webMethods Module for EDI Concepts Guide

Version 9.0

April 2014

This document applies to webMethods Module for EDI Version 9.0 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: ESTD-EDI-CG-90-20150625

Table of Contents

About this Guide 5
Document Titles 5
Document Conventions 6
Online Information 7

1. Overview of webMethods Module for EDI 9
Overview 10
Architecture 11
Supported EDI Standards 14

2. Processing Inbound EDI Documents 15
Overview 16
Before You Process EDI Documents 16
Flat File Schemas 16
Processing Inbound EDI Documents with Trading Networks 17
Creating an EDI Client 17
Defining Trading Partner Information 18
Using Standard or Non-Standard Processing 19
Comparing Standard vs Non-Standard Processing 19
EDI Document Inbound Processing 20
Initial Processing of the EDI Recognizer 21
Processing Documents in Trading Networks 26
Processing Rules for Inbound EDI Documents 29
Preprocessing Actions 29
Processing Actions 30
Creating EDI Trading Partner Agreements 32
Generating Functional Acknowledgments 32
Generating Interchange Acknowledgments 34

3. Forming EDI Documents and Sending Them Outbound 35
Overview 36
Creating a Service for Forming an Outbound EDI Document 37
Delivering the EDI Document Directly from the Service that Forms It 39
Submitting the Outbound EDI Document to Trading Networks to Deliver It 41
Routing the Outbound EDI Document to Trading Networks to Deliver It 43

4. Working with VANs 47
Overview 48
Inbound Processing: Retrieving Documents from VANs 48
Outbound Processing: Delivering Documents to VANs 49

5. Batching Outbound EDI Documents 53
Overview 54
Creating the Batched EDI Documents 55
Delivering the Batched Document 57

6. EDI Documents in Business Processes 59
Overview 60
How You Define the Business Process 60
Conversation IDs for EDI Documents 61
How EDI Documents are Passed to a Business Process 62

A. Using Module for EDI Decoupled from Trading Networks 65
Overview 66
Configuring Module for EDI Decoupled from Trading Networks 66
Processing of Inbound EDI Documents 66
Services to Process the Inbound EDI Document 67
Forming EDI Documents to Send Outbound 69

About this Guide

This guide is for users of the webMethods Module for EDI. It provides an overview of the Module for EDI and its features. It describes the functionality of the Module for EDI when you use it with webMethods Trading Networks and other webMethods components.

Document Titles

Some webMethods document titles have changed during product releases. The following table will help you locate the correct document for a release on the Software AG Documentation Web site or the Empower Product Support Web site.

Documentation	8.2	9.0 and later
Designer Process Development online help		
webMethods BPM Process Development Help	x	x
Designer Service Development online help		
webMethods Service Development Help	x	x
Integration Server administration guide		
webMethods Integration Server Administrator's Guide		x
Administering webMethods Integration Server	x	
Integration Server built-in services reference guide		
webMethods Integration Server Built-In Services Reference	x	x
Integration Server clustering guide		
webMethods Integration Server Clustering Guide	x	x
Integration Server publish-subscribe developer's guide		
Publish-Subscribe Developer's Guide	x	x
My webMethods administration guide		
Administering My webMethods Server	x	x
Optimize administration guide		
Administering webMethods Optimize	x	x
Optimize user's guide		
webMethods Optimize User's Guide		x

Documentation	8.2	9.0 and later
Optimizing BPM and System Resources with BAM: webMethods Optimize User's Guide	x	
Process Engine administration guide		
Administering webMethods Process Engine	x	x
Trading Networks administration guide		
webMethods Trading Networks Administrator's Guide		x
Building B2B Integrations: webMethods Trading Networks Administrator's Guide	x	
Trading Networks built-in services reference guide		
webMethods Trading Networks Built-In Services Reference	x	x
Trading Networks concepts guide		
webMethods Trading Networks Administrator's Guide and *webMethods Trading Networks User's Guide*		x
Understanding webMethods B2B: webMethods Trading Networks Concepts Guide	x	
Trading Networks user's guide		
webMethods Trading Networks User's Guide		x
Managing B2B Integrations: webMethods Trading Networks User's Guide	x	
webMethods installation guide		
Installing webMethods Products and *Using the Software AG Installer*	x	x
webMethods logging guide		
webMethods Audit Logging Guide	x	x
webMethods upgrade guide		
Upgrading webMethods Products	x	x

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies storage locations for services on webMethods Integration Server, using the convention *folder.subfolder:service*.

Convention	Description
UPPERCASE	Identifies keyboard keys. Keys you must press simultaneously are joined with a plus sign (+).
Italic	Identifies variables for which you must supply values specific to your own situation or environment. Identifies new terms the first time they occur in the text.
`Monospace font`	Identifies text you must type or messages displayed by the system.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at http://documentation.softwareag.com. The site requires Empower credentials. If you do not have Empower credentials, you must use the TECHcommunity website.

Software AG Empower Product Support Website

You can find product information on the Software AG Empower Product Support website at https://empower.softwareag.com.

To submit feature/enhancement requests, get information about product availability, and download products, go to Products.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the Knowledge Center.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at http://techcommunity.softwareag.com. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.

- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

1 Overview of webMethods Module for EDI

- Overview 10
- Architecture 11
- Supported EDI Standards 14

Overview

webMethods Module for EDI (Module for EDI) enables business partners to exchange EDI documents within and across the extended enterprise, providing comprehensive EDI support as a key part of the webMethods total business process automation solution.

Module for EDI provides services and transformation management tools that enable you to recognize, transform, convert, validate, and map multi-transactional EDI documents in real time or in batch.

When you use Module for EDI with webMethods Trading Networks (Trading Networks), you use the features of Trading Networks to exchange EDI documents with your trading partners.

Important! Module 9.0 for EDI requires Trading Networks to be installed. For complete instructions on installing, configuring, and completing the installation of Trading Networks, see the webMethods installation guide for your release. See "About this Guide" for specific document titles. For more information on installing Module for EDI, see *webMethods Module for EDI Installation and User's Guide*.

When you use Module for EDI along with other webMethods components, you can extend its capabilities. The following table shows the capabilities of the module based on the components you use:

Using Module for EDI with:	Provides the functionality to...
webMethods Trading Networks (Required component)	■ Maintain information about your trading partners that exchange EDI documents. ■ Send EDI documents to and retrieve EDI documents from trading partners. ■ Connect to Value Added Networks (VANs) to pick up and deliver EDI documents. ■ Process EDI documents using the features of Trading Networks (for example, using processing rules). ■ Batch the sending of EDI documents rather than sending them to their destinations in real time, as each is received. ■ Automatically generate functional acknowledgments (FA) for inbound documents. ■ View EDI documents that have passed through your system.

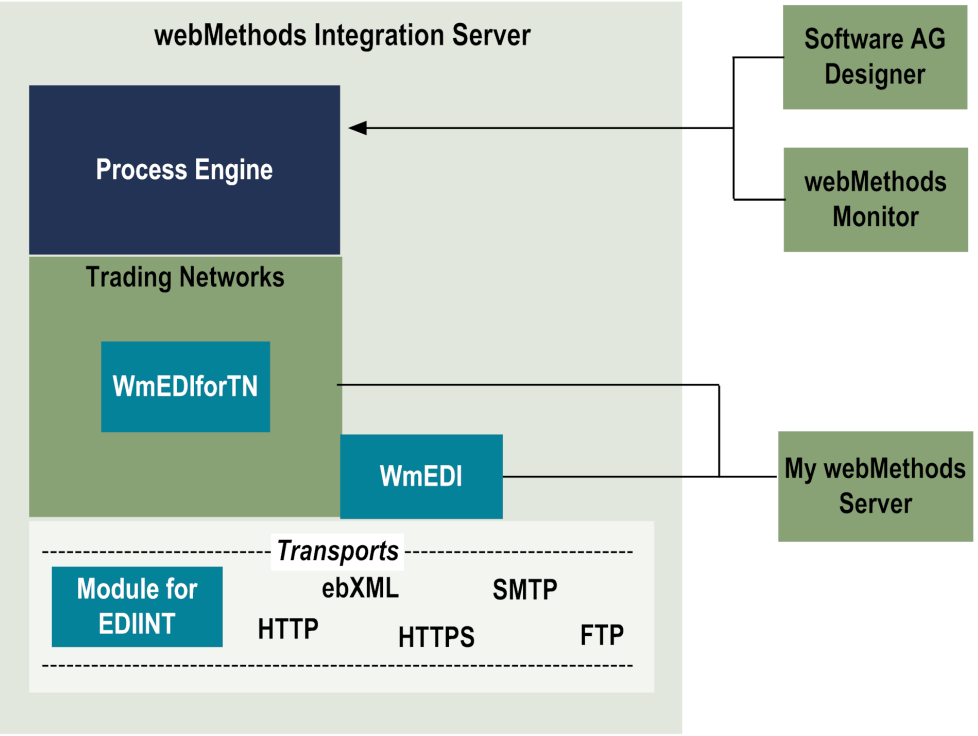
Using Module for EDI with:	Provides the functionality to...
My webMethods Server and its user interface, My webMethods	■ Configure webMethods Module for EDI. ■ Submit documents to Trading Networks. ■ Manage control numbers. ■ Manage Trading Networks profiles. ■ View the Executive Dashboard. ■ View transaction summary reports.
Software AG Designer	■ Model and manage business processes.
webMethods Monitor	■ Monitor the progress of business processes.

Architecture

When you install Module for EDI, two packages are installed into webMethods Integration Server: the WmEDI package and the WmEDIforTN package.

Note: The WmEDIsamples package contains sample EDI flow services, mappings, and IS document types that demonstrate how to use Module for EDI and Software AG Designer to execute typical EDI processing scenarios. This package is located in the Technical community area of the Empower Product Support website at https://communities.softwareag.com/. EDI developers can use the WmEDIsamples as a reference. Before going into production, you should delete the WmEDIsamples package.

The following diagram illustrates how Module for EDI fits into the webMethods architecture. For more information, see the text after the diagram.



Component	Description
webMethods Integration Server	The underlying foundation of webMethods components.
Trading Networks	A webMethods component that enables your enterprise to link with other companies (buyers, suppliers, strategic partners) and marketplaces to form a business-to-business trading network. You must install Trading Networks before installing Module for EDI. For more information, see the Trading Networks concepts guide, Trading Networks user's guide, and Trading Networks administration guide for your release. See "About this Guide" for specific document titles.

Component	Description
Module for EDI	A production environment is comprised of the following two packages: ■ The WmEDI package is the basic functionality that provides support for the EDI standard to the webMethods components. ■ The WmEDIforTN package allows for the interaction between the WmEDI package and Trading Networks. This interaction allows you to use Trading Networks as a gateway for EDI document exchange. Module for EDI uses the functionality of Trading Networks to provide additional features, such as support for VANs, reconciling FAs, and batching the sending of EDI documents.
Software AG Designer	A design-time tool that you can use to create process models that define how to include EDI documents in business processes (also called conversations). After you design the process models, you build and upload them to create the run-time elements (for example, flow services and triggers) that reside in webMethods Integration Server. Process Engine of webMethods Integration Server executes the business processes (conversations) at run time. To include EDI documents in business processes, you must use Trading Networks. At run time, after Trading Networks performs its processing, it can pass documents to Process Engine to perform the logic that you designed in a process model. For more information about designing process models, see Designer online help.
webMethods Monitor	Allows you to monitor the progress and status of the business processes (conversations) involving EDI documents. webMethods Monitor interacts with Process Engine to obtain the status information.

Component	Description
My webMethods Server	A web-based, monitoring and administration user interface for managing your webMethods components. You use My webMethods Server (and its user interface, My webMethods) with Module for EDI to: ■ View and edit the configuration settings for the WmEDI, WmTN, and WmEDIINT packages ■ View and edit the EDI format.xml file ■ Submit EDI and XML docs to Trading Networks ■ Update document type attributes ■ Manage control numbers For more information about managing Module for EDI using My webMethods, see *webMethods Module for EDI Installation and User's Guide*.

Supported EDI Standards

The table lists the EDI standards supported by Module for EDI.

EDI Standard	Version
EANCOM	2, 3, 01B, 93A, 96A
ODETTE	3, 94
TRADACOMS	2, 3, 4, 5, 6, 8, 9
UCS	4010, 4020, 4030, 5010
UNEDIFACT	00A, 00B, 01A, 01B, 01C, 02A, 02B, 03A, 03B, 04A, 04B, 05A, 05B, 06A, 06B, 07A, 07B, 08A, 08B, 09A,09B, 10A, 10B, 11A, 3, 40, 41, 901, 902, 911, 912, 921, 93A, 94A, 94B, 95A, 95B, 96A, 96B, 97A, 97B, 98A, 98B, 99A, 99B, S93A, 2932
VDA	4905_4, 4906_2, 4907_2, 4908_3, 4911_1, 4913_4, 4913_5, 4915_2, 4916_1, 4918_1, 4919_1, 4920_1, 4921_1, 4927_3
VICS	3010, 4010, 4020, 4030, 4050, 5010
X12	2000, 2001, 2001FORD, 2002, 2002FORD, 2003, 2003GM, 2040, 2040CHRY, 3010, 3020, 3030, 3040, 3041, 3050, 3051, 3060, 3070, 4010, 4010RIFMAT, 4020, 4030, 4040, 4050, 4060, 5010, 5020, 5030, 5040, 5050, 6010, 6020

2 Processing Inbound EDI Documents

- Overview 16
- Processing Inbound EDI Documents with Trading Networks 17
- Initial Processing of the EDI Recognizer 21
- Processing Documents in Trading Networks 26
- Processing Rules for Inbound EDI Documents 29
- Creating EDI Trading Partner Agreements 32
- Generating Functional Acknowledgments 32
- Generating Interchange Acknowledgments 34

Overview

webMethods Module for EDI (Module for EDI) provides a toolkit of built-in services that you use as the building blocks for creating your own EDI solution.

There are two aspects to your EDI solution:

- How to send EDI documents to webMethods Trading Networks and how you process them; that is, inbound processing
- How you form EDI documents from internal-format documents (for example, documents from back-end systems) and send the EDI documents outbound; that is, outbound processing

Before You Process EDI Documents

The following table lists the tasks you must complete before you can process EDI documents with webMethods Trading Networks (Trading Networks). For more information, see *webMethods Module for EDI Installation and User's Guide*.

- Define Module for EDI properties
- Configure how you want your format services to convert field values in documents from EDI format to internal format and vice versa
- Specify how to associate your format services to fields defined in a flat file schema for an EDI document
- Create flat file schemas for EDI documents and your internal format documents. For more information about Flat File Schemas, see "Flat File Schemas" on page 16
- Install TN document types that correspond to the types of EDI documents you want to process
- Define profiles for the partners with whom you want to exchange EDI documents
- Define EDITPAs that contain EDI-specific information
- Define settings for inbound control number validation
- Define processing rules that specify how to process documents
- Define public queues to hold EDI documents

Note: For more information about preparing Module for EDI to use EDIINT to transport your EDI documents, see *webMethods Module for EDIINT Installation and User's Guide*.

