



TIBCO BusinessConnect™ EDI Protocol powered by Instream®

User's Guide

*Software Release 6.10
October 2019*



Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

ANY SOFTWARE ITEM IDENTIFIED AS THIRD PARTY LIBRARY IS AVAILABLE UNDER SEPARATE SOFTWARE LICENSE TERMS AND IS NOT PART OF A TIBCO PRODUCT. AS SUCH, THESE SOFTWARE ITEMS ARE NOT COVERED BY THE TERMS OF YOUR AGREEMENT WITH TIBCO, INCLUDING ANY TERMS CONCERNING SUPPORT, MAINTENANCE, WARRANTIES, AND INDEMNITIES. DOWNLOAD AND USE OF THESE ITEMS IS SOLELY AT YOUR OWN DISCRETION AND SUBJECT TO THE LICENSE TERMS APPLICABLE TO THEM. BY PROCEEDING TO DOWNLOAD, INSTALL OR USE ANY OF THESE ITEMS, YOU ACKNOWLEDGE THE FOREGOING DISTINCTIONS BETWEEN THESE ITEMS AND TIBCO PRODUCTS.

This document is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, TIB, Information Bus TIBCO ActiveMatrix BusinessWorks, TIBCO ActiveMatrix BusinessWorks Plug-in for BusinessConnect, TIBCO Administrator, TIBCO BusinessConnect, TIBCO BusinessConnect Palette, TIBCO Business Studio, TIBCO Designer, TIBCO Foresight EDISIM, TIBCO Hawk, TIBCO Rendezvous, and TIBCO Runtime Agent are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle Corporation and/or its affiliates.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

This software may be available on multiple operating systems. However, not all operating system platforms for a specific software version are released at the same time. See the readme file for the availability of this software version on a specific operating system platform.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

This and other products of TIBCO Software Inc. may be covered by registered patents. Please refer to TIBCO's Virtual Patent Marking document (<https://www.tibco.com/patents>) for details.

Copyright © 1999-2019. TIBCO Software Inc. All Rights Reserved.

Contents

Figures	ix
Tables	xi
Preface	xiii
Related Documentation	xiv
TIBCO BusinessConnect EDI Protocol powered by Instream Documentation	xiv
Other TIBCO Product Documentation	xv
Typographical Conventions	xvi
TIBCO Product Documentation and Support Services	xviii
How to Access TIBCO Documentation	xviii
How to Contact TIBCO Support	xviii
How to Join TIBCO Community	xviii
Chapter 1 Introduction	1
EDI Standard	2
Product Overview	3
Protocol Variants	4
Transports	6
Set Up Trading Partner Transport via Inbox	7
Security	8
Validation and Conversion	9
EDI Data Formats	11
Integration	12
Messages	13
Chapter 2 Managing EDI Guidelines	15
Overview	16
EDI Guideline Files	17
Managing Guidelines with EDISIM	19
Starting EDISIM Standards Editor	19
Customizing a Guideline	20
Exporting Guideline and Schema Files	24

Storing Guidelines and Schemas in the Configuration Store	25
EDI Conversion Engine	27
EdiConvert Utility	28
Chapter 3 Exchanging Documents	31
Overview	32
Setting Up Trading Partners	34
EDI Information Not In Guidelines	34
Creating Private Processes	35
TIBCO BusinessConnect Activities	35
Exchanging URI Definitions	38
Exchanging Identity Information	39
Chapter 4 Configuring Protocol Features	41
Overview	42
Validation and XML Conversion	43
TEXT to EDI Conversion	44
Control Number Management	45
Control Number Validation	45
Alphanumeric Control Numbers and Validation	49
Validation Error Reporting	50
Acknowledgment and Reconciliation	51
Transaction Batching and Scheduling	52
CAQH Connectivity Rules	53
SOAP + WSDL Envelope	53
HTTP MIME Multipart Envelope	57
Process Flows and Transaction Types	62
Tuning for Improved Performance	64
Memory Tuning	64
Tuning Thread Count	65
Tuning Flow Control	65
Chapter 5 Acknowledgments and Reconciliation	67
Acknowledgments	68
Types of Acknowledgments	68
Acknowledgment Timeouts	69
How Acknowledgment Timeouts Work	69
Reconciling Acknowledgments	70
How Acknowledgments Are Reconciled	70

Reconciling Acknowledgments After Timeout	72
Reconcile After Timeout Check box	72
Completing Transactions After Timeout	72
Chapter 6 File Masks	73
FTP Inbound File Name Masks	74
File Name Mask Examples	74
File Name Mask Syntax	74
FTP and FILE Outbound File Name Masks	76
File Name Mask Examples	76
File Name Mask Syntax	76
Ack File Name Masks	78
File Name Mask Examples	78
File Name Mask Syntax	78
Chapter 7 Viewing Logs	79
Overview	80
Using the Log Viewer	80
Status Codes	81
Audit Log	84
Summary View	84
Transaction Details View	86
State Details View	89
User Keys	92
Non-Repudiation Log	94
Summary View	94
Message Queue Log	95
Summary View	95
Log Viewer Options	96
Transactions in the Message Queue Viewed in the Audit Log	96
Resend Log	97
Manual Resend Scenarios	98
Document Archiving	100
Chapter 8 Private Messages	101
Overview	102
Initiator Messages	103
Initiator Outbound Request	103
Initiator Inbound Response	110
Initiator Acknowledgment From Partner	113
Responder Messages	115

Responder Inbound Request	115
Responder Inbound Interchange	119
ResponderInterchangeComplete	120
Responder Outbound Synchronous Response	122
Responder Acknowledgment to Partner	124
General Messages	127
CAQH Message	127
Timeout Alert	128
Validation Alert	129
Error Message	130
Insight Report Message	131
Push Real-Time Partner and Guideline Updates Message	132
Objects Included in Private Messages	136
TPName	136
EnvelopeInfo	136
InterchangeInfo	137
GroupInfo	137
Identity	137
ControlNumberInfo	138
TransactionInfo	138
ConversionEngineInfo	139
Attachment	140
Appendix A Property Reference	141
Property Reference	142
Appendix B Status and Error Codes	147
HTTP Status Codes: 200-510	148
Status Codes: 600-699	149
Appendix C EDI Migration and Operations Updater Tools	151
EDI Migration Tool	152
Using the EDI Migration Tool	152
Migrating from the Version 5.x to 6.x and Later	153
Operations Updater Tool	155
Using the Operations Updater Tool	155
Appendix D Database Schema Definition	156
EDI Batch Schema Details	157
TIBEDI_BATCH_TASK	158
TIBEDI_BATCH_TX	159

TIBEDI_BATCH_TX_BIN 163

Appendix E Troubleshooting165

 Troubleshooting Tips166

 Senders and Receivers Falling out of Sync with Incremental Control Numbers166

Index167

Figures

Figure 1	Features of TIBCO BusinessConnect EDI Protocol powered by Instream	3
Figure 2	Data Element Specification in a Guideline	17
Figure 3	New Guideline - Select Guideline	20
Figure 4	New Guideline - Select Transaction Sets	21
Figure 5	Edit Element	22
Figure 6	Generate Map Files	22
Figure 7	BusinessConnect EDI and ActiveMatrix BusinessWorks	35
Figure 8	ActiveMatrix BusinessWorks Process Interacting with BusinessConnect	37
Figure 9	CAQH Real-time Interaction	53
Figure 10	CAQH Batch Submission Request	55
Figure 11	CAQH Batch Retrieval Request	56
Figure 12	CAQH Real-time Interaction	58
Figure 13	Batch Submission Request	59
Figure 14	Batch Retrieval Request	60
Figure 15	Initiator Outbound Request	103
Figure 16	Initiator Inbound Response	110
Figure 17	Responder Inbound Request	115
Figure 18	Responder Outbound Synchronous Response	122

Tables

Table 1	General Typographical Conventions	xvi
Table 2	Values for the ediconvert Script	28
Table 3	EDI Protocol Setup Tasks	32
Table 4	Memory Tuning Features	64
Table 5	FTP-GET File Name Mask Syntax	75
Table 6	Outbound Transport File Name Mask Syntax	77
Table 7	Ack File Name Mask Syntax	78
Table 8	Status Codes	81
Table 9	Common Audit Log Columns	84
Table 10	X12 and EDIFACT Audit Log Columns	85
Table 11	TEXT Audit Log Columns	85
Table 12	Gateway Audit Log Columns	85
Table 13	TRADACOMS Audit Log Columns	86
Table 14	Transaction Detail Columns	86
Table 15	Transaction Detail Columns	87
Table 16	State Detail Columns	90
Table 17	EDIFACT CONTRL Acknowledgment Action Codes	91
Table 18	997 X12 Functional Acknowledgment	91
Table 19	TA1 X12 Interchange Acknowledgment	92
Table 20	Non-Repudiation Log Columns	94
Table 21	Message Queue Log Columns	95
Table 22	Resendable States	97
Table 23	Initiator Request	103
Table 24	InitiatorResponse	110
Table 25	TransactionAckFromTradingPartner	113
Table 26	ResponderRequest	115
Table 27	ResponderInterchange	119
Table 28	ResponderInterchangeComplete	121

Table 29	ResponderResponse.	123
Table 30	ResponderAck.	125
Table 31	CAQHMessage	127
Table 32	TimeoutAlert	128
Table 33	ValidationAlert	129
Table 34	Error	131
Table 35	Insight Report	132
Table 36	CSUpdateNotify Object	133
Table 37	CSUpdateDetailInfo Object	134
Table 38	TPName Object.	136
Table 39	EnvelopeInfo Object	136
Table 40	InterchangeInfo Object	137
Table 41	GroupInfo Object.	137
Table 42	Identity Object	137
Table 43	ControlNumberInfo Object.	138
Table 44	TransactionInfo Object	138
Table 45	ConversionEngineInfo Object	139
Table 46	Attachment Object.	140
Table 47	EDI Plugin Property Reference	142
Table 48	HTTP Status Codes	148
Table 49	TIBCO BusinessConnect EDI Protocol powered by Instream Status Codes	149
Table 50	Runtime Schema Details.	157
Table 51	TIBEDI_BATCH_TASK: Details	158
Table 52	TIBEDI_BATCH_TASK: Columns Summary (Sheet 1 of 3)	158
Table 53	TIBEDI_BATCH_TX	159
Table 54	TIBEDI_BATCH_TX: Columns Summary	160
Table 55	TIBEDI_BATCH_TX_BIN: Details.	163
Table 56	TIBEDI_BATCH_TX_BIN: Columns Summary	164

Preface

TIBCO BusinessConnect™ EDI Protocol powered by Instream® is the TIBCO business-to-business (B2B) solution for transferring EDI documents between trading partners. This manual describes how to use TIBCO BusinessConnect EDI Protocol powered by Instream.

