package and identifies whether you can edit a particular handler. For more information about any of these services, see the *webMethods RosettaNet Module Built-In Services Reference Guide*. You can find other handler services provided with the Chem eStandards sample as well as other built-in handler services provided in the WmRoot and WmPublic packages.

| Service | Description | Can Edit | Do Not Edit |
|---|---|---|---|
| wm.ip.cm.handlers:defaultHandler | The handler for generic situations. Causes a message to be written to the log. | ✔ | |
| wm.ip.cm.handlers:end | The last step in a conversation. | ✔ | |
| wm.ip.cm.handlers:error | The service called when an error occurs during the processing of a Chem eStandards conversation. | | ✔ |
| wm.ip.cm.handlers:inboundValidation | The service for validating inbound Chem eStandards records. | | ✔ |
| wm.ip.cm.handlers:processNOF | This service **does not** apply to the webMethods Chem eStandards Module. | | ✔ |

| Service | Description | Can Edit | Do Not Edit |
|---|---|---|---|
| wm.ip.cm.handlers:processUnhandledDocument | The service for handling a Chem eStandards record that does not conform to any TN XML document type. | | ✔ |
| wm.ip.cm.handlers:send | The service for sending Chem eStandards records to a trading partner via RNIF 1.1 transport protocol. | | ✔ |
| wm.ip.cm.handlers:sendReceiptACK | The service for sending the Receipt ACK upon receipt and validation of a Chem eStandards record. | | ✔ |
| wm.ip.cm.handlers:sendSyncResponse | The service for invoking the wm.ip.rnif11.trp:sendSyncResponse service that is provided in the WmRNIF11TRP package. | | ✔ |
| wm.ip.cm.handlers:start | The placeholder for the step in a conversation. | ✔ | |
| wm.ip.cm.handlers:startNOF | This service **does not** apply to the webMethods Chem eStandards Module. | | ✔ |
| wm.ip.cm.handlers:verifyAcknowledgement | The service for invoking the wm.ip.rnif11.trp:verifyAcknowledgement service that is provided in the WmRNIF11TRP package. | | ✔ |

**Note:** If you **must** edit a handler service that webMethods has recommended you not to edit, you should be an advanced, experienced developer or get an advanced, experienced developer to edit the handler service.

# When Might Errors or Exceptions Occur?

Errors might occur at any point in a Chem eStandards Module conversation. They might occur when:

■ A Chem eStandards record is created or sent.

■ A Chem eStandards record is unpacked into its components (preamble header, service header, etc.).

■ A Chem eStandards record is validated for structure and content.

■   A Chem eStandards process expires or exceeds the number of retries allowed.

■   A Chem eStandards record is received out of sequence.

## General Errors With Chem eStandards Records

The following table lists the general errors that might occur with a Chem eStandards record based on whether the record is inbound or outbound.

| Inbound/ Outbound | Error | How webMethods Chem eStandards Module Responds to the Error |
|---|---|---|
| Outbound | Validation fails. | The PRT ignores the Chem eStandards record. Does not send an exception. Sets the conversation to an error state. (This error can occur only if outbound validation is turned on.) |
| Outbound | Cannot package the Chem eStandards object. | The PRT ignores the Chem eStandards record. Does not send an exception. Sets the conversation to an error state. |
| Outbound | Error occurs during HTTP/HTTPS transmission. | The webMethods Chem eStandards Module retries the transmission the designated number of times. If the transmission fails on all attempts, the PRT places the conversation in an error state. |
| Inbound | Header validation fails. | The process run time (PRT) ignores the Chem eStandards record. Does not send an exception. |
| Inbound | Signature verification fails. | The PRT ignores the Chem eStandards record. Does not send an exception. |
| Inbound | Content validation fails. | The PRT places the conversation in an error state. Sends an exception to the trading partner with the reason for the exception in the text of the message. |

## Errors When a Transaction Expires or a Transaction Exceeds the Number of Retry Attempts

When a step, typically a wait step, in a conversation expires or times out and when a process exceeds the number of retry attempts, the process transitions to an error state. That is, the PRT, which manages the execution of conversation logic, sends the conversation to the **Error** step. The **Error** step invokes the service that you assigned to the step to handle the error.

For example, in the Two Action Initiator Model.xml process model template, if the initiator does not receive an External Response document within 24 hours of sending the External Request document, the process transitions to the error state. Similarly, if the initiator sends

an External Response document more than 4 times to the fulfiller without receiving a Response ACK, the process transitions to the error state.

### Errors When a Business Document Is Out of Sequence

A process model specifies steps exactly as they are to happen based on the results of the execution of each step. Some steps invoke services to create outbound business documents or to process inbound business documents. Other steps wait to receive business documents from trading partners.

If you receive a business document from a trading partner, but either you did not expect a business document (that is, the conversation is not waiting for a business document) or the business document that you received is not the business document that you expected (that is, you receive the wrong business document), the PRT handles the received business document as an out-of-sequence or unhandled business document. The arrival of an unhandled business document causes the PRT to place the conversation in an error state by sending the conversation to the **Error** step. The **Error** step invokes the wm.ip.cm.handlers:processUnhandledDocument service in the *WmIPRoot* package. This service generates a general exception and sends the exception to the trading partner who sent the unexpected business document.

An example might be when the process model specifies the business documents to send and receive in a certain order, but the order in which the conversation actually sends and receives the business documents becomes scrambled during the communications over the Internet. This could cause a business document to arrive before it is scheduled, according to the specified order.

For details on error handling, see "Handler Services Provided by the webMethods Chem eStandards Module" on page 96 in this guide and the *webMethods Chem eStandards Module Built-In Services Reference Guide*.

# Where to Find Information About Errors

You can view information about transactions and conversations that are in various error states by looking at the webMethods Trading Networks **Transaction Analysis** screen and webMethods Monitor. For more information about these resources, see "Sources of Conversation Status and Information" on page 92 in this guide.

For sample handler services for Receipt ACKs and exceptions, see the WmChemSample package.

# Index