Flat File Schemas

A *flat file schema* is the blueprint that contains the instructions for parsing or creating a flat file. It resides as a namespace element in webMethods Integration Server. This blueprint details the structure of the document.

Module for EDI uses flat file schemas to parse and validate the structure of inbound EDI documents and to convert documents from EDI format into the format used by internal applications and vice versa.

You can create two kinds of flat file schemas:

- Schemas that define the structure of EDI documents

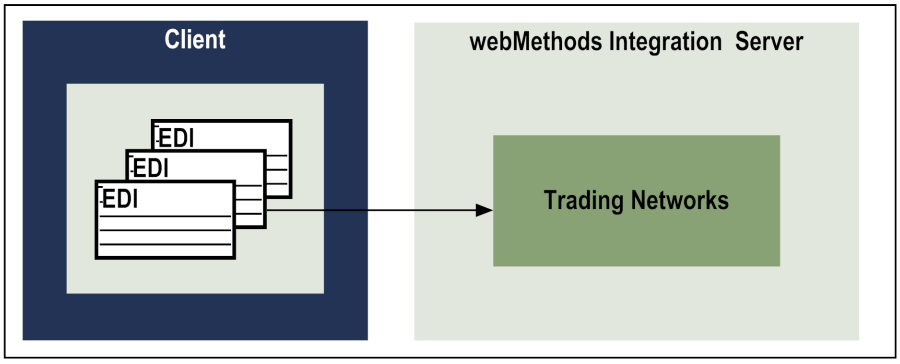
  Module for EDI automatically installs these flat file schemas for you when you install TN document types.

- Schemas that define the structure of your internal-format documents (for example, documents required for back-end systems)

  The service you create to process an Interchange, Group, or Transaction document might create an internal-format document (for example, the format required by a back-end system). Module for EDI provides a service that you can invoke to create the internal-format document based on the flat file schema. Use Software AG Designer to create the flat file schema. For more information, see *Flat File Schema Developer's Guide*.

Processing Inbound EDI Documents with Trading Networks

For inbound processing, you create clients that send EDI documents to Trading Networks and you set up information in Trading Networks to process the documents.



Creating an EDI Client

To send documents to Integration Server to be processed through Trading Networks, you create a client. The client can use one of the following transports to send the EDI documents:

- HTTP or HTTPS
- FTP
- File Polling

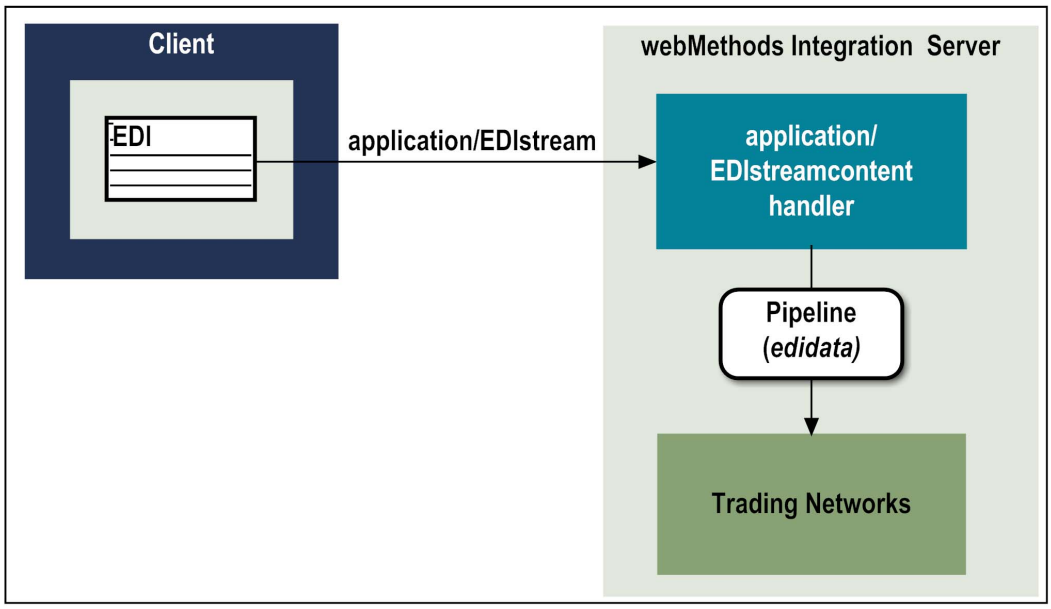
File Polling is a feature of Integration Server that allows your client to drop the document into a directory that Integration Server monitors. When documents arrive in the directory, Module for EDI can process them.

- EDIINT AS1, EDIINT AS2, or EDIINT AS3

  If you want to use EDIINT to transport your EDI documents, use Module for EDIINT and Trading Networks. For more information, see *webMethods Module for EDIINT Installation and User's Guide*.

The rest of this section describes clients that use HTTP, HTTPS, FTP, or File Polling.

When your client sends the EDI document to Integration Server, it must associate the document with a content type that Module for EDI recognizes, for example, `application/EDIstream`. When Integration Server receives a document that has an EDI content type, it passes the document to the appropriate EDI content handler. The EDI content handler performs initial processing on the document, which includes creating the pipeline with the *edidata* variable that contains the EDI document. After performing the initial processing, the EDI content handler invokes the service that the client specifies, that is, the wm.tn:receive service, to have Trading Networks process the document.



For more information about how to create the client, see *webMethods Module for EDI Installation and User's Guide*.

Defining Trading Partner Information

To process documents when you are using webMethods Trading Networks with webMethods Module for EDI, you must define both trading partner profiles and trading partner agreements (EDITPAs) for the partners with whom you will trade EDI documents. How you define information for your partners depends on whether you want Module for EDI to use standard or non-standard processing.

Using Standard or Non-Standard Processing

When Module for EDI processes an EDI document, it processes one interchange of the document at a time. An interchange can contain groups and transaction sets. The module can process the interchange, groups, and transaction sets using either standard or non-standard processing.

- For standard processing

  Module for EDI processes the interchange, all its groups, and all its transaction sets using settings that you define for the interchange sender/receiver pair. To set up standard processing, you use standard Trading Networks objects.

- For non-standard processing

  You can specify different processing settings for each group within the interchange. Module for EDI processes the groups and the transaction sets within each group using the settings you define for the group sender/receiver pairs. To set up non-standard processing, in addition to using Trading Networks objects, you must define information from the module's home page, which includes interchange sender/receiver pair information.

Comparing Standard vs Non-Standard Processing

The table below lists the advantages and disadvantages of using standard and non-standard processing:

Processing Type	Advantages	Disadvantages
Standard Processing	Information you define is in profiles and EDITPAs, so management of all information can be done through My webMethods. Note: This approach is recommended.	You must define profiles for all partners (senders/receivers) at both the interchange and group level.

Processing Type	Advantages	Disadvantages
Non-Standard Processing	You only need to define profiles for partners (senders/receivers) at the group level through My webMethods. You do not need to define profiles for partners at the interchange level; therefore, there are fewer profiles to set up and maintain.	■ You must maintain information for interchange level sender/receiver pairs through the Module for EDI home page, as well as maintain EDITPAs through My webMethods. ■ You must maintain information about the sender/receiver pairs at the group level that are associated with each interchange level. This information is maintained through the Module for EDI home page. ■ When you save a copy of the interchange-level document in the Trading Networks database, the document will be saved with the sender and receiver set to unknown. Because you do not create profiles for senders and receivers at the interchange level, Trading Networks is unable to determine the sender and receiver. ■ You cannot use the batching feature of Module for EDI.

For more information about defining trading partner information, see *webMethods Module for EDI Installation and User's Guide*.

EDI Document Inbound Processing

There are three steps in processing an inbound EDI document:

- "Initial Processing of the EDI Recognizer" on page 21
- "Processing Documents in Trading Networks" on page 26
- "Processing Rules for Inbound EDI Documents" on page 29

Trading Networks processes ANSI X12 and UN/EDIFACT documents identically. Though VDA and TRADACOMS documents are structured differently than other documents, they are roughly analogous to other document types and, so, are processed similarly:

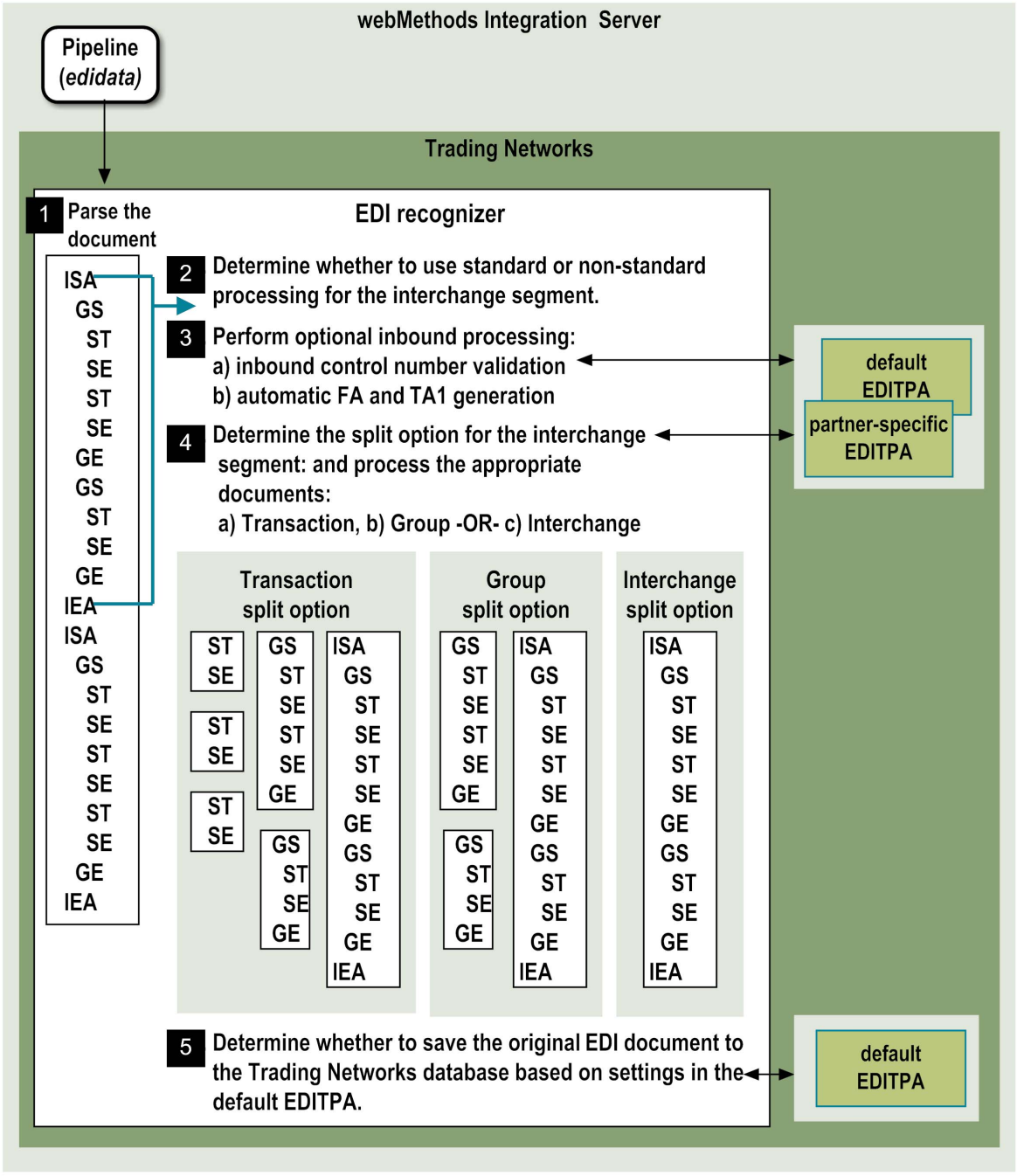
ANSI X12 and UN/EDIFACT document type	VDA document type	Analogous TRADACOMS document type
Interchange	Envelope	Transmission
Transaction set	Envelope Note: Only one VDA message is present in a transaction.	File
Group	No equivalent	Batch
Functional Acknowledgment (FA)	No equivalent	No equivalent

Important! Although Trading Networks can process documents of any supported EDI standard, it cannot properly process a mixture of TRADACOMS and non-TRADACOMS documents in a single submission. If the first inbound document is a TRADACOMS document, Trading Networks considers any subsequent non-TRADACOMS documents to be of the Unknown document type. Similarly, if the first inbound document is a non-TRADACOMS document, Trading Networks considers any subsequent TRADACOMS documents to be of the Unknown document type.

Initial Processing of the EDI Recognizer

When you install Module for EDI, the WmEDIforTN package of the module enhances the capabilities of Trading Networks to allow it to recognize and to begin processing EDI documents. The WmEDIforTN package adds an *EDI recognizer* to Trading Networks.

The following diagram illustrates the initial processing that the EDI recognizer performs on an ANSI X12 document. If you use the TRADACOMS standard, as you view this diagram remember that a reference to an *interchange* is a reference to a TRADACOMS transmission, a *transaction set* is a TRADACOMS file, and a *group* is a TRADACOMS batch.


webMethods Integration Server
Pipeline
(edidata)
Trading Networks
EDI recognizer
1 Parse the document
ISA
GS
ST
SE
ST
SE
GE
GS
ST
SE
GE
IEA
ISA
GS
ST
SE
ST
SE
ST
SE
GE
IEA
2 Determine whether to use standard or non-standard processing for the interchange segment.
3 Perform optional inbound processing:
a) inbound control number validation
b) automatic FA and TA1 generation
4 Determine the split option for the interchange segment: and process the appropriate documents:
a) Transaction, b) Group -OR- c) Interchange
default EDITPA
partner-specific EDITPA
Transaction split option
Group split option
Interchange split option
5 Determine whether to save the original EDI document to the Trading Networks database based on settings in the default EDITPA.
default EDITPA

Step	Description
1	When Trading Networks receives an EDI document, it passes the EDI document to the EDI recognizer. The EDI recognizer parses the inbound EDI document, so subsequent processing can process the following types of documents: ■ For ANSI X12 and UN/EDIFACT documents: ■ *Transaction documents* that each contain a single transaction set ■ *Group documents* that contain a single group segment along with its transaction sets ■ *Interchange documents* that contain a single interchange envelope along with its group segments and transaction sets ■ For VDA documents: ■ *Envelope documents* that each contain a single VDA message ■ For TRADACOMS documents: ■ *File documents* that each contain multiple TRADACOMS messages ■ *Batch documents* that contain a single TRADACOMS batch segment along with its files ■ *Transmission documents* that each contain a single TRADACOMS transmission envelope along with its batch segments and files When parsing, the EDI recognizer also performs envelope validation and compliance check and places the results in the *errors* variable in the pipeline.
2	For each interchange segment in the original EDI document, the EDI recognizer determines whether to use standard or non-standard processing for the interchange segment. The type of processing (standard or non-standard) determines which EDI trading partner agreement (*EDITPA*) the EDI recognizer retrieves to obtain settings that tailor how to process documents from the interchange segment. For more information about EDITPAs, see "Creating EDI Trading Partner Agreements" on page 32. Note: Non-standard processing is not applicable with TRADACOMS or VDA documents.