Topics

- [Related Documentation, page xiv](#)
- [Typographical Conventions, page xvi](#)
- [TIBCO Product Documentation and Support Services, page xviii](#)

Related Documentation

This section lists documentation resources you may find useful.

TIBCO BusinessConnect EDI Protocol powered by Instream Documentation

The following documents form the TIBCO BusinessConnect EDI Protocol powered by Instream documentation set:

- *TIBCO BusinessConnect EDI Protocol powered by Instream Installation* Read this manual to learn about installing and deploying TIBCO BusinessConnect EDI Protocol powered by Instream.
- *TIBCO BusinessConnect EDI Protocol powered by Instream User's Guide* Read this manual for instructions on using the product to configure all the EDI protocols.
- *TIBCO BusinessConnect EDI Protocol powered by Instream EDIFACT Configuration* Read this manual for instructions on configuring the EDIFACT protocol.
- *TIBCO BusinessConnect EDI Protocol powered by Instream Gateway Configuration* Read this manual for instructions on configuring the Gateway protocol.
- *TIBCO BusinessConnect EDI Protocol powered by Instream Service Configuration* Read this manual for instructions on configuring the Service protocol.
- *TIBCO BusinessConnect EDI Protocol powered by Instream TEXT Configuration* Read this manual for instructions on configuring the TEXT protocol.
- *TIBCO BusinessConnect EDI Protocol powered by Instream TRADACOMS Configuration* Read this manual for instructions on configuring the TRADACOMS protocol.
- *TIBCO BusinessConnect EDI Protocol powered by Instream X12 Configuration* Read this manual for instructions on configuring the X12 protocol.
- *TIBCO BusinessConnect EDI Protocol powered by Instream Release Notes* Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release.

Other TIBCO Product Documentation

You may find it useful to read the documentation for the following TIBCO products:

- TIBCO ActiveMatrix BusinessWorks™
- TIBCO ActiveMatrix BusinessWorks™ Plug-in for BusinessConnect™
- TIBCO Administrator™
- TIBCO BusinessConnect™
- TIBCO BusinessConnect™ Palette
- TIBCO Business Studio™
- TIBCO Designer™




Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
<i>ENV_HOME</i> <i>TIBCO_HOME</i>	<p>TIBCO products are installed into an installation environment. A product installed into an installation environment does not access components in other installation environments. Incompatible products and multiple instances of the same product must be installed into different installation environments.</p> <p>An installation environment consists of the following properties:</p> <ul style="list-style-type: none">• Name Identifies the installation environment. This name is referenced in documentation as <i>ENV_NAME</i>. On Microsoft Windows, the name is appended to the name of Windows services created by the installer and is a component of the path to the product shortcut in the Windows Start > All Programs menu.• Path The folder into which the product is installed. This folder is referenced in documentation as <i>TIBCO_HOME</i>.
<i>TIBEDI_HOME</i>	<p><i>TIBCO BusinessConnect EDI Protocol powered by Instream</i> installs into a directory within a <i>TIBCO_HOME</i>. This directory is referenced in documentation as <i>TIBEDI_HOME</i>. The default value of <i>TIBEDI_HOME</i> depends on the operating system. For example, on Windows systems, the default value is C:\tibco\bc\version\protocols\tibedi.</p>
code font	<p>Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:</p> <p>Use MyCommand to start the foo process.</p>
bold code font	<p>Bold code font is used in the following ways:</p> <ul style="list-style-type: none">• In procedures, to indicate what a user types. For example: Type admin.• In large code samples, to indicate the parts of the sample that are of particular interest.• In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, MyCommand is enabled: MyCommand [enable disable]

Table 1 General Typographical Conventions (Cont'd)

Convention	Use
<i>italic font</i>	<p>Italic font is used in the following ways:</p> <ul style="list-style-type: none"> To indicate a document title. For example: See <i>TIBCO BusinessConnect EDI Protocol powered by Instream Installation</i>. To introduce new terms. For example: A portal page may contain several portlets. <i>Portlets</i> are mini-applications that run in a portal. To indicate a variable in a command or code syntax that you must replace. For example: <code>MyCommand PathName</code>.
Key combinations	<p>Key names separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C.</p> <p>Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.</p>
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

TIBCO Product Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, or join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website mainly in the HTML and PDF formats.

The TIBCO Product Documentation website is updated frequently and is more current than any other documentation included with the product. To access the latest documentation, visit <https://docs.tibco.com>.

Documentation for TIBCO BusinessConnect EDI Protocol powered by Instream is available on the <https://docs.tibco.com/products/tibco-businessconnect-edi-protocol-powered-by-instream> Product Documentation page.

How to Contact TIBCO Support

You can contact TIBCO Support in the following ways:

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit <http://www.tibco.com/services/support>
- For accessing the Support Knowledge Base and getting personalized content about products you are interested in, visit the TIBCO Support portal at <https://support.tibco.com>.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to <https://support.tibco.com>. If you do not have a user name, you can request one by clicking Register on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the [TIBCO Ideas Portal](#). For a free registration, go to <https://community.tibco.com>.

Chapter 1 **Introduction**

This chapter gives an introduction to EDI and TIBCO BusinessConnect EDI Protocol powered by Instream.

Topics

- [EDI Standard, page 2](#)
- [Product Overview, page 3](#)
- [Protocol Variants, page 4](#)
- [Transports, page 6](#)
- [Security, page 8](#)
- [Validation and Conversion, page 9](#)
- [Integration, page 12](#)
- [Messages, page 13](#)

EDI Standard

Electronic Data Interchange (EDI) is a standardized messaging framework developed by industry groups for exchanging information between trading partners in a structured, pre-determined format.

Early electronic interchanges were based on proprietary formats agreed upon between two trading partners. Because of differing document formats, it was difficult for a company to exchange data electronically with many trading partners. A standard format for data exchange was needed.

In the late 1970s, work began on national EDI standards. Together, users and vendors created a set of data formats that were hardware-independent and unambiguous, and which reduced the labor-intensive tasks of exchanging data and allowed the sender of the data to control the exchange.

Although today many syntaxes exist for EDI, American National Standards Institute (ANSI) X12 and UN Electronic Data Interchange For Administration, Commerce are the most widely recognized. ANSI X12 is an EDI standard developed and used mainly in the United States.

Standards are mandated for use within the United States government.

Product Overview

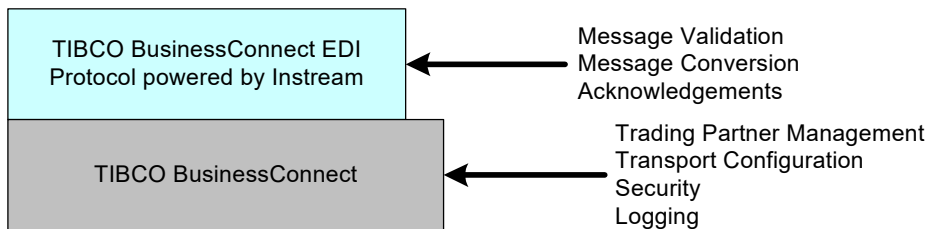
TIBCO BusinessConnect EDI Protocol powered by Instream is the TIBCO B2B solution for transferring EDI documents between trading partners. It can be used for all EDI integration scenarios, including connecting directly to trading partners and connecting to value added networks (VANs).

TIBCO BusinessConnect EDI Protocol powered by Instream offers advantages in the following situations:

- A company wants to migrate from an existing EDI translator to a comprehensive B2B integration server.
- A company wants to integrate its existing EDI system with other XML-based systems that it is already using.
- A company wants to focus on process-oriented B2B integration to allow legacy data exchange to transition over time from EDI to XML.
- A company must adapt to the EDI format preferred by its trading partners.

TIBCO BusinessConnect EDI Protocol powered by Instream provides for the exchange of EDI documents with your trading partners. As shown in [Figure 1](#), it augments features of TIBCO BusinessConnect with protocol-specific features such as message validation and conversion, and acknowledgments.

Figure 1 Features of TIBCO BusinessConnect EDI Protocol powered by Instream



Protocol Variants

TIBCO BusinessConnect EDI Protocol powered by Instream includes several protocol variants, which provide support for the leading EDI industry standards.

- EDIFACT

Provides support for UN/EDIFACT.

See *TIBCO BusinessConnect EDI Protocol powered by Instream, EDIFACT Configuration* for more information about configuring and using this variant.

- Gateway

Provides support for batching documents and communicating with VANs.

See *TIBCO BusinessConnect EDI Protocol powered by Instream, Gateway Configuration* for more information about configuring and using this variant.

- Service

Provides Framework service for EDI to access information from TIBCO BusinessConnect to aggregate information.

See *TIBCO BusinessConnect EDI Protocol powered by Instream, Service Configuration* for more information about configuring and using this variant.

- TEXT

Supports conversion from TEXT to XML, TEXT to EDI, and from XML to TEXT both for the delimited and positional files.

See *TIBCO BusinessConnect EDI Protocol powered by Instream, TEXT Configuration* for more information about configuring and using this variant.

- TRADACOMS

Provides the necessary capability of converting TRADACOMS documents to XML and XML documents to TRADACOMS.

See *TIBCO BusinessConnect EDI Protocol powered by Instream, TRADACOMS Configuration* for more information about configuring and using this variant.

- X12

Provides support for ANSI X12.

See *TIBCO BusinessConnect EDI Protocol powered by Instream, X12 Configuration* for more information about configuring and using this variant.

NAESB NAESB Internet ET stands for North American Energy Standards Board Internet Electronic Transport. Its earlier version, GISB EDM - GISB (Gas Industry

Standards Board) electronic Delivery Mechanism standard was an EDI transport mechanism to support reliable and secure transactions over the Internet.

Only the X12 protocol is supported for NAESB Internet ET, which is designed to transfer all data types: text, EDI, or XML. However, NAESB is used only by energy industry for transmitting EDI messages using the X12 protocol. For more information about the packaging standard NAESB Internet ET, see NAESB in *TIBCO BusinessConnect EDI Protocol powered by Instream X12 Configuration*, .