Step	Description
	■ For standard processing, Module for EDI uses all standard Trading Networks objects for processing. The EDI recognizer obtains profiles for the sender and receiver identified in the interchange header (or TRADACOMS STX segment). Then it retrieves the EDITPA for the sender/receiver pair specified in the interchange header or TRADACOMS STX segment. When you use standard processing, all documents (for example, Transaction, Group, and Interchange) within the interchange segment or TRADACOMS STX segment are processed using the settings you defined for the sender/receiver specified in the interchange header or TRADACOMS STX segment. ■ For non-standard processing (which is not applicable for TRADACOMS or VDA documents), Module for EDI uses custom interchange sender/receiver pair information that you must define. This interchange sender/receiver pair information is stored in the Trading Networks database but is not associated with a standard Trading Networks object. The EDI recognizer uses the interchange sender/receiver pair information to determine the EDI ID qualifiers to use with the senders and receivers on the group headers within the interchange segment. Then for each group in the interchange segment, the EDI recognizer obtains the profiles for the sender and receiver identified in the group header. It then retrieves the EDITPA that corresponds to the sender/receiver specified in the group header. When you use non-standard processing, Transaction and Group documents are processed using the settings you defined for the sender/receiver specified in the group header. Note: Non-standard processing is deprecated. If you want to use non-standard processing, however, you must use the WmEDIforTN package home page to define interchange sender/receiver pair information. The EDI recognizer determines whether to perform standard or non-standard processing based on the existence of this interchange sender/receiver pair information. That is, if the sender/receiver pair information exists for the sender/receiver of the interchange segment, the Module for EDI performs non-standard processing. If it does not exist, the module performs standard processing.

Step	Description
3	Performs optional inbound processing based on how you set up Module for EDI. The module can perform: ■ Inbound control number validation for control numbers in the: ■ Interchange and/or group headers of the inbound ANSI X12 or UN/EDIFACT document ■ Envelope headers of the inbound VDA document ■ Transmission and/or batch of the inbound TRADACOMS document Note: The term *control number* is used in the EDI standards ANSI X12, UN/EDIFACT, and VDA. It refers to a number in the header of an EDI document that is used for validation and for the ordering of documents exchanged between trading partners. If you use the TRADACOMS EDI standard, the term *control number* is equivalent to the transmission reference numbers specified in the STX and BAT segments of your TRADACOMS documents. Whether your EDI standard includes control numbers or transmission reference numbers, you define them in Trading Networks in the same way; the only difference is in the terminology. In other words, Trading Networks and Module for EDI use the term *control number* to mean either control number or transmission reference number. ■ Automatic generation of functional acknowledgments (FAs). ■ When automatically generating an FA for an ANSI X12 document, Module for EDI generates an FA for each group in the inbound document. ■ When automatically generating an FA for a UN/EDIFACT document, the module generates an FA for each interchange in the inbound EDI document. FAs are not applicable to the TRADACOMS or VDA EDI standards. ■ Automatic generation of interchange acknowledgments (TA1s) for X12 interchanges.

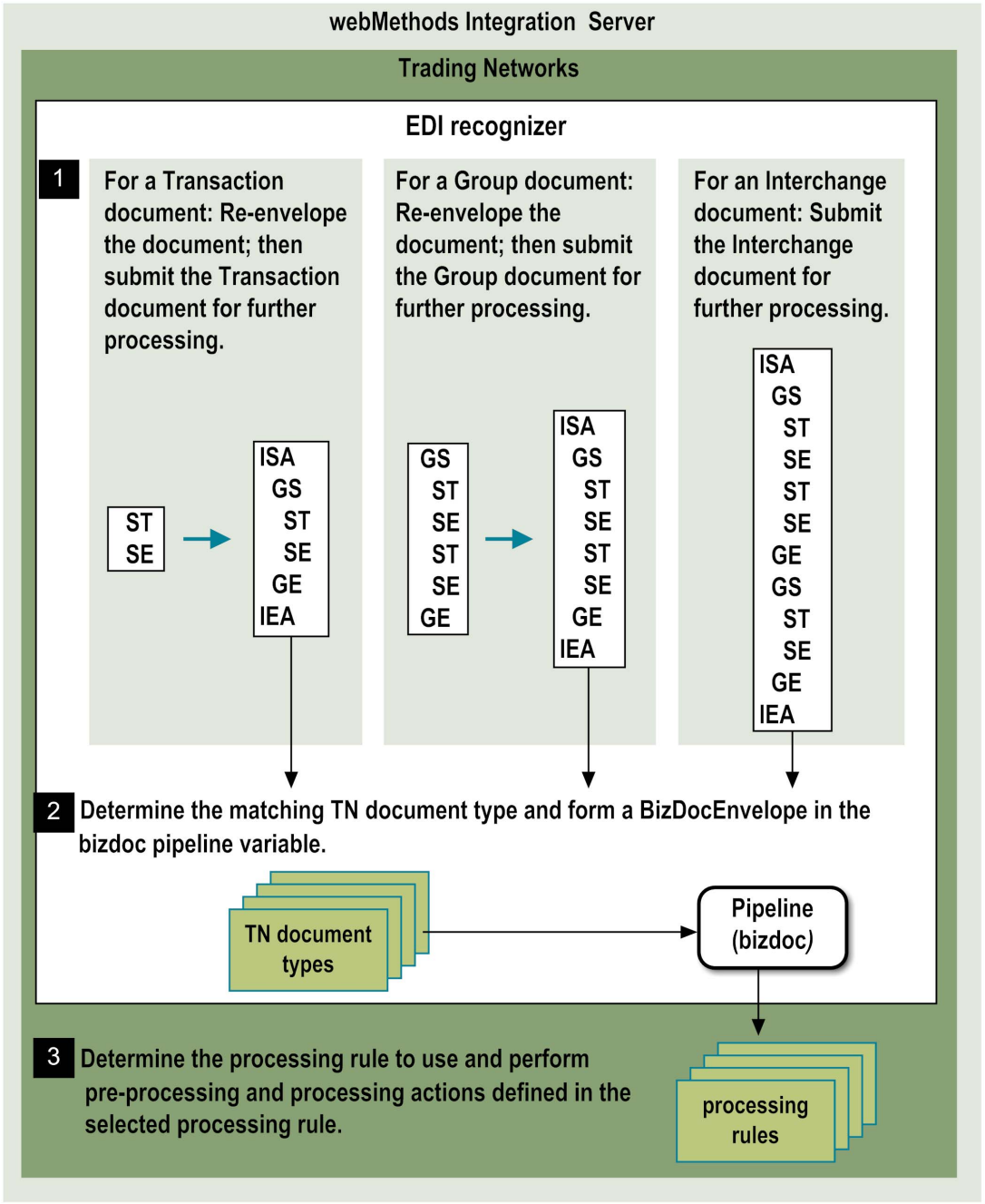
Step	Description
4	Using the EDITPA that it retrieved, the EDI recognizer determines the types of documents that you want to process using the split option variable within the EDITPA. You can set it to one of the following: ■ For ANSI X12 and UN/EDIFACT documents, set it to: ■ *Transaction split option* to have the EDI recognizer submit all Transaction, Group, and Interchange documents to Trading Networks for processing. ■ *Group split option* to have the EDI recognizer submit all Group and Interchange documents to Trading Networks for processing. ■ *Interchange split option* to have the EDI recognizer submit only Interchange documents to Trading Networks for processing. You cannot use this setting when using non-standard processing. ■ For TRADACOMS documents, set it to: ■ *File split option* to have the EDI recognizer submit all File, Batch, and Transmission documents to Trading Networks for processing. ■ *Batch split option* to have the EDI recognizer submit all Batch and Transmission documents to Trading Networks for processing. ■ *Transmission split option* to have the EDI recognizer submit only Transmission documents to Trading Networks for processing. ■ For VDA documents, the split option is not used. Regardless of the split option setting, the EDI recognizer will submit the VDA envelope containing a single VDA message to Trading Networks for processing. For information about processing each type of document that results from the split of the interchange segment, see "Processing Documents in Trading Networks" on page 26.
5	If the original EDI document contains multiple interchanges or TRADACOMS transmissions, after splitting documents and sending them for processing, the EDI recognizer can save the entire original EDI document to the Trading Networks database. It determines whether to save the original EDI document based on a setting that you define in the default EDITPA.

For more information, see *webMethods Module for EDI Installation and User's Guide*.

Processing Documents in Trading Networks

After determining the split option from the EDITPA, the EDI recognizer creates the appropriate documents (for example, Transaction, Group, and Interchange) based on the split option and submits each document individually to Trading Networks for processing.

The following diagram illustrates the processing for each type of ANSI X12 document. Again, if you use the TRADACOMS standard, as you view this diagram remember that a reference to an *interchange* is a reference to a TRADACOMS transmission, a *transaction set* is a TRADACOMS file, and a *group* is a TRADACOMS batch.



Step	Description
1	The EDI recognizer does the following: ■ For an ANSI X12 or UN/EDIFACT document: ■ *When the split option is Transaction*, the EDI recognizer forms a Transaction document for each transaction set within an interchange segment. To form a Transaction document, the EDI recognizer re-envelopes a transaction set with both: ■ A group envelope using the group header from the group segment in which the transaction set resided in the original EDI document AND ■ An interchange envelope using the interchange header from the interchange segment in which the transaction set resided in the original EDI document ■ *When the split option is Group*, the EDI recognizer forms a Group document for each group segment within an interchange segment. To form a Group document, the EDI recognizer re-envelopes the group segment with an interchange envelope using the interchange header from the interchange segment in which the group resided in the original EDI document. ■ *When the split option is Interchange,* the EDI recognizer forms an Interchange document comprised of the information from the interchange segment in the original EDI document. ■ For a TRADACOMS document: ■ *When the split option is File*, the EDI recognizer forms a Transmission document, a Batch document (if present), and one File document for the file within the transmission. ■ *When the split option is Batch*, the EDI recognizer forms a Transmission document and a Batch document. ■ *When the split option is Transmission*, the EDI recognizer forms only a Transmission document. ■ For a VDA document, the split option is not used. Regardless of the split option setting, the EDI recognizer forms only one Trading Networks document, because a VDA document contains a single VDA message.

Step	Description
2	After the EDI recognizer forms each document, it resubmits the newly formed document (for example, Transaction, Group, or Interchange) to Trading Networks for processing. Because the newly formed document is an EDI document, Trading Networks passes the document to the EDI recognizer for processing. The EDI recognizer does not parse the newly formed document again. The parsed version of the original EDI document remains in the pipeline. The EDI recognizer uses the newly formed EDI document only to recognize what type of document it is, to add the document as a content part to the BizDocEnvelope, and to save it to the Trading Networks database. The EDI recognizer uses the TN document types to determine the type of document. Module for EDI provides the TN document types for EDI documents. You install the TN document types for the types of EDI documents that you will want to process. For example, if you want to process the 4010 version of the ANSI X12 850 EDI document, you would install the X12 4010 850, X12 Group, and X12 Envelope TN document types into Trading Networks. After recognizing the type of document using TN document types, the EDI recognizer forms a BizDocEnvelope for the EDI document. The BizDocEnvelope is in the *bizdoc* pipeline variable. A BizDocEnvelope contains the original document (for example, Transaction, Group, or Interchange) and includes additional information that Trading Networks requires for routing and processing the document. In other words, the BizDocEnvelope represents a routable Trading Networks transaction.
3	After forming the BizDocEnvelope, Trading Networks determines the processing rule to use to process the document and executes the processing rule. You create processing rules to define the processing you want performed on each type of document. For example, you can define a processing rule that executes a service that you create to form an internal-format document based on information in the EDI document, and send that internal-format document to your back-end system. For more information, see "Processing Rules for Inbound EDI Documents" on page 29.

Processing Rules for Inbound EDI Documents

Processing rules specify two categories of actions: preprocessing and processing.

Preprocessing Actions

You define default preprocessing actions in the TN document types for EDI documents. However, in the processing rule, you can override the default settings that are defined in the TN document type. If a preprocessing action fails, Trading Networks records the error and continues processing.

You can use the following preprocessing actions for EDI documents:

- Validate Structure. Validate the structure of the EDI document against your flat file schema. By default, the TN document types for EDI documents indicate that validation should not be performed.
- Check for Duplicate Document. Determine whether the document is unique; that is, has Trading Networks previously received the document for processing? By default, the TN document types for EDI documents indicate that this check should not be performed.

  Module for EDI does not add module-specific logic to Trading Networks to perform this preprocessing action. When you use this preprocessing action, Trading Networks performs its standard logic for the function. For more information about preprocessing actions, see the Trading Networks concepts guide and the Trading Networks administration guide for your release. See “About this Guide” for specific document titles.
- Save Document to Database. Specify whether you want Trading Networks to save the document to its database. By default, the TN document types for EDI documents indicate that the content, attributes, and activity log information for EDI documents should be saved to the database.

  Module for EDI does not add module-specific logic to Trading Networks to perform this preprocessing action. When you use this preprocessing action, Trading Networks performs its standard logic for the function. For more information, about preprocessing actions, see the Trading Networks concepts guide and the Trading Networks administration guide for your release. See “About this Guide” for specific document titles.

Note: You cannot use the Verify Digital Signature preprocessing action for EDI documents. This preprocessing action requires that values for the system attributes SignedBody and Signature be available to use to verify the signature. Values for these system attributes are not set for EDI documents.

Processing Actions

You can use all of the Trading Networks processing actions for EDI documents. For inbound EDI documents, typically you will use the Execute a Service action to process the inbound document. You create the service that the processing action invokes. The logic you add to the service depends on the split option and the type of document (for example, Transaction, Group, or Interchange), as follows:

- When the split option is Transaction (or File), the EDI recognizer sends all three types of documents to Trading Networks for processing. You will create a processing rule for each type of document. The logic you add to the service for the Execute a Service action is based on the type of document:
  - Transaction (or File) document. This document contains a single transaction set or multiple TRADACOMS messages. For this level of document, add the logic that you want to perform against the transaction set or the messages. For example, you might convert the transaction set data to an IData object so you can map information from the transaction set to an internal-format document. Then the service could send the internal-format document to a back-end system.
  - Group (or Batch) document. If you want to generate a functional acknowledgment (FA) for an ANSI X12 or UN/EDIFACT Group document, the logic in the service might generate the FA and send the FA back to the sender.
  - Interchange (or Transmission) document. You might set up the processing rule to ignore this document (that is, perform no processing action) because you have completed the processing in the other documents.
- When the split option is Group (or Batch), the EDI recognizer sends Group and Interchange documents (or Batch and File documents) to Trading Networks for processing. You will create a processing rule for each type of document and specify processing actions based on the type of document:
  - Group (or Batch) document. In the logic you add to the service for the Execute a Service action, you might generate an FA and send the FA to the sender. Then, you might loop through each of the transaction sets in the group segment and perform processing against each. For example, you might convert the transaction set data to an IData object so you can map information from the transaction set to the inputs of another service, then invoke the service.
  - Interchange (or Transmission) document. You might set up the processing rule to ignore this document (that is, perform no processing action) because you have completed the processing in the logic for the Group or Batch document.
- When the split option is Interchange (or Transmission), the EDI recognizer sends only an Interchange or Transmission document to Trading Networks for processing. You will create a processing rule for the Interchange or Transmission document. In the logic you add to the service for the Execute a Service action, you might generate an FA and send the FA to the sender. Then, you might loop through document to access the information in the transaction sets, so you can perform processing against each.

For more information about defining processing rules, see *webMethods Module for EDI Installation and User's Guide*.

Creating EDI Trading Partner Agreements

An EDI Trading Partner Agreement (EDITPA) is a set of variables that you specify to tailor how the module processes documents that are exchanged between two trading partners. For example, an EDITPA contains the *split option* variable, which indicates what level of documents (for example, Transaction, Group, or Interchange) you want to process.

Module for EDI supports partner-specific EDITPAs and a single default EDITPA.

- A partner-specific EDITPA has a specific sender and receiver associated with it. It is specific to which partner represents the sender and which partner represents the receiver; therefore, you might have two EDITPAs for one trading partner pair. For example, for trading partners A and B, you might have one EDITPA where trading partner A is the sender and B is the receiver and another for when B is the sender and A is the receiver.

  A partner-specific EDITPA contains partner-specific variables used by only the particular pair of trading partners (sender and receiver) that are defined in the EDITPA.

- A default EDITPA has a sender and receiver set to "unknown." It contains variables used by all trading partners when partner-specific information is not available. That is, the module uses the values in the default EDITPA if a partner-specific EDITPA does not exist or if the value in the partner-specific EDITPA is null or empty.

For more information about setting up the default and partner-specific EDITPAs and defining EDITPAs variables, see *webMethods Module for EDI Installation and User's Guide*.

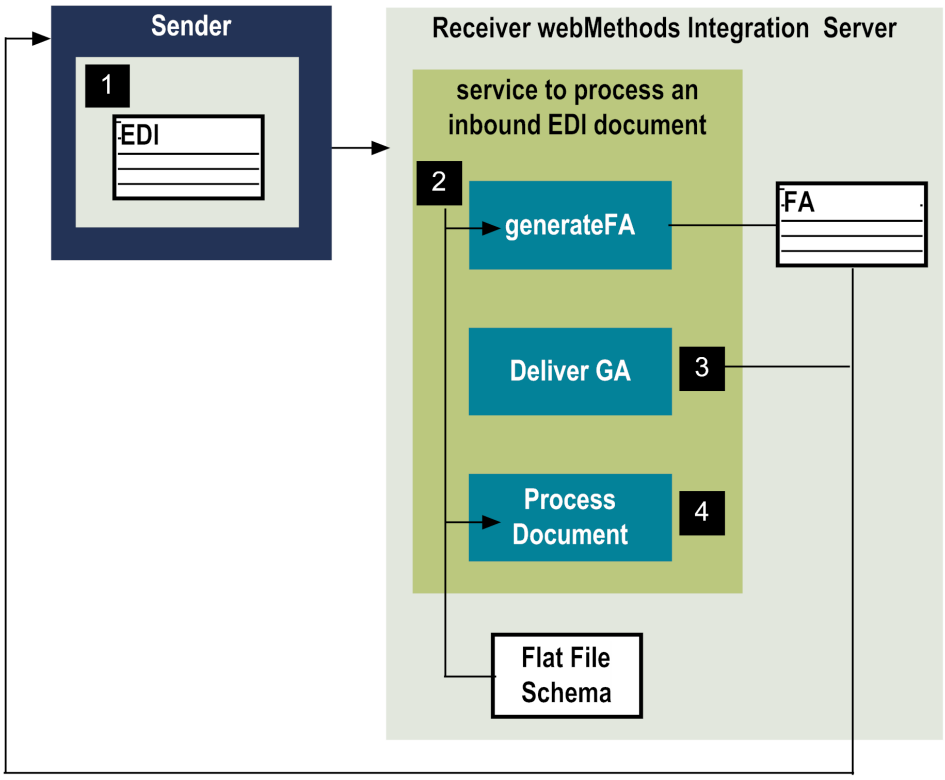
Generating Functional Acknowledgments

A *functional acknowledgment (FA)* is a type of EDI transaction set that acknowledges the receipt, as well as the structural and syntactical validity of an EDI document. When you receive a document, you can choose to generate an FA, which sends an EDI FA document to the sender to acknowledge receipt of the document. FAs validate and acknowledge only the syntax of the document, not that the document has been processed or understood by the receiver.

Module for EDI provides a built-in service that you can invoke to generate an FA. The service does not specify what to do with the FA that it creates. You must add additional logic to your service to deliver the FA to the sender of the original document.

FAs can also be generated automatically using the *FAGeneration/autoGenerateFA* EDITPA variable. For more information, see *webMethods Module for EDI Installation and User's Guide*.

The following diagram illustrates the basics of the FA generation process. For more information, see the table following this diagram.



Step	Description
1	The sender creates a client to send an EDI document to the receiver.
2	The EDI document is passed to the service that you create to process the EDI document in the *edidata* pipeline variable. You add logic to your service to invoke the built-in service that Module for EDI provides to generate the FA. To generate the FA, the built-in service uses the flat file schema associated with the inbound document's EDI standard, version, and transaction set to validate the inbound EDI document. Additionally, the built-in service uses a flat file schema associated with the FA's EDI standard and version to properly create the FA.
3	Your service delivers the FA to the sender by performing logic that you define.
4	Your service continues its processing of the EDI document. For more information, see "Processing Inbound EDI Documents with Trading Networks" on page 17.

For information about how to add logic to services to generate FAs, see *webMethods Module for EDI Installation and User's Guide*.

Generating Interchange Acknowledgments

In an X12 interchange, an *Interchange Acknowledgment* (TA1) notifies whether the control header and trailer envelope that surrounds one or more functional groups is received successfully. Also, the TA1 segment reports the syntactical accuracy of the envelope. The TA1 does not report the status of the functional groups and transaction sets within the interchange envelope.

Module for EDI provides a built-in service, wm.b2b.edi:generateX12TA1, to generate a TA1 acknowledgment manually. While generating the TA1 acknowledgment, this service provides the option to include a functional acknowledgment, as well. To generate a TA1 with or without an FA, you invoke the generateX12TA1 service from the service that you create to process an EDI document.

For more information about the wm.b2b.edi:generateX12TA1 service, see *webMethods Module for EDI Built-In Services Reference*.

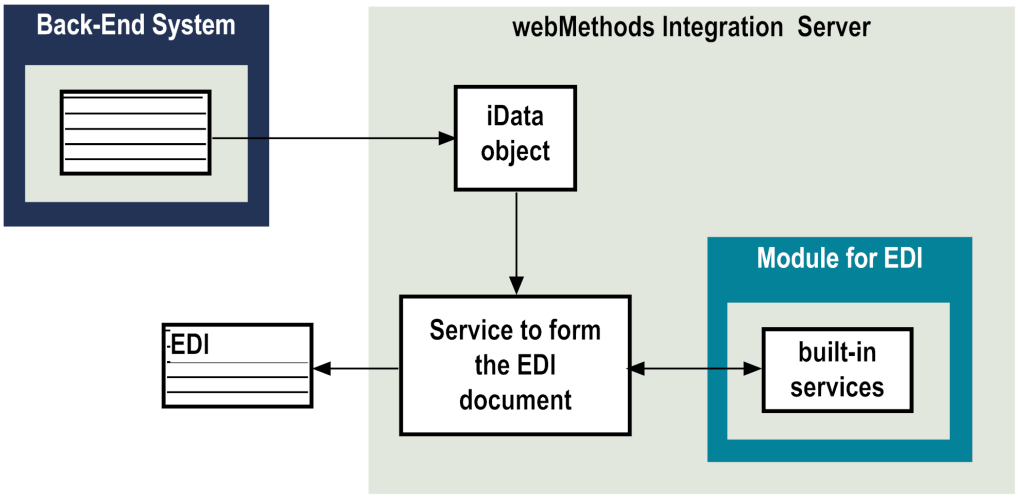
TA1s can also be generated automatically using the *FAGeneration/autoGenerateX12TA1* EDITPA variable. For more information, see *webMethods Module for EDI Installation and User's Guide*.

3 Forming EDI Documents and Sending Them Outbound

- Overview 36
- Creating a Service for Forming an Outbound EDI Document 37
- Delivering the EDI Document Directly from the Service that Forms It 39
- Submitting the Outbound EDI Document to Trading Networks to Deliver It 41
- Routing the Outbound EDI Document to Trading Networks to Deliver It 43

Overview

For outbound processing, you form an EDI document that can be sent outbound. For example, you might use data from an internal document (for example, a document from a back-end system) to form the EDI document.



To form the EDI document, you create a service. The Module for EDI provides built-in services that you can use as building blocks for creating the service.

In addition, you can access information for headers that the module maintains in the Trading Networks database, for example, in the EDITPAs. To deliver an outbound EDI document, rather than write your own logic to deliver the EDI document, if you want to send the outbound document to a VAN or batch it for delivery, you can use the features of the Module for EDI and Trading Networks to do so. For more information, see "Working with VANs" on page 47 and "Batching Outbound EDI Documents" on page 53.

To form the outbound EDI document and deliver it, you would:

- Define a TN document type for the internal-format document.
- Define a processing rule that processes the internal-format document. This processing rule should use the Execute a Service processing action to invoke the service that you create to form the outbound EDI document.

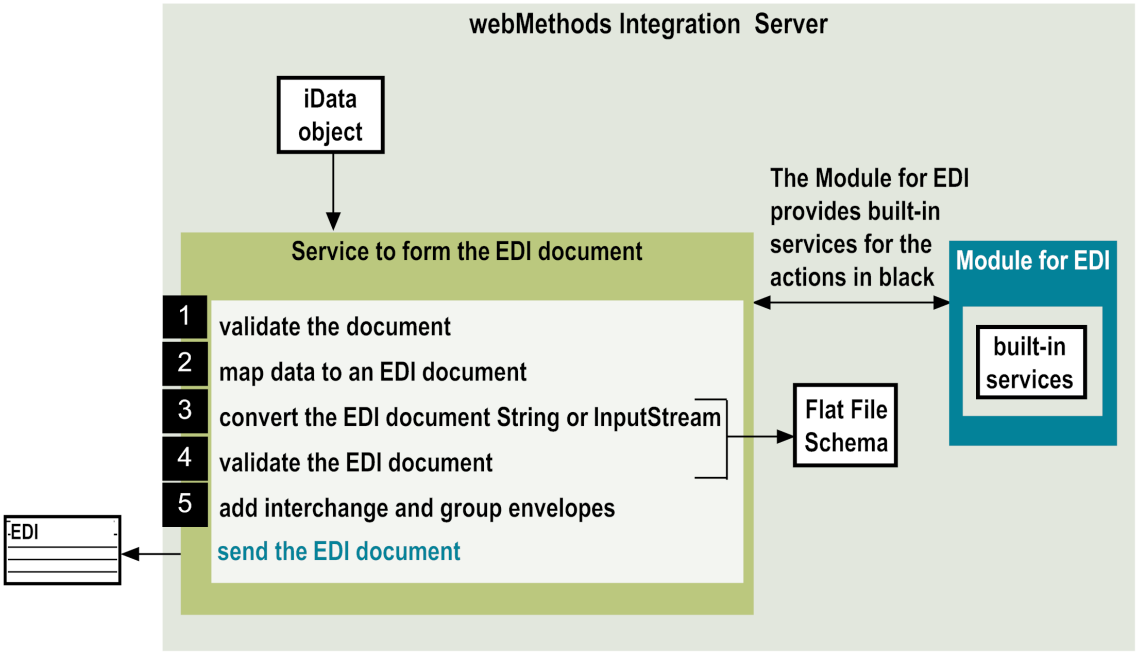
To deliver the outbound EDI document, you can:

Method	Notes
Add logic to deliver the document to your service that forms the outbound EDI document	■ The outbound EDI document is not saved in the Trading Networks database. ■ You cannot use this method if you want to batch the outbound EDI document or deliver the outbound EDI document to a VAN.

Method	Notes
Submit the outbound EDI document to Trading Networks document recognition	■ This method requires that you have a TN document type for the outbound EDI document. ■ This method requires that you have a second processing rule to deliver the outbound EDI document. ■ This method allows you to save the outbound EDI document to the Trading Networks database before delivering it. ■ You can use this method to deliver the outbound EDI document to a VAN or to batch the outbound EDI document.
Route the outbound EDI document to Trading Networks processing rules	■ This method bypasses document recognition, therefore you do not need a second TN document type for the outbound EDI document. ■ This method requires that you have a second processing rule to deliver the outbound EDI document. ■ This method allows you to save the outbound EDI document to the Trading Networks database before delivering it. ■ You can use this method to deliver the outbound EDI document to a VAN or to batch the outbound EDI document.

Creating a Service for Forming an Outbound EDI Document

The following diagram illustrates the basic processing you might want to include in a service that forms an EDI document. In the diagram, the service lists actions in both black and blue. The actions in black are those for which Module for EDI provides built-in services. For more information, see the table after the diagram.



The diagram shows the internal document being passed to the service as an IData object. This might be the case, for example, if the document is being passed to your service by an adapter service. The internal document might also be passed to the service as a String or InputStream. If this is the case, you can use built-in services that are provided with Module for EDI to convert the String or InputStream to an IData object.