CAQH The Committee on Operating Rules for Information Exchange (CORE[®]) Connectivity & Security Subgroup of the Council for Affordable Quality Healthcare (CAQH) has defined envelope and submitter authentication standards to improve interoperability and utilization between healthcare providers, clearinghouses, health plans as part of its Phase II and Phase IV Connectivity Rules. TIBCO BusinessConnect can package outbound messages according to these standards. While CAQH Phase II CORE Connectivity standards are only supported on the X12 protocol and the HTTPS transport, CAQH Phase IV CORE Connectivity standards are supported over SSL based authentication. See HIPAA Transaction Configuration in *TIBCO BusinessConnect EDI Protocol HIPAA Edition powered by Instream Configuration* for more detailed information about CAQH Phase II and Phase IV CORE connectivity standards.

Transports

TIBCO BusinessConnect EDI Protocol powered by Instream offers different transport features.

It mainly provides the following transport features:

- AS1, AS2, HTTP/S, POP/SMTP, FTP/S, FILE, SSHFTP, and Inbox transport protocols.

All variants (EDIFACT, Gateway, TEXT, TRADACOMS, and X12) of the TIBCO BusinessConnect EDI Protocol powered by Instream except for the Service protocol support the use of the Inbox transport to exchange files with partners who are using the following products:

- TIBCO BusinessConnect™ Plug-in for SSH Server
- TIBCO BusinessConnect™ Plug-in for FTP Server
- TIBCO PartnerExpress™

- Outbound batching of transactions for X12 and EDIFACT messages directly to a single trading partner or multiple trading partners through a transporting, Gateway to a VAN.
- Scheduled transmission of outbound EDI documents at regular time intervals.
- Custom FTP scripts are used to control the retrieval of files from an FTP server using FTP-GET and/or the storage of files on an FTP server using the outbound FTP transport. These also provide the ability to perform pre- and post-processing from within a custom FTP script.
- File name masks to control which inbound files FTP-GET retrieves in an FTP server directory and the naming of outbound files sent using FTP or FILE.
- Synchronous request-reply transaction type for X12. This is mainly used in processing real-time HIPAA transaction sets over HTTP.
- Partner and transaction-level routing of inbound documents.
- Send and receive EDI documents with optional message validation and XML conversion.
- Outbound EDI file request pollers to monitor predefined file locations for validation and transmitting outbound EDI documents generated by legacy systems with automatic splitting of interchange envelopes.

Set Up Trading Partner Transport via Inbox

TIBCO BusinessConnect EDI Protocol powered by Instream supports exchange of files with TIBCO BusinessConnect Plug-in for SSH Server, TIBCO BusinessConnect Plug-in for FTP Server, and TIBCO PartnerExpress.

When transport can is configured as Inbox, all variants of TIBCO BusinessConnect EDI Protocol powered by Instream except for the Service protocol are able to exchange files with partners who install the mentioned products.

Support for FTP and SSH Server Plug-ins

On outbound, trading partners can use an FTP client to log into a FTP or an SSH server and download EDI data from an Inbox folder.

On inbound, trading partners can use an FTP client to log on to a FTP or an SSH server, and upload EDI data into the corresponding protocol Outbox folder. TIBCO BusinessConnect EDI Protocol powered by Instream automatically picks up EDI data from the Outbox folder, and processes it as inbound EDI data.

Support for TIBCO PartnerExpress

On outbound, trading partners can log on to TIBCO PartnerExpress administrative GUI, and download EDI data posted in the Inbox folder.

On inbound, TIBCO BusinessConnect EDI Protocol powered by Instream automatically picks up EDI data uploaded from the TIBCO PartnerExpress administrative GUI, and processes it as inbound EDI data.

Security

TIBCO BusinessConnect EDI Protocol powered by Instream offers different security features.

It provides the following security features:

- MIME and S/MIME public message packaging protocols.
- Message and payload encryption (S/MIME) with DES3, RC2-40, RC2-128, AES-128, AES-192, and AES-256 algorithms.
- Signing with digital signatures (S/MIME) used for authentication and Non-repudiation.
- Custom FTP scripts used to support communication with Secure FTP servers are implemented over Secure Socket Layer (SSL) and Transport Layer Security (TLS).

Validation and Conversion

TIBCO BusinessConnect EDI Protocol powered by Instream offers different validation and conversion features.

Logging

- Logging and publication of user-defined keys for each transaction in the audit trail database.
- Logging of raw inbound and outbound EDI documents to a file system organized by trading partners and the sending/receiving dates.
- Logging of raw inbound EDI segments to a file extends to the interchange level.
- Logging of raw inbound EDI segments to a file extends to the transaction level, with optional inclusion of the envelope (ISA) segments into the output raw EDI transaction file.
- Audit, non-repudiation, and messaging queue logs.

Validation

- Validation only mode for inbound EDI documents without XML conversion.

Control Numbers

- Control number management for inbound and outbound EDI documents.
- User-defined control numbers for interchange, group, and transaction envelopes.

Modifiers

- Empty Group and Name modifiers. You can create multiple transaction bindings for one transaction type to target transactions to specific departments at a trading partner.
- Transaction modifiers. You can create multiple transaction definitions with different guidelines and XSDs for one transaction type to target different requirements for different trading partners.

Utilities

- Utility to test your processes before engaging in e-commerce.

Detection

- Duplicate detection of transactions for inbound EDI documents based on digest of the raw EDI segment for each transaction and the corresponding envelope information.

Other

- Pre-processor to strip out unwanted characters from inbound EDI files.
- Automatic X12 997, 999 or EDIFACT CONTRL acknowledgment time-out alerts for transmitted outbound X12 transactions or EDIFACT messages.
- Re-submission of inbound EDI documents for message validation and XML conversion using the resend facility available in the Audit Log viewer.
- Pre-configuration and embedding of standard guidelines for interchanges, groups and acknowledgment (997, TA1, 999 and CONTRL) when customization is not required.
- EDI Conversion Engine, a run-time EDI-to-XML, TEXT-to-EDI, and XML-to-EDI Conversion Engine that is part of TIBCO BusinessConnect EDI Protocol powered by Instream.
- Publication of raw EDI segments of the transaction body along with the converted XML.
- Automatic generation of acknowledgments for inbound EDI documents.
- Automatic reconciliation of inbound EDI acknowledgments with the original outbound EDI document.
- Generation of acknowledgments for inbound EDI message on inbound validation errors only.
- Encoding support for X12, EDIFACT, TRADACOMS, and TEXT protocols:
 - For outbound XML to EDI conversion, all encoding types are supported. You can specify an encoding type for outbound XML to EDI conversion.
 - For inbound EDI data, TIBCO BusinessConnect EDI Protocol powered by Instream identifies encoding automatically. If inbound EDI data does not contain the BOM header, you can specify an encoding type to convert EDI data to XML.
- Access control through trading partner identification and permissions.
- Conditional denial of inbound EDI transactions from inbound documents that are not signed or encrypted by trading partners.
- UNOA, UNOB, and UNOC character sets.
- Wrap data feature is supported for the TEXT protocol so that when it is checked, no extra LF/CRLF breaks are added at the end of each segment. When this option is unchecked, each segment will start in a new line.

EDI Data Formats

Different EDI data formats are supported.

- X12 North American ANSI Standard
- X12N Health Care
- UN/EDIFACT United Nations EDI Trade
- TRADACOMS data format Article Numbering Association UK
- User-defined Delimited and Positional files
- VICS EDI Retail Industry
- HIPAA Health Insurance Portability and Accountability Act

Integration

TIBCO BusinessConnect EDI Protocol powered by Instream offers different integration features.

It provides the following integration features:

- TIBCO ActiveMatrix BusinessWorks activities to send and receive EDI documents and receive advisory messages for acknowledgment alerts and validation errors.
- Compatible with TIBCO BusinessConnect Palette for TIBCO ActiveMatrix BusinessWorks to integrate back-end systems for exchanging EDI documents with your trading partners.

For a list of the features available with each instance of TIBCO BusinessConnect, see *TIBCO BusinessConnect Trading Partner Administration*, "Features."

Messages

In a transaction performed by TIBCO BusinessConnect EDI Protocol powered by Instream, two partners exchange business documents over the Internet based on the predefined rules. These protocol standards specify what message formats and transport protocols the partners have agreed to use.

For more information on the partner agreement, see *TIBCO BusinessConnect Trading Partner Administration*, "Business Agreements Overview." The exchange of business documents is known as the *process flow*. In any process flow, two types of messages are exchanged: private and public messages

Private Messages and Processes

Private messages are exchanged between private processes and TIBCO BusinessConnect. They can contain a request, response, or notification document. For a detailed description see [Chapter 8, Private Messages, page 101](#).

Private processes map the data from your application systems to TIBCO BusinessConnect EDI Protocol powered by Instream and from it to your application systems. The following types of private processes can be used:

- **Standalone**

Standalone private processes use TIBCO Rendezvous Certified Messaging or JMS to communicate with TIBCO BusinessConnect EDI Protocol powered by Instream. Windows examples are available in the following directories:

`BC_HOME/protocols/tibedi/samples/scenario/sender/edisend.exe`

`BC_HOME/protocols/tibedi/samples/scenario/receiver/edircv.exe`

- **TIBCO ActiveMatrix BusinessWorks**

These processes can either send requests to or receive replies from a TIBCO BusinessConnect EDI Protocol powered by Instream server.

For more information, see [Creating Private Processes on page 35](#) tutorial chapters in the EDI protocol configuration guides:

- *TIBCO BusinessConnect EDI Protocol powered by Instream, EDIFACT Configuration*, Chapter 2, Tutorial — Getting Started
- *TIBCO BusinessConnect EDI Protocol powered by Instream, Gateway Configuration*, Chapter 2, Tutorial — Getting Started
- *TIBCO BusinessConnect EDI Protocol powered by Instream, Service Configuration*, Chapter 2, Tutorial — Getting Started
- *TIBCO BusinessConnect EDI Protocol powered by Instream, TEXT Configuration*, Chapter 2, Tutorials — Getting Started
- *TIBCO BusinessConnect EDI Protocol powered by Instream, TRADACOMS Configuration*, Chapter 2, Tutorials — Getting Started
- *TIBCO BusinessConnect EDI Protocol powered by Instream, X12 Configuration*, Chapter 2, Tutorials — Getting Started

Public Messages

TIBCO BusinessConnect care of the public process exchanges documents with a trading partner over the Internet using the message formats and protocols specified by the EDI protocol standards.

Chapter 2 **Managing EDI Guidelines**

This chapter describes how to edit, customize, and export EDI guidelines using the EDISIM utility, and how to store the guideline and schema files in the Configuration Store.