The tables below provides more details about the type of processing the service can do:

Action	Description	Built-in service provided for?
1	Validate the internal document. If you want, you can validate the incoming document when it is an IData object to ensure its structure is valid before you form the EDI document.	Yes, provided with webMethods Integration Server
2	Map data from the internal document to the EDI document.	Yes
3	Convert the EDI document from an IData object to a String or InputStream. To be able to convert the EDI document, the Module for EDI uses a flat file schema that defines the structure of the EDI document it is forming. For more information, see "Before You Process EDI Documents" on page 16.	Yes

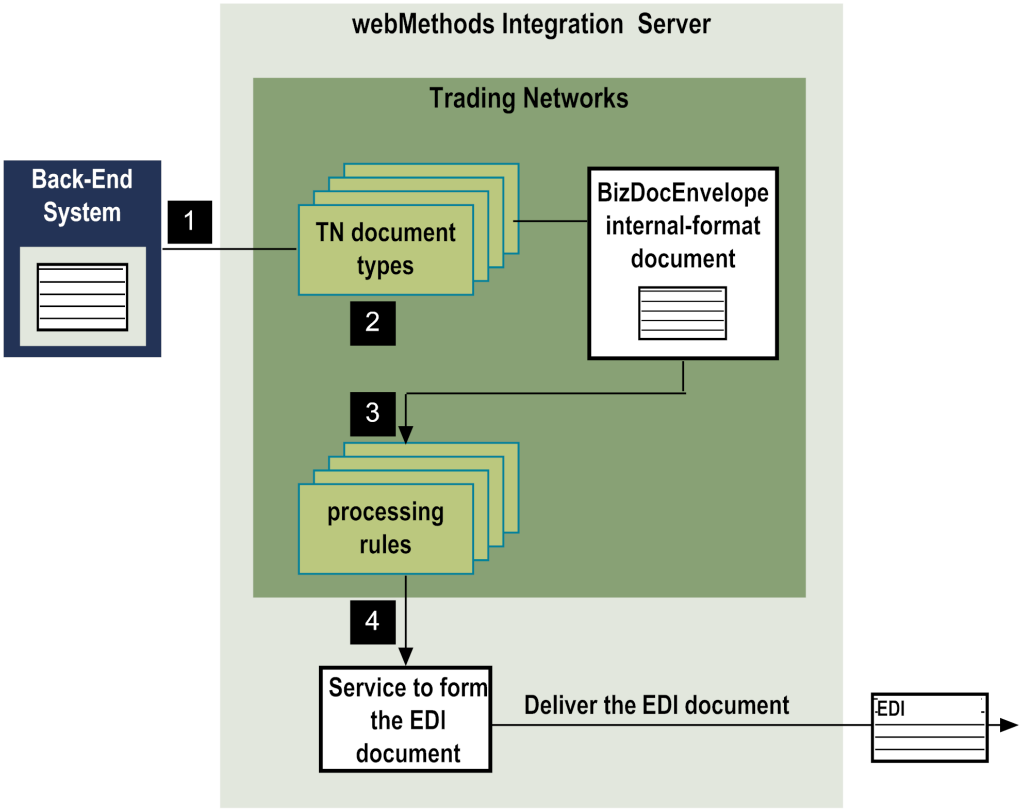
Action	Description	Built-in service provided for?
4	Validate the EDI document. Before you send the EDI document outbound, you can validate it to ensure its structure is correct. To be able to validate the EDI document, the Module for EDI uses a flat file schema. For more information, see "Before You Process EDI Documents" on page 16.	Yes
5	Add interchange and group envelopes to the EDI document to form the final EDI document.	Yes

For more information about how to create a service to form an outbound EDI document when you are using Trading Networks, and how to set up Trading Networks to deliver the outbound EDI document, see *webMethods Module for EDI Installation and User's Guide*.

Delivering the EDI Document Directly from the Service that Forms It

The following diagram illustrates the process of receiving an internal-format document and using a processing rule to invoke a service that you create to:

- Form the outbound EDI document based on the internal-format document
- Deliver the outbound EDI document



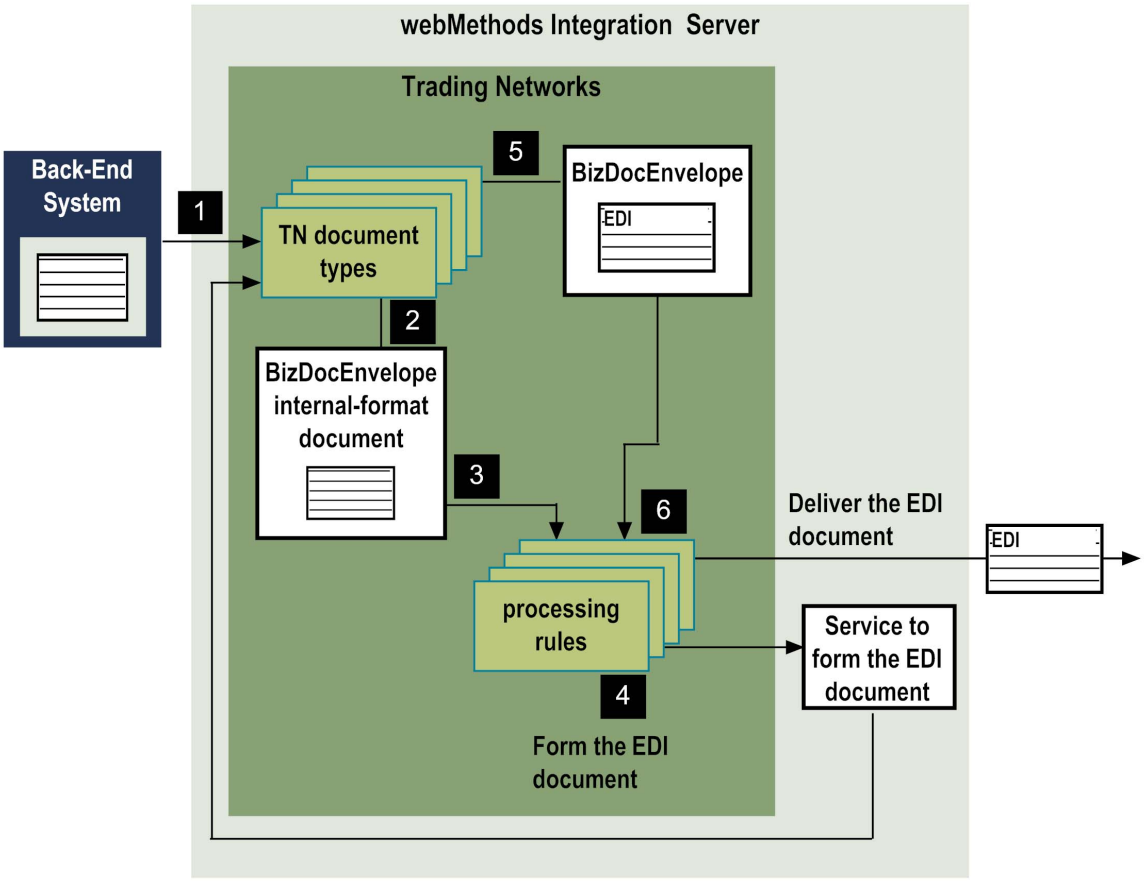
Step	Description
1	A back-end system or client sends an internal-format document to the webMethods Integration Server, invoking the wm.tn:receive service to send the document to Trading Networks.
2	Trading Networks matches the internal-format document to its TN document types. You need to create a TN document type that will match your internal-format. For more information about creating TN document types, see the Trading Networks administration guide for your release. See "About this Guide" for specific document titles. After selecting the appropriate TN document type, Trading Networks forms the BizDocEnvelope that contains the internal-format document as the content and places the BizDocEnvelope in the pipeline in the *bizdoc* variable.
3	Trading Networks searches its processing rules to find the appropriate rule to use to process the internal-format document. You should create a processing rule that uses the Execute a Service processing action to invoke a service that you create to form the EDI document. For more information about creating this service, see "Creating a Service for Forming an Outbound EDI Document" on page 37.
4	The service you create to form the EDI document executes. After forming the EDI document, your service invokes logic that you create to deliver the outbound EDI document.

For more information about how to create the service to form an outbound EDI document and send the outbound EDI document directly from the service, see *webMethods Module for EDI Installation and User's Guide*.

Submitting the Outbound EDI Document to Trading Networks to Deliver It

The following diagram illustrates the process of receiving an internal-format document and using a processing rule to invoke a service that you create to:

- Form the outbound EDI document based on the internal-format document.
- Submit the outbound EDI document back to Trading Networks document recognition to process the outbound EDI document through Trading Networks to deliver it.



Step	Description
1	A back-end system or client sends an internal-format document to the webMethods Integration Server invoking the wm.tn:receive service to send the document to Trading Networks.

Step	Description
2	Trading Networks matches the internal-format document to its TN document types. You need to create a TN document type that will match your internal-format. For more information about creating TN document types, see the Trading Networks administration guide for your release. See "About this Guide" for specific document titles. After selecting the appropriate TN document type, Trading Networks forms the BizDocEnvelope that contains the internal-format document as the content and places the BizDocEnvelope in the pipeline in the *bizdoc* variable.
3	Trading Networks searches its processing rules to find the appropriate rule to use to process the internal-format document. You should create a processing rule that uses the Execute a Service processing action to invoke a service that you create to form the EDI document. For more information about creating this service, see "Creating a Service for Forming an Outbound EDI Document" on page 37.
4	The service you create to form the EDI document executes. After forming the EDI document, your service invokes the wm.tn.doc.xml:routeXml service to submit the outbound EDI document back into Trading Networks document recognition.
5	Trading Networks document recognition passes the document to the EDI recognizer. The EDI recognizer executes as described in "Initial Processing of the EDI Recognizer" on page 21. That means it obtains EDITPA information and splits the document based on the split option in the EDITPA. For outbound documents, set the split option to: ■ `Group` if you want the Module for EDI to perform FA reconciliation in addition to delivering the outbound EDI document. ■ `Interchange` if you just want to deliver the outbound EDI document. Module for EDI selects the TN document type for the outbound EDI document, forms the BizDocEnvelope that contains the outbound EDI document as the content, and places the BizDocEnvelope in the pipeline in the *bizdoc* variable.

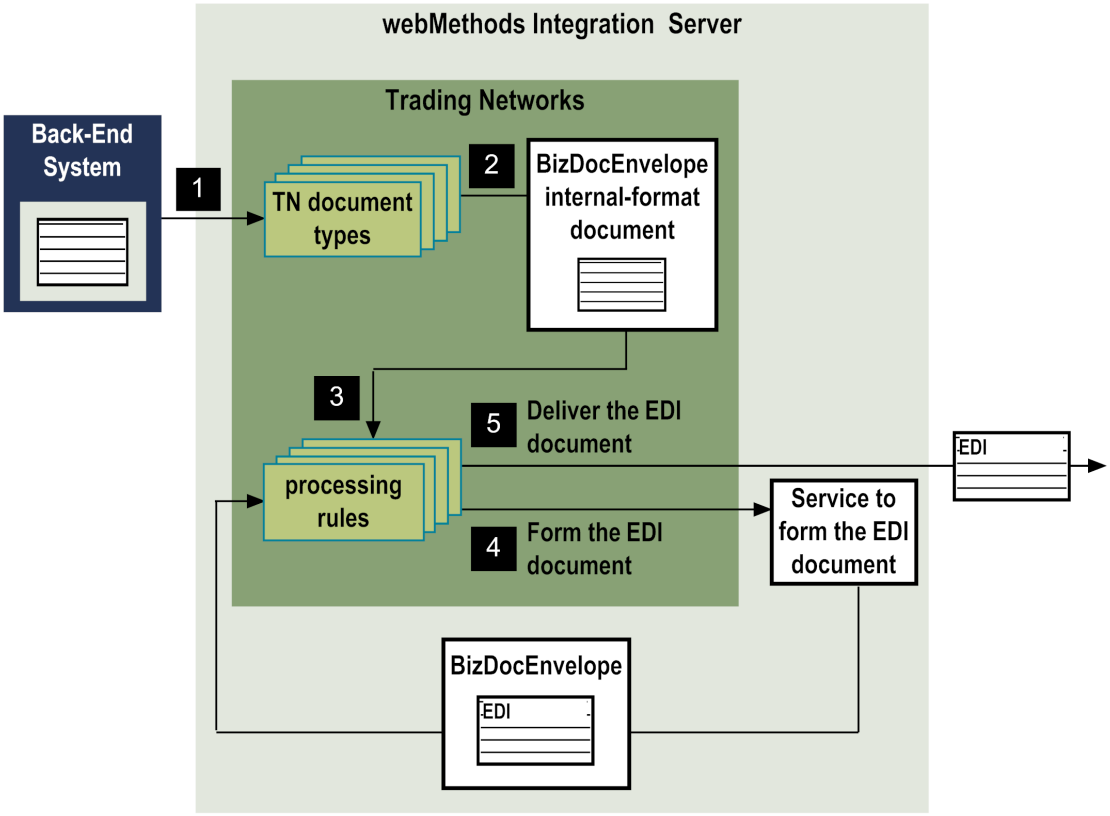
Step	Description

6	Trading Networks searches its processing rules again to find the appropriate rule to use to deliver the outbound EDI document. You need to create a processing rule for the EDI document that does one of the following:
	■ Uses the **Execute a Service** processing action that invokes a service that you created to deliver the EDI document.
	-OR-
	■ Uses the **Deliver Document By** processing action to:
	  ■ Send the EDI document to a VAN. For more information, see "Outbound Processing: Delivering Documents to VANs" on page 49.
	  ■ Batch the EDI document for delivery. For more information, see "Batching Outbound EDI Documents" on page 53.

Routing the Outbound EDI Document to Trading Networks to Deliver It

The following diagram illustrates the process of receiving an internal-format document and using a processing rule to invoke a service that you create to:

- Form the outbound EDI document based on the internal-format document.
- Route the outbound EDI document back to Trading Networks processing rules to select another processing rule to deliver the outbound EDI document.



Step	Description
1	A back-end system or client sends an internal-format document to the webMethods Integration Server, invoking the wm.tn:receive service to send the document to Trading Networks.
2	Trading Networks matches the internal-format document to its TN document types. You need to create a TN document type that will match your internal-format. For more information about creating TN document types, see the Trading Networks administration guide. See "About this Guide" for specific document titles. After selecting the appropriate TN document type, Trading Networks forms the BizDocEnvelope that contains the internal-format document as the content and places the BizDocEnvelope in the pipeline in the *bizdoc* variable.
3	Trading Networks searches its processing rules to find the appropriate rule to use to process the internal-format document. You should create a processing rule that uses the Execute a Service processing action to invoke a service that you create to form the EDI document. For more information about creating this service, see "Creating a Service for Forming an Outbound EDI Document" on page 37.

Step	Description
4	The service you create to form the EDI document executes. After forming the EDI document, your service creates a BizDocEnvelope that contains the EDI document and places it in the pipeline in the *bizdoc* variable (overwriting the previous *bizdoc* variable). Your service then invokes the wm.tn.route:routeBizDoc service to send the BizDocEnvelope that contains the EDI document back into Trading Networks to select a different processing rule.
5	Trading Networks searches its processing rules again to find the appropriate rule to use to deliver the EDI document. You need to create a processing rule for the EDI document that does one of the following: ■ Uses the Execute a Service processing action that invokes a service that you created to deliver the EDI document. -OR- ■ Uses the Deliver Document By processing action to: ■ Send the EDI document to a VAN. For more information, see "Outbound Processing: Delivering Documents to VANs" on page 49. ■ Batch the EDI document for delivery. For more information, see "Batching Outbound EDI Documents" on page 53.

4 Working with VANs

- Overview 48
- Inbound Processing: Retrieving Documents from VANs 48
- Outbound Processing: Delivering Documents to VANs 49

Overview

Many EDI-based business partners use Value Added Networks (VANs) as their primary EDI document exchange engine. You can use Module for EDI with Trading Networks to connect with these VANs. The module provides built-in services that enable VAN connectivity, allowing you to access VANs to retrieve EDI documents from the VAN, deliver EDI documents to the VAN, and obtain reports about EDI documents.

If you need to connect to a VAN other than GXS, ICC.NET, or MCI VANs, you might need to customize the provided service to suit the specific VAN connectivity.

Module for EDI supports PGP (Pretty Good Privacy). The module optionally can PGP-encrypt and sign documents bound for VANs that support version 3 PGP-encryption, as well as decrypt and verify PGP-encrypted documents from VANs. The ICC.NET supports PGP encryption.

Note: PGP-encryption support is deprecated and not generically supported across the webMethods components.

Inbound Processing: Retrieving Documents from VANs

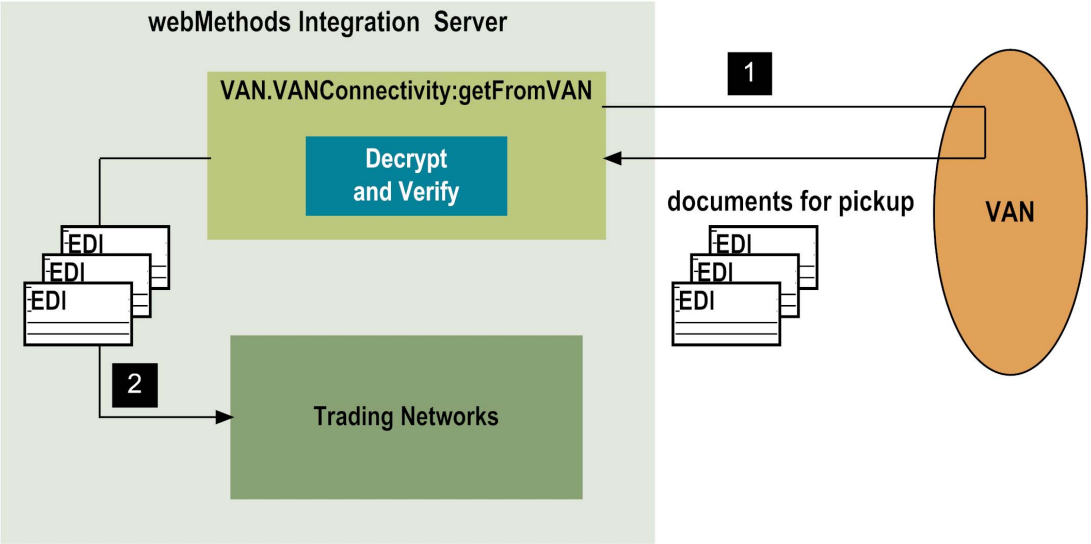
The Module for EDI provides the VAN.VANConnectivity:getFromVAN service to retrieve documents waiting on a VAN. You can use the Integration Server Administrator to create a scheduled task to have the webMethods Integration Server invoke this service at times you schedule.

You can set the inputs to the service to indicate other optional actions you want the VAN.VANConnectivity:getFromVAN service to take in addition to picking up the waiting documents. The optional actions you can request are:

- Decrypt and verify PGP-encrypted documents from VANs.
- Submit the picked up EDI documents to Trading Networks to have Module for EDI process the EDI document.
- Retrieve VAN-generated reports while connected to the VAN. The reports that are available depend on the VAN and can change at any time. Contact VANs directly for timely and accurate information.

  Note: You can also pick up documents from a VAN when you deliver documents to a VAN. This is described below in "Outbound Processing: Delivering Documents to VANs" on page 49.

Retrieving EDI documents from VANs



Step	Description
1	Invoke the VAN.VANConnectivity:getFromVAN service. The service connects to the VAN and returns the EDI documents waiting for pickup.
2	Optionally, you can have the VAN.VANConnectivity:getFromVAN service submit the EDI documents it picks up to Trading Networks. The Module for EDI processes the returned EDI documents like it would any inbound EDI documents. For more information, see "Processing Inbound EDI Documents" on page 15. If you do not submit the picked up documents to Trading Networks, you can create your own logic to process the EDI documents returned from the VAN.

For more information about setting up to retrieve inbound EDI documents from a VAN, see *webMethods Module for EDI Installation and User's Guide*.

Outbound Processing: Delivering Documents to VANs

Module for EDI provides the VAN.VANConnectivity:putToVAN service to deliver outbound EDI documents to a VAN. When you install the module, this service is registered as a Trading Networks registered delivery service and assigned the name VANFTP in Trading Networks.

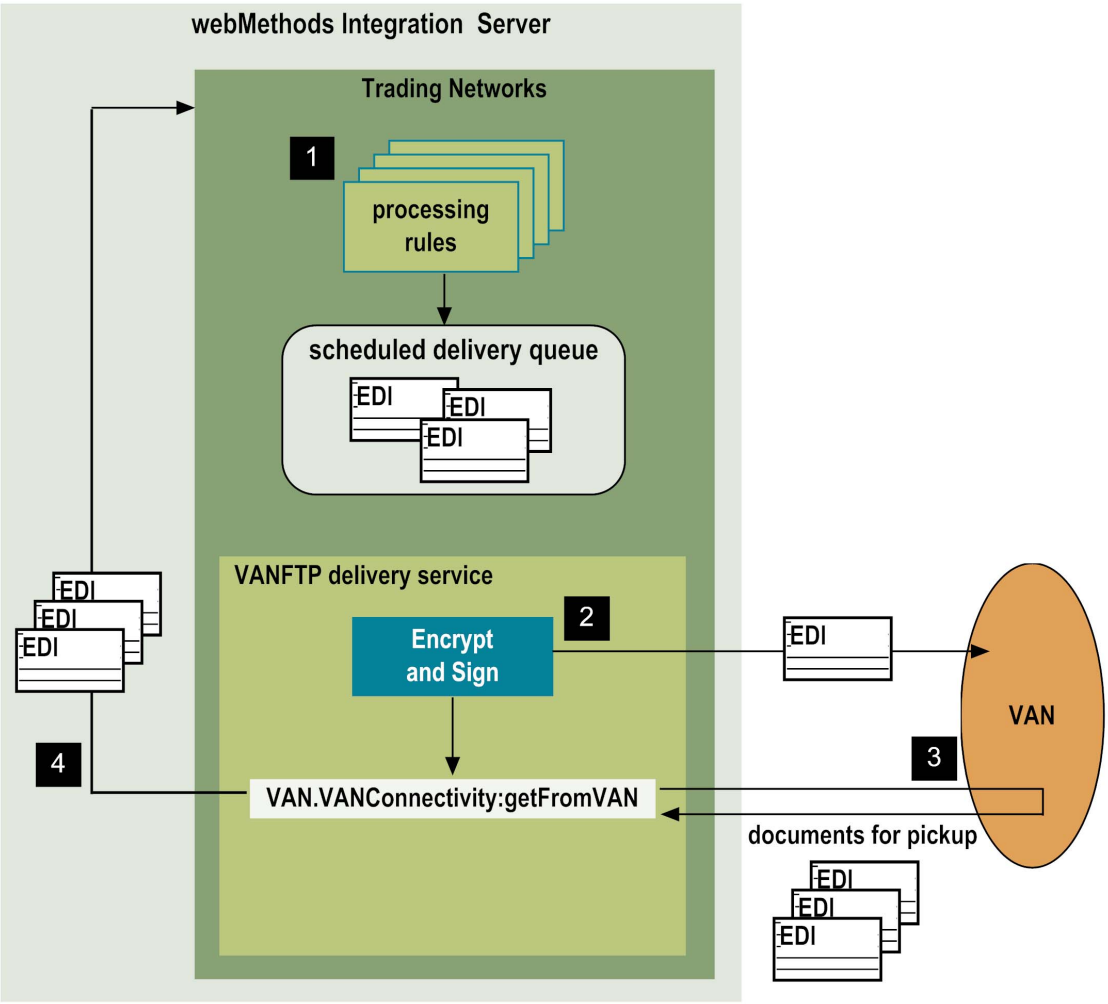
To use this service, you define a public scheduled delivery queue in Trading Networks. When you define the queue, you associate the queue with the VANFTP delivery service and specify a schedule for when Trading Networks is to deliver the documents in the queue. You can define as many queues as you need that use the VANFTP delivery service. For example, you might define two queues if you want to deliver documents to two different VANs.

To get EDI documents into the queue, you define a Trading Networks processing rule that uses the Deliver Document By processing action. When you set the Deliver Document By processing action, you use Scheduled Delivery and identify the name of the queue that uses the VANFTP delivery service. When the schedule that you associated with the queue indicates, Trading Networks invokes the VANFTP delivery service to deliver the documents that are in the queue to the VAN.

When you define the public scheduled delivery queue, you can set the inputs to the service to indicate other optional actions you want the VANFTP delivery service to take in addition to delivering the outbound EDI documents. The optional actions you can request are:

- PGP-encrypt and sign the documents before sending them to the VAN.
- Retrieve VAN-generated reports while connected to the VAN. The reports that are available depend on the VAN and can change at any time. Contact VANs directly for timely and accurate information.
- Retrieve any inbound documents that are waiting on the VAN. The EDI documents that are picked up are automatically submitted to Trading Networks to have Module for EDI process the EDI documents. (Note that in this situation, when you pick up EDI documents after delivery of EDI documents to a VAN, the module supports retrieving only non-PGP encrypted documents.)

Delivering EDI Documents to a VAN



Step	Description
1	Trading Networks uses its processing rules to determine how to deliver an EDI document. For more information about how to form an EDI document and send it to Trading Networks for delivery, see "Forming EDI Documents and Sending Them Outbound" on page 35. To deliver EDI documents to a VAN, you create a processing rule that uses the **Deliver Document By** processing action to deliver a document to a scheduled delivery queue associated with the VANFTP delivery service. For more information about scheduled delivery and defining queues for scheduled delivery, see the Trading Networks administration guide for your release. See "About this Guide" for specific document titles.

Step	Description
2	When the schedule that is associated with the queue indicates, Trading Networks invokes the VANFTP delivery service to deliver the EDI documents in the queue to the VAN. Based on input variables that you set for the service, the service can PGP encrypt and sign documents before sending them to the VAN.
3	Optionally, the VANFTP delivery service can invoke the VAN.VANConnectivity:getFromVAN built-in service, which Module for EDI provides, to retrieve any documents that might be waiting on the VAN for pick up. You specify whether you want documents picked up when you set the input variables of the VANFTP delivery service. Note that if you want to pick up documents that are PGP-encrypted, you should use the VAN.VANConnectivity:getFromVAN service as described in "Inbound Processing: Retrieving Documents from VANs" on page 48.
4	When the VANFTP delivery service invokes the VAN.VANConnectivity:getFromVAN service, it specifies that the EDI documents it picks up should be submitted to Trading Networks. Module for EDI processes the returned EDI documents like it would any inbound EDI documents. For more information, see "Processing Inbound EDI Documents" on page 15.

For more information about how to define the scheduled delivery queue for outbound EDI documents that are to be sent to a VAN, and how to define processing rules that place outbound EDI documents into a queue, see *webMethods Module for EDI Installation and User's Guide*:

5 Batching Outbound EDI Documents

- Overview 54
- Creating the Batched EDI Documents 55
- Delivering the Batched Document 57

Overview

EDI systems typically and historically work on batch documents. You can use the Module for EDI with Trading Networks to batch EDI documents for delivery rather than delivering EDI documents to the systems in real time as the documents are received.

Batching offers a more flexible and affordable approach to EDI document exchange and provides the following benefits:

- Enables documents to be grouped and sent at scheduled times that are more appropriate to organizational requirements.
- Increases system performance, requiring fewer communication connections and less time spent on authenticating envelopes that are sent individually.
- Makes working with legacy systems easier because legacy systems are batch-oriented.

To batch the documents, Module for EDI provides the wm.b2b.editn.batch:batchProcess service. When you install the module, this service is registered as a Trading Networks delivery service and assigned the name `EDI Batch` in Trading Networks. The batchProcess service combines EDI documents into a single document and adds group-level and interchange-level headers and trailers to the document.

To use the batchProcess service, you define public scheduled delivery queues in Trading Networks. When you define a queue, you associate the queue with the batchProcess service and specify a schedule for when Trading Networks is to invoke the batchProcess service to act on the documents in the queue.

To get the EDI documents into the queue so that they can be batched into a batch EDI document, you define a Trading Networks processing rule that uses the Deliver Document By processing action. When you set the Deliver Document By processing action, you use Scheduled Delivery and identify the name of the queue that uses the batchProcess delivery service. When the schedule that you associated with the queue indicates, Trading Networks invokes the batchProcess service to combine the EDI documents in the queue into a batch EDI document.

Note: VDA documents cannot be batched.

The batchProcess service uses its input variables and information in EDITPAs when creating the combined EDI document. You can set the input variables and EDITPA variables to:

- Control the creation of the UNA segment for UN/EDIFACT interchanges (or the BAT segment for TRADACOMS transmissions) for the batch process.
- Set default interchange or TRADACOMS transmission header values.

When batching EDI documents that were placed in the queue, Module for EDI can batch the documents into one of the following:

- A single output batch EDI document with multiple interchange or TRADACOMS transmission envelopes.
- Multiple output batch EDI documents, each containing only a single interchange or TRADACOMS transmission envelope.

For more information about how Module for EDI creates the output batch EDI document, see "Creating the Batched EDI Documents" on page 55.

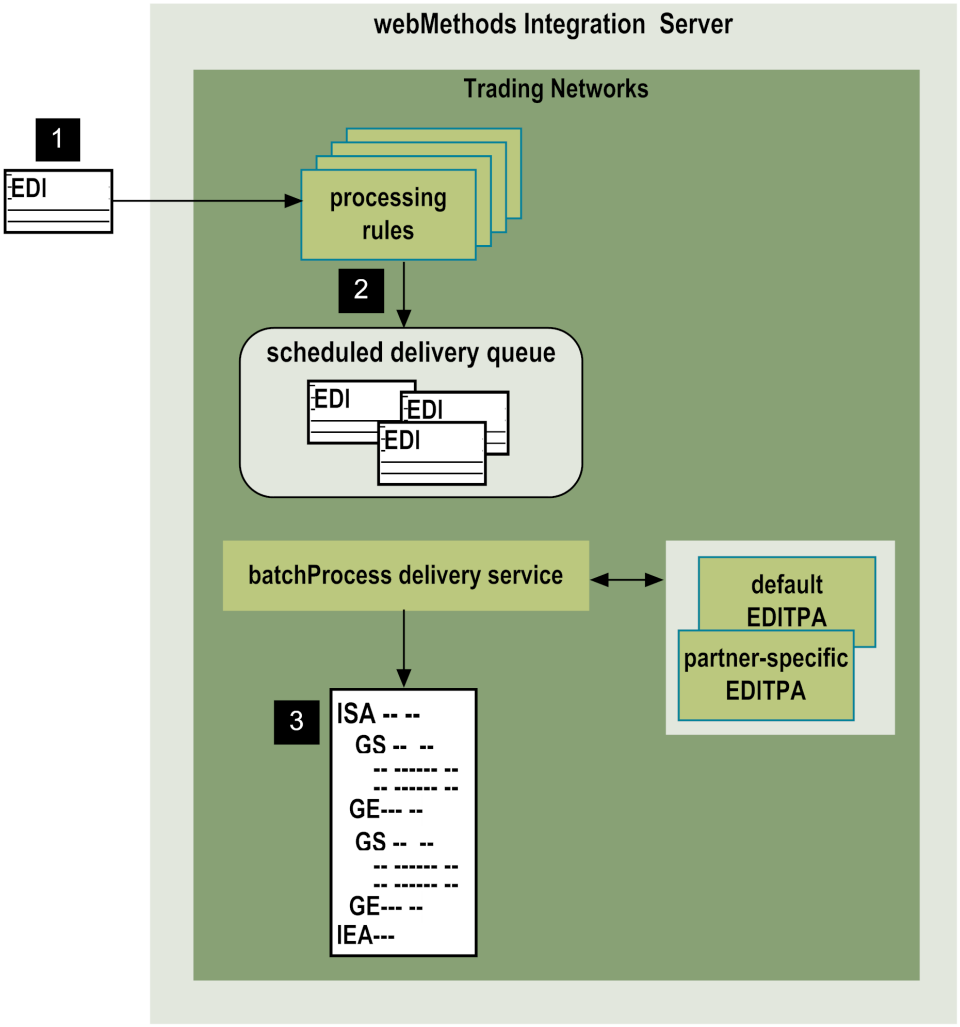
For more information about how to set up EDI batching, see *webMethods Module for EDI Installation and User's Guide*.