Topics

- [Overview, page 16](#)
- [EDI Guideline Files, page 17](#)
- [Managing Guidelines with EDISIM, page 19](#)
- [Exporting Guideline and Schema Files, page 24](#)
- [Storing Guidelines and Schemas in the Configuration Store, page 25](#)
- [EDI Conversion Engine, page 27](#)
- [EdiConvert Utility, page 28](#)

Overview

You can use TIBCO BusinessConnect EDI Protocol powered by Instream to exchange messages with your trading partners and to validate the contents of those messages.

When you consider validation, you must remember that even though EDI is a standard, not all trading partners choose to exchange messages strictly according to this standard. Each trading partner slightly modifies the contents of the messages to satisfy their business requirements.

One possibility is that your trading partners belong to an industry that develops its own set of messages, which are derived from another protocol component. For example, the health care industry in the USA used X12 transaction sets as the basis for HIPAA transaction and code sets.

TIBCO BusinessConnect EDI Protocol powered by Instream is designed to be flexible and can work with any standard. This is accomplished through the use of *guideline files*, which contain a description of the EDI data structures like segments, loops, elements, and code sets that must be contained in the EDI data. They are created by using the TIBCO Foresight® EDISIM® tool.

Separate guideline files must be created for the transactions expected in the EDI data and these guidelines contain the interchange, group and transaction segments as part of the Standards Exchange Format also called SEF format. These transaction guidelines can be used as is, or they can be customized to match the deviations from the EDI standards used by your trading partners. You have to import guideline files into the Configuration Store before configuring the interchange, groups, and transactions used in your system.

Any guidelines that are edited or customized and then saved by using EDISIM Standards Editor are available and listed in the Standards Editor either under User Guidelines, Both Published & User, or under Recently Used. For more information on how to edit guidelines by using this tool, see [Customizing a Guideline on page 20](#).

However, if you intend to use custom guidelines that are coming from another source, you first need to import them.

EDI Guideline Files

Guideline files contain most of the information necessary for the EDI Conversion Engine of TIBCO BusinessConnect EDI Protocol powered by Instream to validate and convert EDI messages.

This section contains a brief introduction to guideline files. See [Storing Guidelines and Schemas in the Configuration Store, page 25](#) for detailed information on how to develop guideline files.

Once a guideline file has been created and the appropriate customization have been applied, the guideline is saved to a file. The default file extension for guideline files is .sef or .std.

Message Guidelines

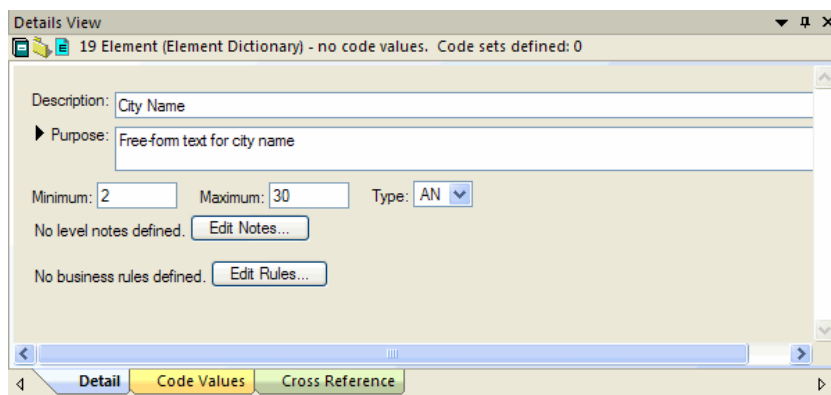
All of the elements used in a message, for example, segments, composite elements, data elements, codelists, are represented in the guideline file.

Detailed information about each element is also contained in the guideline. For example, the specification for a data element consists of:

- The data element type and whether it is required or optional to use it
- How many times the element can be repeated
- The minimum and maximum length of the element

[Figure 2](#) shows an example of a data element specification in a guideline.

Figure 2 Data Element Specification in a Guideline



A guideline is created to encompass the whole message along with the interchange and group envelope segments. TIBCO BusinessConnect EDI Protocol powered by Instream supports customizing these guidelines using the EDISIM Standards Editor and then importing the customized guidelines to the Transaction Leaf node where either the .sef or .std format guidelines can be imported.

Managing Guidelines with EDISIM

You can use EDISIM to manage guidelines.

The guideline management tool EDISIM consists of the following components:

- EDISIM Standards Editor that creates and edits guidelines, and creates new base standards
- EDISIM Doc Builder that prints finished guidelines
- EDISIM Test Data Generator that generates test data
- EDISIM Standards Reference and EDISIM Library that are used for browsing
- EDISIM Validator that checks XML and flat file data
- EDISIM Analyzer that checks EDI data
- EDISIM Comparator that compares guidelines to other transaction sets, or migrates changes from one guideline to another

The guidelines in the database adhere strictly to the various EDI specifications. The database contains the versions of guidelines for most of the EDI specification versions. It is usually easiest to customize a version of a guideline retrieved from the guideline database shipped with EDISIM. If your particular version of a message is not in the database, you can always customize a version.

You can always create new guidelines, but this is a very time consuming process. To facilitate the creation and customization of guidelines, a database of guideline files is included as part of EDISIM.

Starting EDISIM Standards Editor

To manage guidelines, you must start EDISIM Standards Editor.

1. Click **EDISIM > Standards Editor**.
2. Choose whether you want to:

Create a new guideline You have an option to base the new guideline upon the following guidelines that you can modify for your needs:

- Published guidelines
- User guidelines
- Empty guidelines

Open an existing guideline or published standard When you decide to open an existing guideline or published standard to create a new guideline, you have an option to base it upon the following guidelines that you can modify for your needs:

- Published guidelines
- User guidelines

In addition to these guideline groups, two additional tabs can offer the following additional guideline selections:

- Both published and user guidelines
- Recently used guidelines

All EDISIM components are documented in detail in the online documentation provided with the product. For the purpose of this guide, only a set of basic guideline management tasks is presented to help you start using the tool.

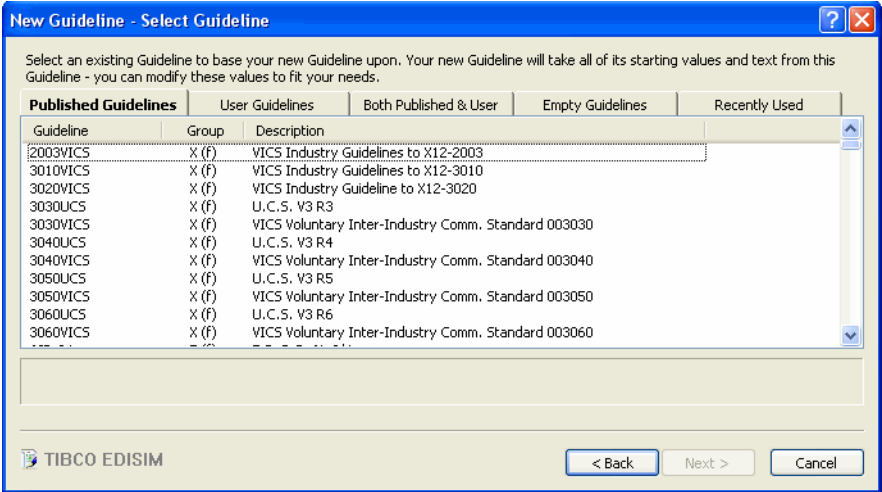
Both when creating a new guideline and when opening an existing guideline or standard, start by choosing a guideline in the database that is close to the version you need.

Customizing a Guideline

You can customize a guideline as required.

Figure 3 shows the EDISIM graphical interface to the guideline database. The **New Guideline - Select Guideline** dialog is displayed after you click **File > New** in the EDISIM Standards Editor.

Figure 3 New Guideline - Select Guideline

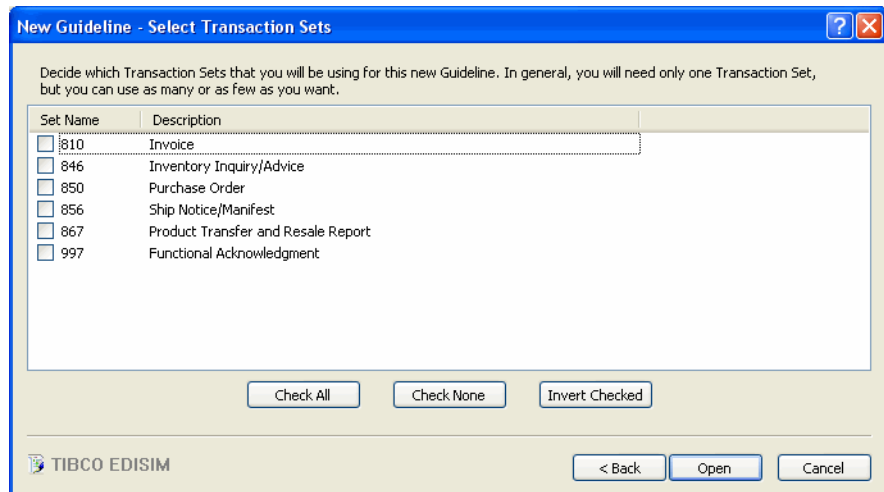


To customize a guideline:

1. Select the group of guidelines, which you want to select a specific one from:
 - Published Guidelines
 - User Guidelines
 - Both Published and User
 - Empty Guidelines (blank guidelines with no data)
 - Recently Used
2. Click **Next**.

The **New Guideline - Select Transaction Sets** dialog is displayed.


Figure 4 New Guideline - Select Transaction Sets



Available options depend on a specific guideline.

3. Select one or more transaction sets that are used with this guideline, and then click **Check All**, **Check None**, or **Invert Checked** for bulk selections.
4. Click **Open**.

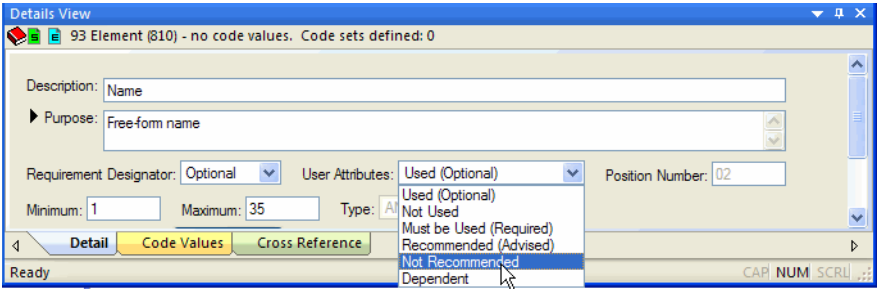
The selected guideline is displayed with the description that corresponds to the transaction set you select, such as **Invoice**.