Creating the Batched EDI Documents

To set up to form batched EDI documents, you define scheduled delivery queues and processing rules in Trading Networks.

The following diagram illustrates how to form a batched EDI document. For more information, see the table after the diagram.

Forming the batched EDI document



Step	Description
1	EDI document is sent to Trading Networks. For more information about how to form an EDI document and send it to Trading Networks for delivery, see "Forming EDI Documents and Sending Them Outbound" on page 35.
2	Trading Networks uses its processing rules to determine how to process the EDI document. Trading Networks selects a processing rule that you create that uses the Deliver Document By processing action to deliver a document to a scheduled delivery queue associated with the batchProcess service. For more information about scheduled delivery, see the Trading Networks administration guide for your release. For more information about defining queues for scheduled delivery, see the Trading Networks user's guide for your release. See "About this Guide" for specific document titles.

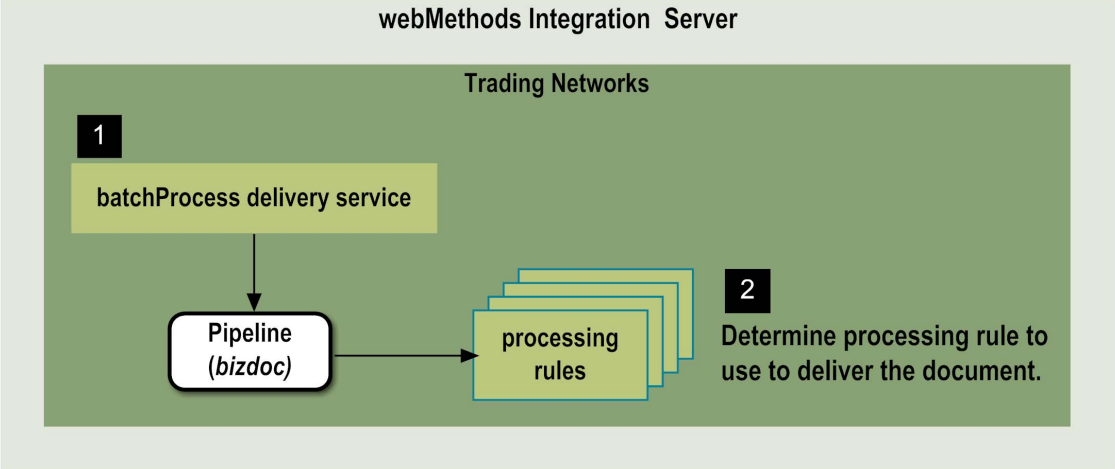
Step	Description
3	When the schedule that is associated with the queue indicates, Trading Networks invokes the batchProcess service to combine the EDI documents in the queue into the output batch EDI document(s). The final EDI document is ready for delivery. For more information about how Module for EDI processes the document so it can be delivered, see "Delivering the Batched Document" on page 57.

For more information about batching EDI documents, see *webMethods Module for EDI Installation and User's Guide*.

Delivering the Batched Document

After the batchProcess service forms the final batched EDI document, it routes the document back to Trading Networks processing rules for delivery. The following diagram illustrates the process for delivering a batched EDI document. For more information, see the table after the diagram.

Delivering a batch EDI document



Step	Description
1	The batchProcess service forms the final outbound EDI document as described in "Creating the Batched EDI Documents" on page 55. The batchProcess service then creates a BizDocEnvelope for the final outbound EDI document and places it in the pipeline in the *bizdoc* variable. It then routes the BizDocEnvelope to the Trading Networks processing rules.

Step	Description
2	After forming the BizDocEnvelope, Trading Networks determines the processing rule to use to deliver the outbound batch EDI document. You create the processing rule to define how you want to deliver the document. For example, you can invoke a service that you create to deliver the batch EDI document, or you can deliver the batch EDI document to a VAN as described in "Outbound Processing: Delivering Documents to VANs" on page 49.

For more information about defining processing rules to deliver outbound batch EDI documents, see *webMethods Module for EDI Installation and User's Guide*.

6 EDI Documents in Business Processes

- Overview 60
- How You Define the Business Process 60
- Conversation IDs for EDI Documents 61
- How EDI Documents are Passed to a Business Process 62

Overview

As an alternative to using processing rule actions, or in addition to using processing rule actions, you can define a business process (also called a conversation) that describes the steps required to process ANSI X12 or UN/EDIFACT, and VDA EDI documents. In the business process you can include steps:

- To send or wait for acknowledgments or response documents
- That require human interaction, which you can implement using Software AG Designer (Designer)

To use a business process for EDI documents, you must use Trading Networks.

Note: You cannot process TRADACOMS documents in a business process.

How You Define the Business Process

You define the actions that take place in a business process by using Designer to design a process model. A *process model* is a diagram that shows the steps in the business process. You set properties for each step to further define information about the actions to take for each step. For example, you can set properties that identify a service to invoke for a step.

Designer is a design-time tool only. Before the process model can be executed, you must create run-time elements for a process model. This is called building and uploading the process model. When you build and upload the process model, Designer generates triggers, flow services, etc. based on the steps in your process model and saves these run-time elements in the Integration Server namespace.

At run time the Process Engine, which is a facility of the Integration Server, manages the execution of business processes. The Process Engine executes the business process by using the appropriate run-time elements that were generated from a process model.

Typically, a business process starts based on the arrival of a document. It is the responsibility of the Process Engine to determine the actions to take for a specific document. The Process Engine determines the process model to use for the document and defines a new instance of the process model to govern the actions to take for the business process. When a subsequent document for the business process arrives, it is the Process Engine that determines the correct running instance of a process model and rejoins the business process by passing the document that just arrived.

The way the Process Engine determines the documents that belong to a single instance of a business process is through the conversation ID. All documents in the same instance of a business process share an identifier called a conversation ID. When the Process Engine receives a document, it determines whether it has a running business process that uses the conversation ID. If it does, the Process Engine passes the document to the running business process to rejoin the running business process. If there is no running business

process that uses that conversation ID, the Process Engine searches for a process model where the first step is "waits for document", and if found, starts a new instance of the process model.

As the Process Engine manages the execution of a business process, it logs its progress and status to the Process Audit Log database component. You can view the progress and status using webMethods Monitor.

For more information about:

- How to create process models, see the Designer online help for your release. See "About this Guide" for specific document titles.
- How to monitor running business processes, see the webMethods Monitor documentation.

Conversation IDs for EDI Documents

Before an EDI document can be used in a business process, it must have a conversation ID.

The original EDI document is split into Transaction, Group, and Interchange documents as described in "Processing Inbound EDI Documents" on page 15. To process these documents using a business process, the document (Transaction, Group, and Interchange) must have a conversation ID.

Note: You are not required to process all three types of documents using a business process. Use the documents that work best to create your solution.

- For a Transaction document, you must provide Module for EDI with information that it uses to form the conversation ID. The information that you provide is an instance ID query. The instance ID query is a query that the module can perform against the Transaction document to retrieve a value for the conversation ID. The instance ID query is specific to the type of transaction set contained in the Transaction document. For example, you might define the instance ID query `ST/BEG/BEG03` for an X12 4010 850 transaction set. Whenever the EDI recognizer creates a Transaction document that contains an X12 4010 850 transaction set, it will use the instance ID query you specify to obtain the value to use for the conversation ID for the Transaction document. If the EDI recognizer creates a Transaction for which there is no instance ID query, the conversation ID is not set.

  Note: If you do not need to process the Transaction document, do not create an instance ID query for the transaction set, and do not create a process model that uses the document.

- For a Group or Interchange document, Module for EDI always assigns conversation IDs. No instance ID query is required. For a Group document, the module sets the value of the conversation ID to the group control number. For an Interchange document, the module sets the value of the conversation ID to the interchange control number.

Note: The Group and Interchange documents will always have a conversation ID. If you do not need to process either the Group or Interchange document in a business process, do not create a process model that uses the document. If the Process Engine is unable to locate a matching process model, it does not perform processing for the document.

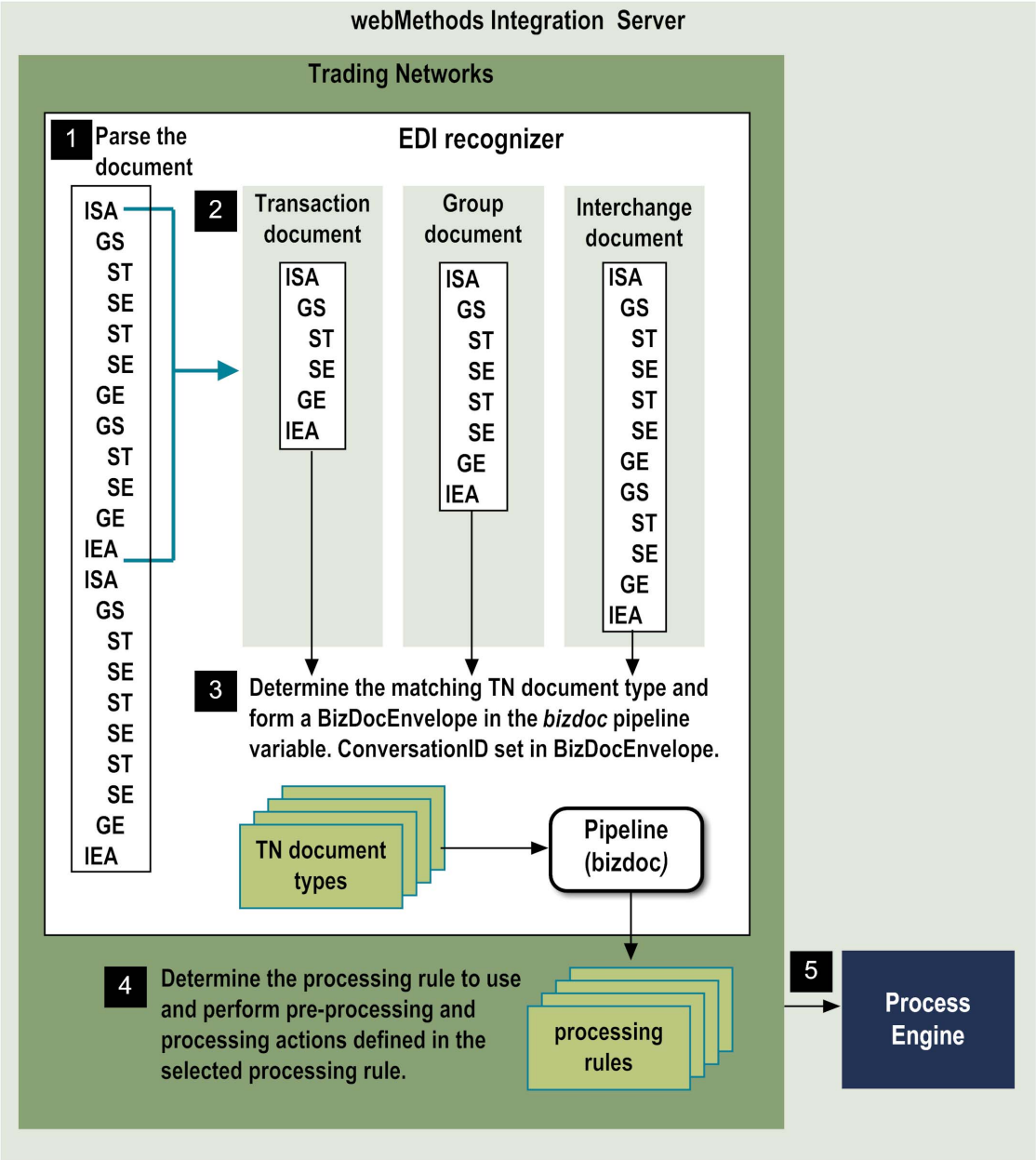
For more information about how to define instance ID queries to set conversation IDs for Transaction documents, see *webMethods Module for EDI Installation and User's Guide*.

How EDI Documents are Passed to a Business Process

For an EDI document to be used in a business process, Trading Networks must send the document to the Process Engine.

First Trading Networks does its own processing (document recognition and performing the actions defined by a processing rule). Then, if the Trading Networks system attribute ConversationID has a value, Trading Networks passes the document to the Process Engine.

The following diagram illustrates the steps taken to send an EDI document to the Process Engine for processing. For more information, see the table after the diagram.



Step	Description
1	When Trading Networks receives an EDI document, it passes the EDI document to the EDI recognizer. The EDI recognizer parses the inbound EDI document.

Step	Description
2	For each interchange segment, the EDI recognizer retrieves the appropriate EDITPA to determine the split option to use for the EDI document. The EDI recognizer splits the EDI document into the appropriate documents (i.e., Transaction, Group, and Interchange documents). For more information about how EDI documents are split, see "Processing Inbound EDI Documents" on page 15.
3	The EDI recognizer submits the newly formed documents (Transaction, Group, or Interchange) to Trading Networks for processing. The EDI recognizer uses the TN document types to determine the type of document. After recognizing the type of document using TN document types, the EDI recognizer forms a BizDocEnvelope for the EDI document and sets the ConversationID system attribute in the BizDocEnvelope. For more information about the value used for the Conversation ID, see "Conversation IDs for EDI Documents" on page 61.
4	After forming the BizDocEnvelope, Trading Networks determines the processing rule to use to process the document and executes the processing rule. When you have a document that you plan to process using a business process, you will typically either 1) set up the processing rule to ignore the document -or- 2) not define a processing rule for the document causing Trading Networks to select the Default processing rule, which ignores the document.
5	Because the ConversationID system attribute contains a value, Trading Networks passes the document to the Process Engine. The Process Engine either starts a new business process based on a process model that you have designed or determines the running business process that the document is to join.

A Using Module for EDI Decoupled from Trading Networks

- Overview 66
- Processing of Inbound EDI Documents 66
- Forming EDI Documents to Send Outbound 69

Overview

In a distributed environment, to handle heavy EDI document processing loads, you can configure some of your servers to do the job of just translating the EDI documents by decoupling Module for EDI from Trading Networks. When you decouple the module from Trading Networks, you only use the core EDI document processing built-in services in the WmEDI package. You can have a centralized Trading Networks server for storing the transactions in your EDI network and routing the EDI documents to your partners.

Using the WmEDI package with only the functionality available in the webMethods Integration Server provides the functionality to:

- Process most EDI standards, such as ANSI X12, VICS, UCS, UN/EDIFACT, ODETTE, EANCOM, and VDA

  Note: The WmEDI package does not support TRADACOMS document processing.

- Parse, convert, format, and validate EDI documents
- Process EDI documents containing multiple interchanges/groups/transactions with multiple versions
- Generate functional acknowledgments (FAs), if they are applicable to your standard
- Create envelopes for EDI documents
- Transport EDI documents using the FTP, HTTP, and HTTPS protocols

Configuring Module for EDI Decoupled from Trading Networks

There are three aspects to your EDI solution to decouple Module for EDI from Trading Networks:

- How to decouple the module fromTrading Networks. For more information, see *webMethods Module for EDI Installation and User's Guide*.
- How to send EDI documents to the webMethods Integration Server and how you process them; that is, inbound processing. For more information, see "Processing of Inbound EDI Documents" on page 66.
- How you form EDI documents from internal-format documents (for example, documents from back-end systems) and send the EDI documents outbound; that is, outbound processing. For more information, see "Forming EDI Documents to Send Outbound" on page 69.

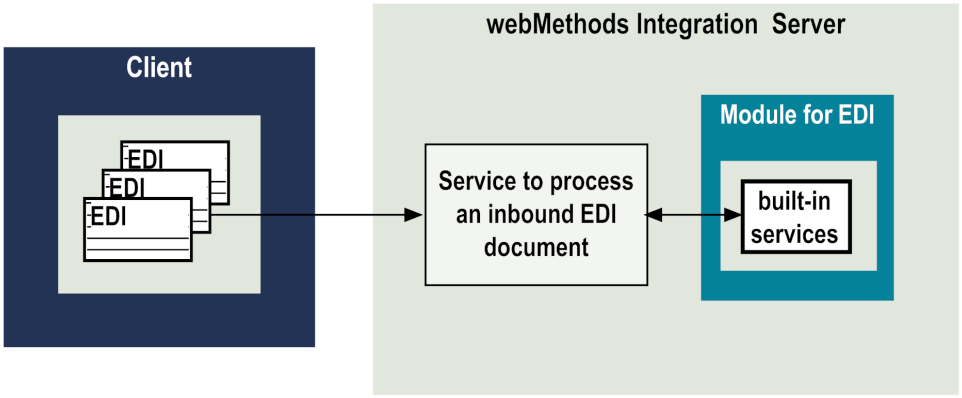
Processing of Inbound EDI Documents

For inbound processing, you create:

- Clients that send EDI documents to the webMethods Integration Server. For more information, see "Creating an EDI Client" on page 17.
- Services that process the inbound EDI document. For more information see, "Processing of Inbound EDI Documents" on page 66.
- Services for generating the functional acknowledgments to acknowledge the receipt, as well as the structural and syntactical validity of an EDI document. For more information, see "Generating Functional Acknowledgments" on page 32.

The EDI documents are documents in standard EDI format, such as ANSI X12, UCS, VICS, UN/EDIFACT, ODETTE, EANCOM, or VDA.

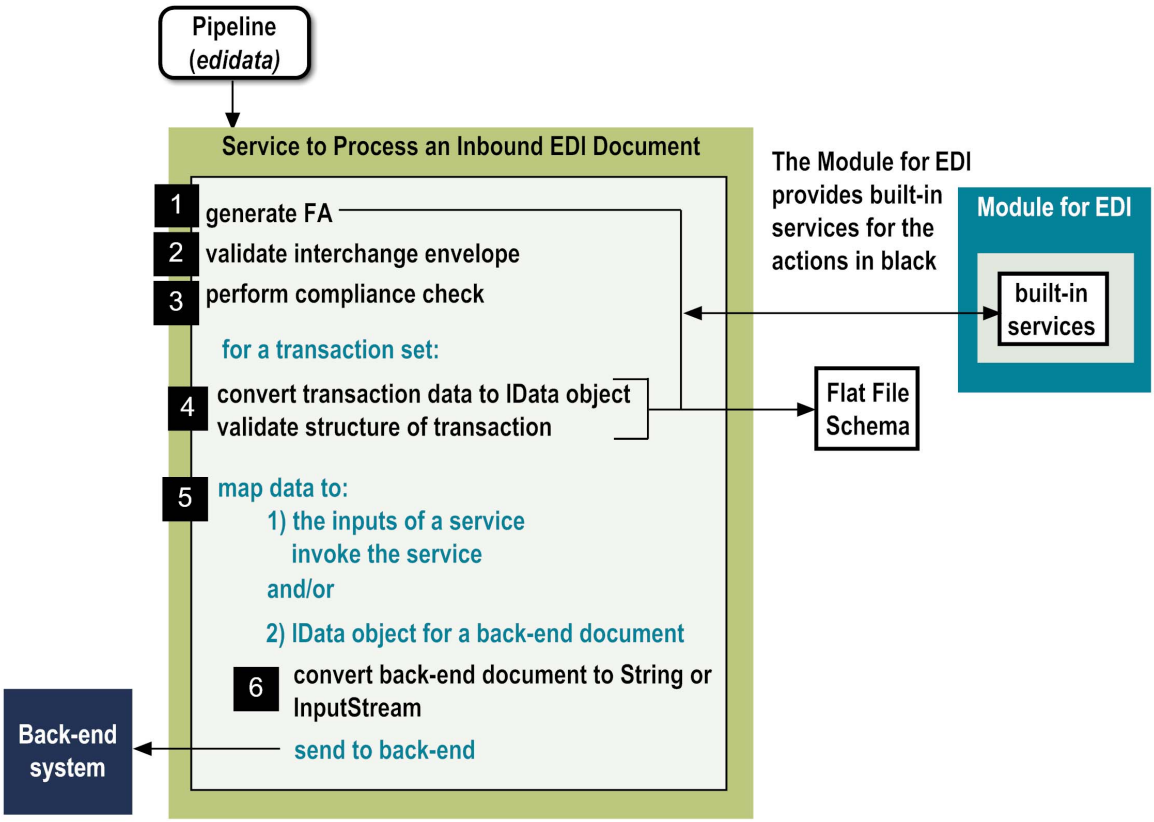
Note: The WmEDI package does not support TRADACOMS document processing. Support for the TRADACOMS standard is provided when you use the Module for EDI in conjunction with webMethods Trading Networks.



Services to Process the Inbound EDI Document

Module for EDI provides built-in services that you use as building blocks for creating services that process your inbound EDI documents. Typical ways to process an EDI document might be to map the data from the EDI document to another format (for example, the format that a back-end system requires) or to map data from the EDI document to the inputs of a service.

The following diagram illustrates the basic processing you might want to include in a service that processes an EDI document. In the diagram, the service lists actions in both black and blue. The actions in black are those for which the Module for EDI provides built-in services. For more information, see the tables after the diagram.



The service receives the EDI document in the *edidata* variable in the pipeline. The tables below provides more details about the type of processing the service can do:

Action	Description	Built-in service provided for?
1	Generate a functional acknowledgment (FA) for the EDI document. For more information, see "Generating Functional Acknowledgments" on page 32.	Yes
2	Perform an interchange envelope validation that includes validating field lengths, code lists, ranges, and partitions.	Yes
3	Perform a compliance check to check for matching interchange control numbers, matching group control numbers, matching transaction control numbers, segment counts, transaction counts, and group counts.	Yes

When processing an EDI document, the majority of the effort will most likely be in processing the individual transaction sets within the EDI document. You can perform the following when processing a transaction set within the EDI document:

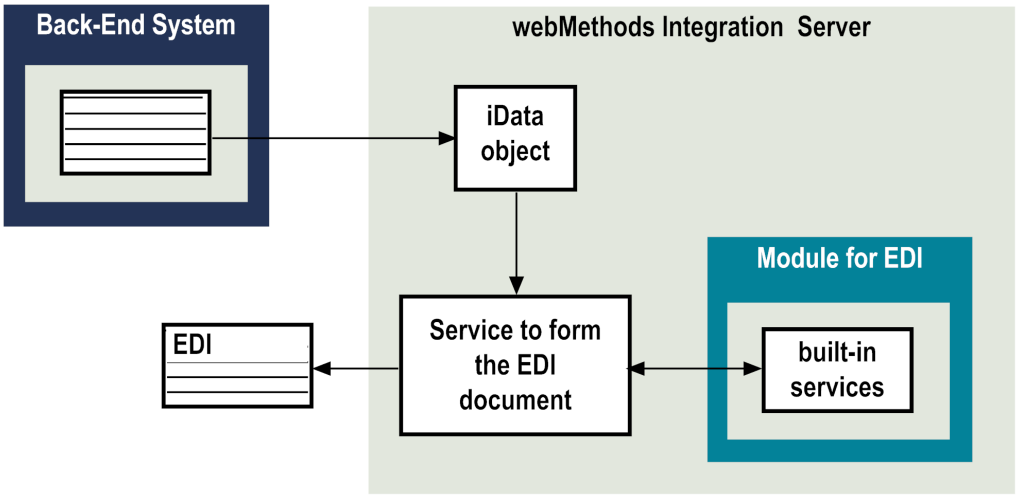
Action	Description	Built-in service provided for?
4	Convert the EDI transaction set from a String or InputStream into an IData object and validate its structure. To be able to convert a transaction set to an IData object, the Module for EDI uses a flat file schema that defines the structure of the transaction set. Additionally, it uses the flat file schema to validate that the structure of the EDI transaction is correct. For more information, see "Flat File Schemas" on page 16.	Yes
5	Map data from an EDI transaction to a target. After the transaction set is in an IData object, you can map the data from the IData object for the EDI transaction set. For example, you can map the data to the inputs of another service or to an internal-format document (for example, a format required by a back-end system).	Yes
6	If the service mapped EDI data to an internal-format document (for example, the back-end system document), convert the internal-format document from an IData object to a String or InputStream. To be able to convert this document, the Module for EDI uses a flat file schema that defines the structure of the your back-end system. For more information, see "Before You Process EDI Documents" on page 16. The internal-format document as a String or InputStream is now in a format that you can use to deliver it (for example, to the back-end system).	Yes

For information about how to create a service to process an inbound EDI document, see *webMethods Module for EDI Installation and User's Guide*.

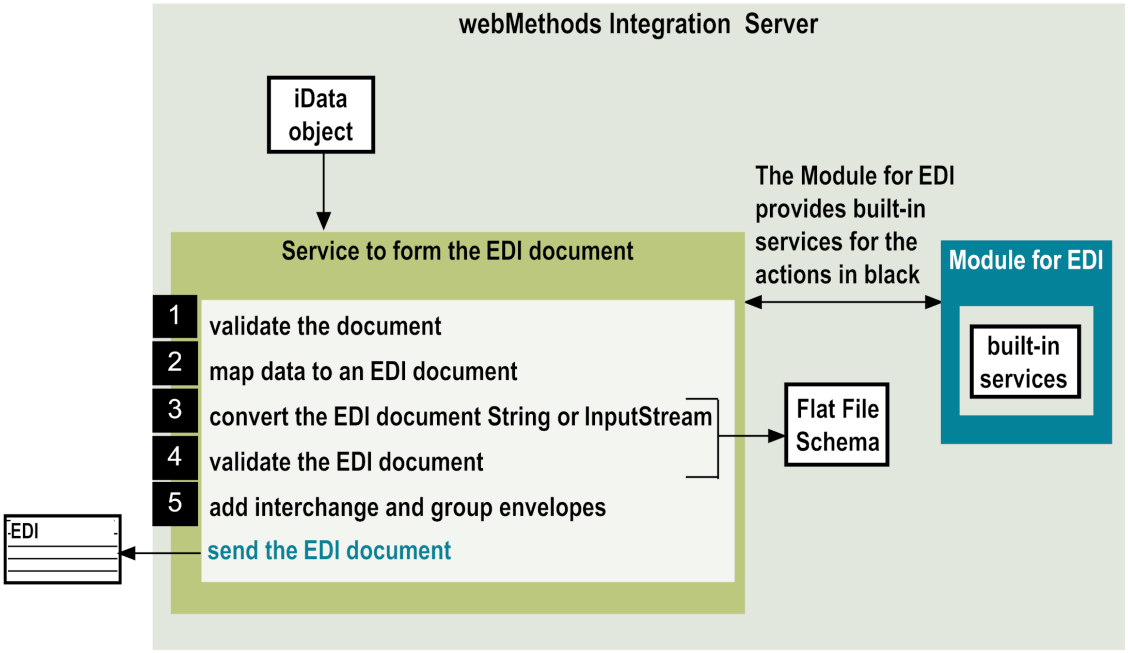
Forming EDI Documents to Send Outbound

For outbound processing, you form an EDI document that can be sent outbound. For example, you might use data from an internal document (for example, a document from a back-end system) to form the EDI document.

To form the EDI document, you create a service. Module for EDI provides built-in services that you can use as building blocks for creating the service.



The following diagram illustrates the basic processing you might want to include in a service that forms an EDI document. In the diagram, the service lists actions in both black and blue. The actions in black are those for which Module for EDI provides built-in services. For more information, see the table after the diagram.



The above diagram shows the internal document being passed to the service as an IData object. This might be the case, for example, if the document is being passed to your service by an adapter service. The internal document might also be passed to the service as a String or InputStream. If this is the case, you can use built-in services that are provided with the Module for EDI to convert the String or InputStream to an IData object.

The table below provides more details about the type of processing the service can do:

Action	Description	Built-in service provided for?
1	Validate the internal document. If you want, you can validate the incoming document when it is an IData object to ensure its structure is valid before you form the EDI document.	Yes, provided with webMethods Integration Server
2	Map data from the internal document to the EDI document.	Yes
3	Convert the EDI document from an IData object to a String or InputStream. To be able to convert the EDI document, Module for EDI uses a flat file schema that defines the structure of the EDI document it is forming. For more information, see "Before You Process EDI Documents" on page 16.	Yes
4	Validate the EDI document. Before you send the EDI document outbound, you can validate it to ensure its structure is correct. To be able to validate the EDI document, Module for EDI uses a flat file schema. For more information, see "Before You Process EDI Documents" on page 16.	Yes
5	Add interchange and group envelopes to the EDI document to form the final EDI document.	Yes

For information about how to create a service to form an outbound EDI document, see *webMethods Module for EDI Installation and User's Guide*.