5. Click the plus sign (+) next to **Transaction Set**  **810**.

The Transaction Set objects are displayed, such as Header, Reference Numbers, and so on.

6. Click the plus sign (+) next to the object that you want to edit, such as Administrative Communications Contact, and also select the specific element to edit, such as Name.

Figure 5 Edit Element



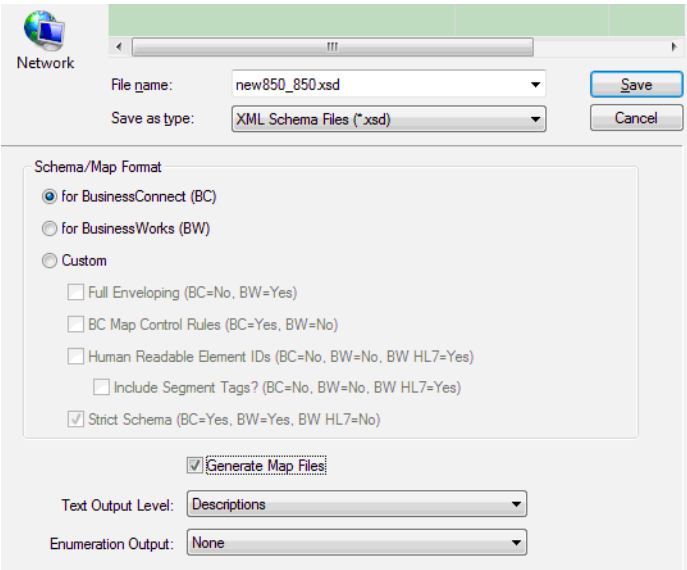
7. Edit any of the available categories, such as **User Attributes**.
8. After you enter all your changes, click **File > Export > Export Current Guideline >** (available export format).

In the example, the changed guideline can be exported as a Schema.

9. Save the exported file, in this case a .xsd schema file, in a directory from which you are able to use it when working with TIBCO BusinessConnect EDI Protocol powered by Instream.

While saving the file, you can also generate map files.

Figure 6 Generate Map Files



Define the Text Output Level and Enumeration Output Level for the new map files.

10. All saved files are now placed in the chosen directory:

../schema_name.xsd

../schema_name_EX.map

../schema_name_XE.map

Exporting Guideline and Schema Files

You can configure TIBCO BusinessConnect EDI Protocol powered by Instream to convert XML documents into EDI documents and EDI documents into XML documents.

Inbound EDI documents can be converted to XML for processing by your backend processes. Your backend processes can also pass outbound XML data to TIBCO BusinessConnect EDI Protocol powered by Instream for conversion to EDI data prior to sending the data to your trading partners.

When TIBCO BusinessConnect EDI Protocol powered by Instream converts a document into XML, any application that uses the XML document, for example, TIBCO ActiveMatrix BusinessWorks, has to know the structure of the XML to expect. Therefore, after you complete customizing a guideline, you also have to generate an XSD schema file, which reflects the structure of the guideline.

While the guideline is open in EDISIM, you can export the XSD for the guideline by clicking **File > Export > Export Current Guideline > To Schema**, and you can also export the guideline by clicking **File > Export > Export Current Guideline/Export Multiple Guidelines > To SEF**.

With EDISIM, you can export the guideline in several different formats:

- Multiple guidelines can be exported to the SEF format.
- A specific guideline can be exported to any location into SEF, or XSD Schema formats.

The exported XSD is configured into TIBCO BusinessConnect along with the guideline file.

The XSD can be retrieved from the TIBCO BusinessConnect Configuration Store by TIBCO BusinessConnect Palette in TIBCO Designer/TIBCO Business Studio. Then, you can use the XSD in TIBCO Designer or TIBCO Business Studio when assembling your processes for mapping and routing of the XML messages to and from TIBCO BusinessConnect.

Storing Guidelines and Schemas in the Configuration Store

Use the **Operations Editor** tab of the configuration GUI to import and further configure guidelines. After configuration, the guideline file is stored in the TIBCO BusinessConnect Configuration Store.

At run time, the EDI Conversion Engine retrieves the guideline file from the Configuration Store when it receives a message corresponding to the guideline. The guideline is used to validate messages.

Import a Guideline File into the Configuration Store

Import the .sef or .std guideline file into the TIBCO BusinessConnect Configuration Store when you configure a transaction guideline. Select the guideline file when you click **change** for the **Transaction Guideline** field.

For information on importing guideline files, see the appropriate sections of the relevant configuration guides.

Import a Schema File into the Configuration Store

Import the schema file into the TIBCO BusinessConnect Configuration Store when you configure a transaction guideline.

Select the schema file when you click **change** for the **Transaction Schema** field. You must also input the **Request Root Element Name** field for the root element of the XSD of the transaction that they are exchanging documents with the trading partner.

For information on importing schema files, see the appropriate sections of the relevant configuration guides.

Import Map files into the Configuration Store

For XML to EDI and EDI to XML translations, import the map files into the TIBCO BusinessConnect Configuration Store when you configure a transaction.

Select the **EDI to XML translation Map file (.map)** and **XML to EDI translation Map file (.map)** fields to import the *_EX.map and *_XE.map files.

For information about importing map files, see the appropriate sections of the relevant configuration guides.

Confirm that the Configuration Store is configured correctly

Use EdiConvert to confirm that the Configuration Store is configured correctly. See [EdiConvert Utility on page 28](#).

Test your configuration

After configuring your host and trading partner information, you can test your configuration with sample messages.

To do this test, you can either start the TIBCO BusinessConnect engine or use a utility included with TIBCO BusinessConnect EDI Protocol powered by Instream to test your configuration off-line.

With the `ediconvert` utility, you can confirm that your Configuration Store is configured correctly and that your guideline works properly with your messages. Additionally, you can use the main functionality of the EDI Conversion Engine without starting TIBCO BusinessConnect.

EDI Conversion Engine

The EDI Conversion Engine in TIBCO BusinessConnect EDI Protocol powered by Instream performs various tasks.

- Inbound EDI documents
 - Validates EDI data using the .sef or .std guideline file.
 - Converts each EDI file into separate XML objects for each transaction in the inbound file.
 - Parses EDI files and extracts the various fields.
 - Creates functional acknowledgments as needed.
- Outbound EDI documents
 - Converts XML objects to EDI.
 - Validates EDI data using the .sef or .std guideline file.



Outbound XML to EDI conversion is available for the X12, TEXT, TRADACOMS, and EDIFACT standards.

The EDI Conversion Engine uses EDI guideline files in the Configuration Store to parse, validate, and convert inbound EDI documents to XML and outbound XML to EDI documents.

Outbound validation occurs for both the envelope and the transaction information. For example, for X12, the interchange and group level fields are retrieved from the entries in the TIBCO BusinessConnect console. These fields are validated for outbound messages.

The interchange and group level fields are retrieved from the TIBCO BusinessConnect console only if the outbound document is XML.

See [EdiConvert Utility on page 28](#) for a description of how to use EdiConvert to test your configuration, guidelines, and XML before conducting e-commerce.

EdiConvert Utility

With the EdiConvert utility, you can use the main functionality of the EDI Conversion Engine without starting TIBCO BusinessConnect.

After importing your guideline file into the Configuration Store, you can use EdiConvert to confirm that the Configuration Store is configured correctly and that your guideline works properly with your XML or EDI data. You can use EdiConvert to test your configuration by converting EDI .dat files into XML files and XML files into EDI .dat files.



In case you are using the MySQL database, you have to update the JDBC_PATH with the exact MySQL driver.

To convert files by using EdiConvert:

1. Start the TIBCO BusinessConnect Configuration Store database if it is not running.
2. Navigate to the `BC_HOME/protocols/tibedi/tools/ediconvert` directory.
3. Edit the `ediconvert` script, providing values for environment variables.

For details, see [Table 2](#).

4. Start `ediconvert`.

Table 2 Values for the ediconvert Script

Value	Required Y/N	Definition
DATA_MODE	Required	The type of conversion. Use xml for XML to EDI, or edi for EDI to XML.
INPUT_FILE	Required	The input file to be processed.
OUTPUT_FILE_PREFIX	Required	The prefix of output files. All output files are prefixed with this value.
TP_NAME	Required	The trading partner name.
HOST_NAME	Optional	The trading host name. If not specified, the default host is assumed.
OPERATION_ID	Required only for XML mode	The operation name.
EDI_STANDARD	Required	The standard name, for example, X12, TEXT, TRADACOMS, or EDIFACT.

Table 2 Values for the ediconvert Script (Cont'd)

Value	Required Y/N	Definition
DIRECTION	Optional	Whether to treat the file as an inbound or outbound file. It determines whether the trading partner is the sender (in) or receiver (out).
DB_VENDOR	Required	The vendor of the Configuration Store database.
JDBC_URL	Required	The JDBC URL for connecting to the Configuration Store database.
JDBC_DRIVER	Required	The class name of the JDBC driver used to connect to the Configuration Store database.
JDBC_CLASSPATH	Required	The path to the JAR containing the driver.
JDBC_USER	Required	The database user name.
JDBC_PWD	Required	The password for the database user.

Chapter 3 **Exchanging Documents**

This chapter describes the steps involved in setting up TIBCO BusinessConnect EDI Protocol powered by Instream for the exchange of documents with your trading partners.

Topics

- [Overview, page 32](#)
- [Setting Up Trading Partners, page 34](#)
- [Creating Private Processes, page 35](#)
- [Exchanging URI Definitions, page 38](#)
- [Exchanging Identity Information, page 39](#)

Overview

Setup of TIBCO BusinessConnect EDI Protocol powered by Instream for document exchange requires several steps.

Table 3 lists the tasks required to set up TIBCO BusinessConnect EDI Protocol powered by Instream for document exchange.

Table 3 EDI Protocol Setup Tasks

Step #	Step Definition	Software	Tasks
1	Managing EDI Guidelines	TIBCO BusinessConnect Configuration GUI	<ul style="list-style-type: none">• Configure EDI transactions• Load EDI guidelines• Load XSDs• Load translation maps
2	Setting Up Trading Partners	TIBCO BusinessConnect Configuration GUI	<ul style="list-style-type: none">• Configure hosts• Configure trading partners
3	Creating Private Processes	TIBCO ActiveMatrix BusinessWorks	<ul style="list-style-type: none">• Connect to TIBCO BusinessConnect• Retrieve XSDs from TIBCO BusinessConnect• Create processes• Map data
4	Exchanging URI Definitions		
5	Exchanging Identity Information		

As you set up TIBCO BusinessConnect EDI Protocol powered by Instream for document exchange, you might also want to consider whether the location of the TIBCO BusinessConnect shared temporary directory is appropriate; TIBCO BusinessConnect EDI Protocol powered by Instream makes extensive use of this directory.

For more information about the shared temporary directory, see *TIBCO BusinessConnect Server Administration Guide*, "Configure Large, Shared, and Temp File Locations."



When using TIBCO BusinessConnect EDI Protocol powered by Instream with TIBCO PartnerExpress™, TIBCO BusinessConnect Plug-in for SSH Server™, or TIBCO BusinessConnect Plug-in for FTP Server™, you must upload EDI documents through the **EDI > Inbound > Interchange** operation.

Setting Up Trading Partners

The next step in preparing to run TIBCO BusinessConnect EDI Protocol powered by Instream is to specify your host and trading partner information.

- Host information includes your interchange identifier and qualifier.
- Trading partner information includes:
 - Interchange identifier and qualifier
 - Transport settings
 - Batching options
 - Information on whether to convert from XML to EDI messages and the reverse

The following sections discuss EDI-specific trading partner configuration options. For other host and trading partner configuration information, see *TIBCO BusinessConnect Trading Partner Administration Guide* and the configuration guides for the EDI protocol standards:

- *TIBCO BusinessConnect EDI Protocol powered by Instream, EDIFACT Configuration*
- *TIBCO BusinessConnect EDI Protocol powered by Instream, Gateway Configuration*
- *TIBCO BusinessConnect EDI Protocol powered by Instream, TEXT Configuration*
- *TIBCO BusinessConnect EDI Protocol powered by Instream, TRADACOMS Configuration*
- *TIBCO BusinessConnect EDI Protocol powered by Instream, Service Configuration*
- *TIBCO BusinessConnect EDI Protocol powered by Instream, X12 Configuration*

EDI Information Not In Guidelines

The EDI-specific information not included in the message guidelines is specified for each trading partner using the TIBCO BusinessConnect configuration GUI. You have to enter information in several tabs under the **Participants > Protocols** tab:

- Interchange Header
- Group Header
- Acknowledgments
- Delimiters

While configuring your trading partner, you have to review the field settings in these panels.

Creating Private Processes

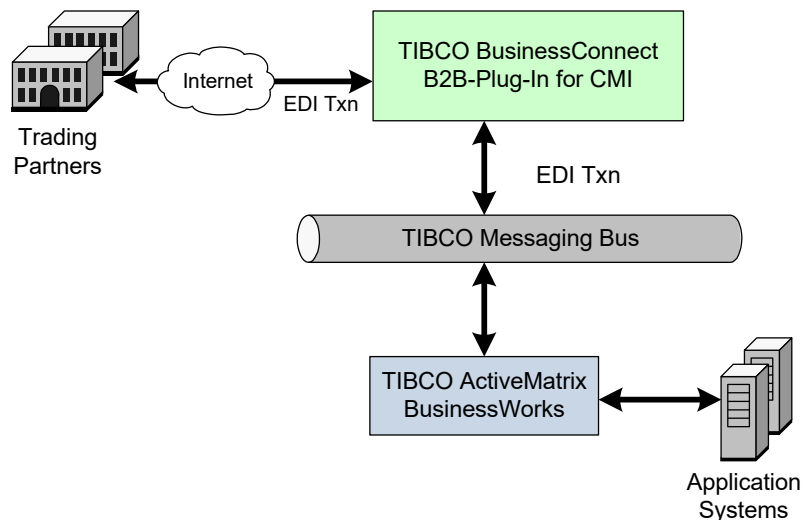
The final step in configuring TIBCO BusinessConnect EDI Protocol powered by Instream for EDI message exchange is creating private processes to interface to the protocol.

TIBCO BusinessConnect EDI Protocol powered by Instream will exchange EDI messages with your trading partners, but it does not perform mapping or routing of transaction message data to your internal applications.

Use TIBCO Designer or TIBCO Business Studio to design and implement private processes or you can use TIBCO Rendezvous or TIBCO Enterprise Message Service™ programming API to implement your standalone private process.

Figure 7 shows the flow of messages through a system using TIBCO BusinessConnect EDI Protocol powered by Instream and TIBCO ActiveMatrix BusinessWorks.

Figure 7 BusinessConnect EDI and ActiveMatrix BusinessWorks



You can use TIBCO Designer or TIBCO Business Studio to create TIBCO ActiveMatrix BusinessWorks processes, which act as private processes to map and route the data. TIBCO BusinessConnect EDI Protocol powered by Instream can exchange messages with TIBCO ActiveMatrix BusinessWorks in XML or EDI format.

TIBCO BusinessConnect Activities

You can use TIBCO BusinessConnect activities to enable communication between TIBCO ActiveMatrix BusinessWorks processes and TIBCO BusinessConnect engine.

You must install TIBCO BusinessConnect Palette or TIBCO ActiveMatrix BusinessWorks Plug-in for BusinessConnect in your existing TIBCO ActiveMatrix BusinessWorks installation before trying to design processes that interface to TIBCO BusinessConnect EDI Protocol powered by Instream.



You can import TIBCO BusinessConnect activities into TIBCO Designer and TIBCO Business Studio.

Before importing TIBCO BusinessConnect activities, perform either of the following steps:

- Install TIBCO BusinessConnect Palette if you want to import TIBCO BusinessConnect activities into TIBCO Designer.
- Install TIBCO ActiveMatrix BusinessWorks Plug-in for BusinessConnect if you want to import TIBCO BusinessConnect activities into TIBCO Business Studio.

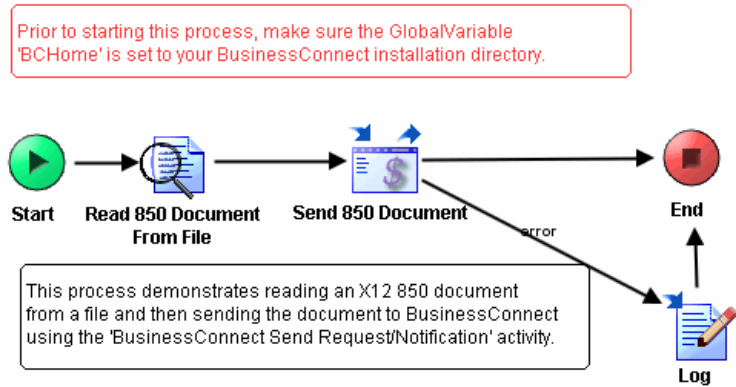
After installing the required software, the TIBCO BusinessConnect activities are displayed in TIBCO Designer or TIBCO Business Studio. You can use one of these activities to connect to the TIBCO BusinessConnect Configuration Store and import the XSDs for your guidelines into TIBCO Designer or TIBCO Business Studio. These XSDs are then used for mapping XML data between your application systems and TIBCO BusinessConnect EDI Protocol powered by Instream.

For further information on the activities available for interfacing from TIBCO Designer to TIBCO BusinessConnect EDI Protocol powered by Instream, see *TIBCO BusinessConnect Palette User's Guide*.

For further information on the activities available for interfacing from TIBCO Business Studio to TIBCO BusinessConnect EDI Protocol powered by Instream, see *TIBCO ActiveMatrix BusinessWorks Plug-in for BusinessConnect*.

[Figure 8](#) shows a process in TIBCO Designer that interacts with TIBCO BusinessConnect EDI Protocol powered by Instream.

Figure 8 *ActiveMatrix BusinessWorks Process Interacting with BusinessConnect*



Exchanging URI Definitions

Partners must exchange Uniform Resource Identifiers (URIs) as part of the partner agreement before they can transact e-commerce. Trading partners can trade URI definitions by email, the web or any other method. The URI is part of the trading partner's URL.

For instructions on how to enter the URI when setting up your trading partner, see the section of configuring transports in the TIBCO BusinessConnect EDI Protocol powered by Instream configuration guides.

The following information is a list of the available transport protocols and their URI formats:

- **EMAIL** URI format: `mailto:username@domain`

Example: `mailto:johndoe@tibco.com`

- **FILE** URI format `file://dir/dir`

The directory must be present.

- **FTP** URI format `ftp://server:port/dir/dir`

The directory can be absent.



For FILE and FTP, the file name must be specified in the file mask field. See [File Masks on page 73](#) for information on how to specify the file mask field in FILE and FTP transport.

- **HTTP** URI format: `http://server:6700/XXX`

where *XXX* is the TIBCO BusinessConnect EDI Protocol powered by Instream protocol variant: X12, EDIFACT, TEXT, TRADACOMS, or Gateway.

Example: `http://www.gizmo.com:6700/X12`

Exchanging Identity Information

Trading partners must exchange information that they can use to identify the other partner when conducting e-commerce. Trading partners can exchange identity information by email, the web, or any other method.

The identity information that you must share with a trading partner includes the following information:

- Trading partner name

See the sections of setting up trading partners in the relevant configuration guide for information on where you add your partner's company name in the TIBCO BusinessConnect console.

- Interchange qualifiers and identifiers

Each trading partner involved in the exchange of EDI documents has a interchange qualifier and ID. The qualifier indicates the type of the ID. For example, a qualifier of 01 indicates that the ID is a D-U-N-S number.

After qualifiers and IDs are set up in the TIBCO BusinessConnect console, the values are used as the interchange sender or interchange receiver qualifier and ID.

See the sections of setting up trading partners in the relevant configuration guide for information on where you add your partner's interchange qualifier and identifier.

- Public certificates

Trading partners have to exchange their public certificates for encryption and authentication. Your trading partner public certificate is used to encrypt outgoing messages and authenticate incoming messages signed by your trading partner.

See *TIBCO BusinessConnect Trading Partner Administration Guide* for information on security in general, certificates files, and instructions on how to set credentials in the TIBCO BusinessConnect console.

Chapter 4 **Configuring Protocol Features**

This chapter describes the key features in TIBCO BusinessConnect EDI Protocol powered by Instream.

Topics

- [Overview, page 42](#)
- [Validation and XML Conversion, page 43](#)
- [TEXT to EDI Conversion, page 44](#)
- [Control Number Management, page 45](#)
- [Validation Error Reporting, page 50](#)
- [Acknowledgment and Reconciliation, page 51](#)
- [Transaction Batching and Scheduling, page 52](#)
- [CAQH Connectivity Rules, page 53](#)
- [Tuning for Improved Performance, page 64](#)

Overview

TIBCO BusinessConnect EDI Protocol powered by Instream provides several features, all of which can be used to implement EDI exchanges between business partners. Not all features will be used in all configurations.

For example, though TIBCO BusinessConnect EDI Protocol powered by Instream includes validation and conversion, if an alternate validation and conversion method is already present in the system, that existing method can continue to be used along with its other components.

This chapter contains a description of the main features and how to use them to exchange files between trading partners.

Validation and XML Conversion

You can configure TIBCO BusinessConnect EDI Protocol powered by Instream to perform EDI syntax checking of the X12, TEXT, TRADACOMS, or EDIFACT documents you exchange with your trading partners. This product is designed to be flexible and can work with any EDI standard based upon X12, TEXT, TRADACOMS, or EDIFACT.

You can also configure TIBCO BusinessConnect EDI Protocol powered by Instream to convert XML documents into EDI documents and EDI documents into XML documents. Inbound EDI documents can be converted to XML for processing by your private processes.

Your private processes can also pass XML data to TIBCO BusinessConnect EDI Protocol powered by Instream for conversion to EDI data prior to sending the data to your trading partners.



Outbound XML to EDI conversion is available for the X12, TEXT, TRADACOMS, or EDIFACT standard.

EDI guideline files contain the information required by the validation engine to perform its syntax checking. A guideline file is created for each EDI transaction set or message type exchanged with your trading partners. Guideline files can be customized to suit your business needs.

For more information, see [Customizing a Guideline, page 20](#).

TRADACOMS to XML

For inbound transfers, TIBCO BusinessConnect EDI Protocol powered by Instream validates TRADACOMS data from the trading partner and converts it to XML data.

For outbound transfers, TIBCO BusinessConnect EDI Protocol powered by Instream converts XML data to the TRADACOMS format and sends it to trading partner, or directly sends TRADACOMS data to the trading partner.

TEXT to EDI Conversion

Conversion of files in TEXT format to EDI is performed in the following situations:

- In the X12 or EDIFACT protocol, when converting TEXT data in a Notify operation to EDI format.

When trading partners want to implement TEXT to EDI conversion, they have to enable both protocols (TEXT and X12, or TEXT and EDIFACT).

TIBCO BusinessConnect EDI Protocol powered by Instream converts TEXT data to X12 or EDIFACT data, and then send the data to the trading partner.

- In the X12 protocol, when converting TEXT data in a Synchronous Request Response operation to EDI format.

When the Responder receives a Synchronous Request and has to send back a Synchronous Response. TIBCO BusinessConnect EDI Protocol powered by Instream converts TEXT data to the response EDI data and send it back to the Initiator.

There are two options for outbound TEXT to EDI conversion in the TEXT protocol operation editor **Guideline** tab:

- **None**
- **TEXT to EDI** When this option is selected, TIBCO BusinessConnect EDI Protocol powered by Instream automatically recognizes the target EDI type (X12 or EDIFACT) from the TEXT to EDI translation map file.

All features for X12/EDIFACT protocol such as batching, ack reconciliation, are available for TEXT to EDI conversion.

Control Number Management

Control numbers are used to uniquely identify the interchanges, groups, and transactions that you exchange with a trading partner.



Control numbers are available for the X12, TRADACOMS, and EDIFACT standards.

- **Interchange control numbers** These numbers used by one trading partner uniquely identify each interchange from that trading partner.
- **Group control numbers** These numbers uniquely identify the functional group within an interchange. For a particular trading partner, the interchange control and functional group control number together uniquely identify each functional group.
- **Transaction control numbers** These numbers uniquely identify each transaction within a functional group. The interchange control number, functional group control number, and transaction control number together uniquely identify each transaction for a particular trading partner.

TIBCO BusinessConnect EDI Protocol powered by Instream provides the following control number management capabilities:

- Automatic generation of control numbers
- Reset of control numbers for all qualifier IDs.
- Beginning control number specification
- Incrementing of control numbers
- Control number sequencing checking

Control Number Validation

Control numbers are used to identify EDI transactions. For example, any acknowledgment must indicate what EDI transaction it is acknowledging.

A control number is used to indicate the transaction. Also, if a problem occurs with a transaction, trading partners contact each other and note that control number *X* transaction had a problem.

For information on how to use control numbers to reconcile acknowledgments, see [Reconciling Acknowledgments on page 70](#). The fields are found in the X12 protocol Control Numbers tabs. See the Setting Control Number Properties sections in the configuration guides for these protocols.

Increment Group Control Numbers Across Outbound Interchanges

This field determines whether to generate group control numbers that always increase when sending EDI documents.

For example, if it is set to false, the pattern of control numbers are similar to the following information:

```
int 1
grp 1
txn 1
txn 2
grp 2
txn 1
int 2
grp 1 <---- the group number has been reset
txn 1
```

If it is set to true, the pattern looks like the following information:

```
int 1
grp 1
txn 1
grp 2
txn 1
int 2
grp 3 <---- the group number continues to increase
txn 1
```

Increment Transaction Control Number Sequentially Across Interchanges

This field determines whether to generate the transaction control numbers that always increase when sending EDI documents across interchanges.

For example, if it is set to false, the pattern of control numbers are similar to the following information:

```
int 1
grp 1
txn 1
txn 2
grp 2
txn 1
int 2
grp 1
txn 1 <---- the transaction number has been reset
```

If it is set to true, the pattern looks like the following information:

```
int 1
grp 1
txn 1
grp 2
txn 2 <---- the transaction number continues to increase
int 2
grp 1
```

txn 3 <---- the transaction number continues to increase

Group Control Numbers Increment Across Inbound Interchanges

This field determines whether group control numbers are expected to increase when receiving EDI documents. This is important only if the control number validation (Group Control Number Sequence Check) is on.

If the field is set to true and the Group Control Number Sequence Check is set to Incremental or Chronological, the following sequence comes in:

```
int 1
grp 1
txn 1
txn 2
grp 2
txn 1
int 2
grp 1 <---- the group number has been reset
txn 1
```

This results in a validation failure. The group number must be 3 (for Incremental) or >2 (for Chronological). The group control numbers can always increase because of acknowledgment reconciliation. 997s only acknowledge at the group and transaction level, not the interchange level.

Any trading partner that only sends back 997s (and not TA1) might have reconciliation problems unless the group control numbers always increase. For example, without setting the field to true, the following sequence is ambiguous for 997-only acknowledgment reconciliation:

```
int 1
grp 1
txn 1
int 2
grp 1
txn 1
```

Since 997s have no interchange information, int 1/grp 1/txn 1 looks exactly the same as int 2/grp 1/txn 1.

Interchange Control Number Sequence Check

This field determines how interchange control numbers on inbound must be validated.

If the field is set to Incremental, the following sequence comes in:

```
int 1
grp 1
txn 1
txn 2
grp 2
txn 1
```

```

int 2 <---- this number must be exactly one greater than the      previous interchange any value other than 2 will
fail
validation
grp 1
txn 1

```

For chronological, int 2 can also be any value greater than 1. For none, the value can be anything. When a resend of an interchange is received from a trading partner, the chronological or incremental sequence check of the Control Number is not validated.

Interchange Control Number Seed

This field determines what number to start at for the interchange control number.

Group Control Number Sequence Check

This field determines how group control numbers are validated. It works with the Group Control Number Increment Across Interchanges.

For example, the following sequence might come in:

```

int 1
grp 1
txn 1
grp 2
txn 1
int 2
grp 1
txn 1

```

The preceding information is valid for Incremental or Chronological, but the group is not expected to always increase.

```

int 1
grp 1
txn 1
grp 2
txn 1
int 2
grp 3 <-----
txn 1

```

The preceding information is only valid for Chronological and always increase.

For transaction settings:

```

int 1
grp 1
txn 1
txn 3 <-----
grp 2
txn 1

```

This is only valid for Chronological. If validation for transactions is set to Incremental, the highlighted value must be 2.

When a resend of an interchange is received from a trading partner, the chronological or incremental sequence check of the Control Number is not validated.

Transaction Control Number Sequence Check

When a resend of an interchange is received from a trading partner, the chronological or incremental sequence check of the Control Number is not validated.

Alphanumeric Control Numbers and Validation

If you set control number validation to Incremental or Chronological and try to process control numbers that are not strictly integers, control number validation always fails.

For example, if control number validation is set to anything but None, and you try to process a control number like A123SXX, control number validation fails.

Control Number Validation and Non-Matching Segment Headers and Trailers

In UN/EDIFACT and ASC X12, the envelope segments have control numbers that must match. For example, in ASC X12, the control number for the IEA segment must match the control number specified in the ISA segment. The same is true for the group and transaction envelopes.

If the control number for a segment header passes validation, it is accepted as the next control number even if the segment trailer fails.

For example, if validation is set to Incremental, and the following information arrives:

```
ISA, control number 1
....
IEA, control number 2
```

The preceding interchange fails validation at the IEA segment because $2 \neq 1$. But the next expected interchange control number will be 2.

Generate a Unique Control Number for a Transaction Automatically

When the property named Increment Transaction Control Number Sequentially is added in the **Control Numbers** tab for the X12 and EDIFACT protocol, its default value is `False`.

By selecting the value `True`, this option supports an automatic generation of a unique control number. To select the option, see:

- *TIBCO BusinessConnect EDI Protocol powered by Instream, X12 Configuration, "Increment Transaction Control Number Sequentially across Interchanges"*
- *TIBCO BusinessConnect EDI Protocol powered by Instream, EDIFACT Configuration, "Increment Transaction Control Number Sequentially Across Interchanges"*

Validation Error Reporting

TIBCO BusinessConnect EDI Protocol powered by Instream reports validation errors in different ways.

- Generates EDI acknowledgments reporting the errors
- Creates entries in the audit log indicating the errors
- Sends error advisory messages to the private process

The EDI acknowledgments contain information on the type and location of the validation errors according to each EDI standard. For information on EDI acknowledgments, see [Chapter 5, Acknowledgments and Reconciliation, page 67](#).

The audit log contains more descriptive error information along with the location of the error in the EDI data. When a transaction contains a validation error, the state and status of the last event in the audit log for the transaction will also indicate that a validation error occurred. In this way, the audit log can easily be searched for validation errors.

For information on how to view the audit log, see [Chapter 7, Viewing Logs, on page 79](#).

Error advisory messages to the private process also contain a more descriptive validation error message. The trading partner, transaction, and error location inside of the transaction are all communicated in the error advisory message. The private process can use the advisory to trigger alternate error handling for the transaction in question.

For information on the messages sent to the private process, see [Chapter 8, Private Messages, on page 101](#).

Acknowledgment and Reconciliation

TIBCO BusinessConnect EDI Protocol powered by Instream can be configured to automatically generate EDI acknowledgments for documents received from your trading partners.

The following EDI acknowledgment configurations are available:

- Interchange, group, and transaction acknowledgments
- Interchange and group acknowledgments
- Interchange acknowledgments only
- Group and transaction acknowledgments
- Group acknowledgments only
- None

TIBCO BusinessConnect EDI Protocol powered by Instream will also match any acknowledgments received from your trading partners to the original requests sent to your trading partners. This is known as *ack reconciliation*.

See [Chapter 5, Acknowledgments and Reconciliation, page 67](#) for further information.

Transaction Batching and Scheduling

TIBCO BusinessConnect EDI Protocol powered by Instream provides several options for sending EDI documents to your trading partners. For individual trading partners you can:

- Send each EDI document immediately
- Schedule a transmission window for sending EDI documents
- Schedule a transmission interval for sending EDI documents
- Group EDI documents into a batch prior to sending



Batching is not available for TRADACOMS.

Using the Gateway protocol, it is also possible for EDI documents intended for multiple trading partners to be grouped together and sent to an intermediary partner such as a VAN.

Gateway provides the same options for sending EDI documents as listed for individual trading partners. See *TIBCO BusinessConnect EDI Protocol powered by Instream, Gateway Configuration* for further information.

CAQH Connectivity Rules

TIBCO BusinessConnect EDI protocol powered by Instream supports CAQH connectivity rules, including Phase II (version 2.2.0) and Phase IV (version 4.0.0) of the CAQH connectivity rules.

There are 2 type of envelope specifications - HTTP MIME Multipart and SOAP + WSDL. Phase II (version 2.2.0) connectivity rules support both envelopes, but Phase IV (version 4.0.0) only support SOAP + WSDL envelope.

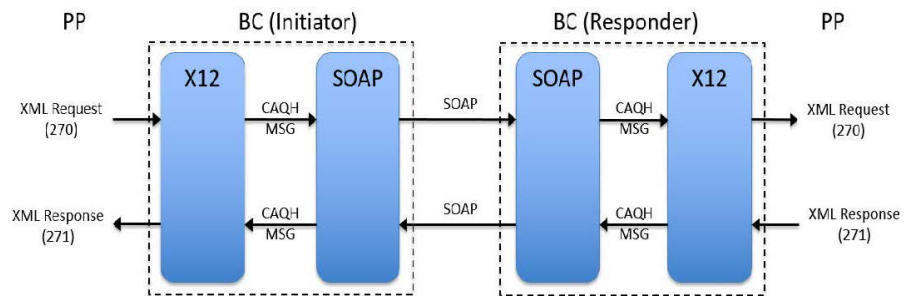
SOAP + WSDL Envelope

The SOAP +WSDL envelope feature combines both TIBCO BusinessConnect EDI protocol powered by Instream and TIBCO BusinessConnect SOAP protocol to handle CAQH requests and supports SOAP + WSDL format for CAQH.

CAQH Real-time Interaction

SOAP synchronize request-response and EDI synchronize request-response operations combine to support CAQH real-time interaction. The following is a sample shown in [Figure 9](#).

Figure 9 CAQH Real-time Interaction



The following is the process flow:

- Private Process (PP) sends an XML request to Initiator BC X12 protocol. X12 protocol converts the XML request to EDI, validates the EDI data, packages it in a CAQH message, and then sends it to the SOAP protocol. The SOAP protocol packages the CAQH message in SOAP body and sends it to Responder BC.

- The SOAP protocol unpacks the SOAP message, extracts the CAQH message, and then sends it to X12 protocol. X12 protocol validates the EDI data, converts it to an XML request, and then sends it to PP.
- PP sends back an XML response to Responder BC. X12 protocol converts the XML request to EDI, validates the EDI data, packages it in a CAQH message, and then sends it to the SOAP protocol. The SOAP protocol packages the CAQH message in SOAP body and sends it to Initiator BC.
- The SOAP protocol unpacks the SOAP message, extracts the CAQH message, and then sends it to X12 protocol. X12 protocol validates the EDI data, converts it to an XML response, and then sends it to PP.

For version 2.2.0, the CAQH message is packaged as SOAP body and for version 4.0.0, the CAQH message is package as a SOAP MTOM attachment.

SOAP operations that are used to exchange CAQH real-time transactions are pre-defined. You can find them in the CAQH samples of the SOAP protocol:

- CAQH/2.2.0/RealTimeTransaction
- CAQH/4.0.0/RealTimeTransaction



You must also configure the corresponding synchronize request-response operations in EDI protocol to exchange CAQH real-time transactions.

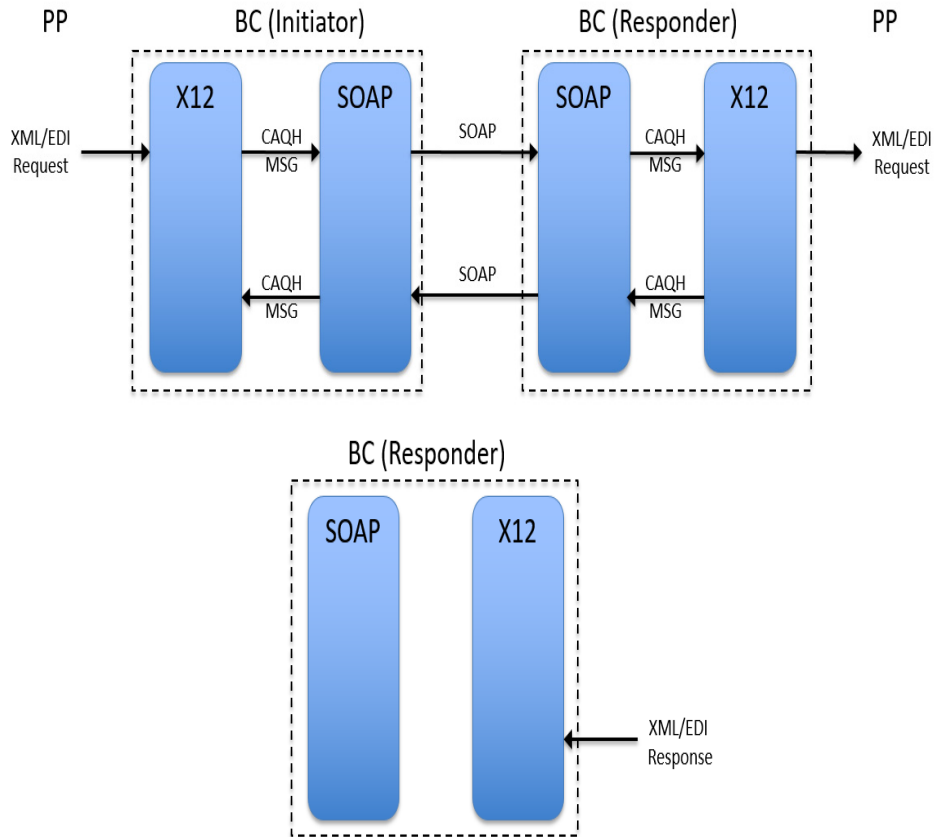
CAQH Batch Interaction

SOAP synchronize request-response and EDI notify operations combine to support CAQH batch interactions.

(A) Batch Submission Request

The process flow of a sample CAQH batch submission request is shown in [Figure 10](#).

Figure 10 CAQH Batch Submission Request



The following is the process flow for a batch submission request:

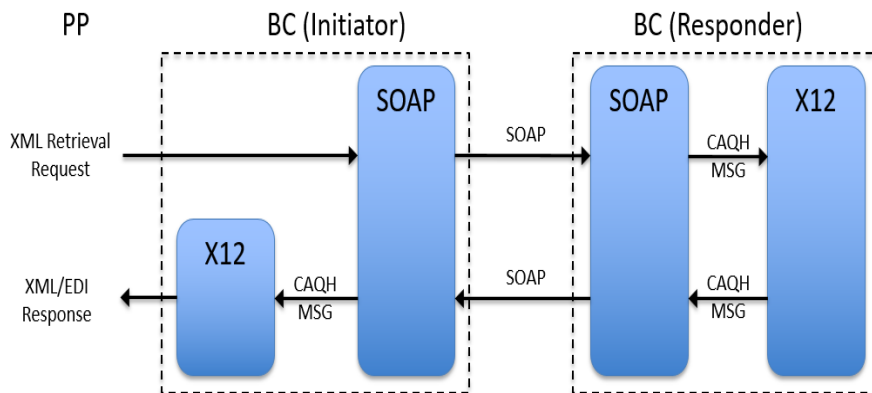
- PP sends an XML (single transaction) or batch EDI request to Initiator BC X12 protocol. X12 protocol converts XML to EDI, validates the EDI data, adds it to the batch or validates the batch EDI request (for directly sending the batch EDI case), packs it in a CAQH message, and sends it to the SOAP protocol. The SOAP protocol packages the CAQH message as MTOM attachment, and sends it to Responder BC.
- The SOAP protocol unpacks the SOAP message, extracts the CAQH message, and sends it to X12 protocol. X12 protocol creates a CAQH response as receipt and sends it back to the SOAP protocol.
- The SOAP protocol packages the CAQH response in a SOAP body and sends it back to the Initiator. The Responder X12 protocol also extracts EDI data from the CAQH request message, validates it, converts it to XML request, and sends it to PP.

- Before the Initiator retrieves a response, Responder needs to send response data to X12 protocol. X12 protocol will hibernate the response and wait for retrieval by the Initiator.

(B) Batch Retrieval Request

The process flow of a sample CAQH batch retrieval request is shown in [Figure 11](#).

Figure 11 CAQH Batch Retrieval Request



The following is the process flow for a batch retrieval request:

- PP sends a CAQH retrieval request to Initiator BC SOAP protocol. The SOAP protocol packages CAQH message in SOAP body and sends the retrieval request to Responder BC.
- The SOAP protocol unpacks the SOAP message, extracts the CAQH message, and then sends it to X12 protocol. X12 protocol retrieves the EDI data from hibernation according to the CAQH retrieval request, packages it as a CAQH message, and then sends it back to the SOAP protocol.
- The SOAP protocol packs the CAQH message as MTOM attachment, and sends it back to Initiator BC. The SOAP protocol unpacks the SOAP message, extracts the CAQH message, and sends to X12 protocol. X12 protocol unpacks the CAQH message, extracts acknowledgment / response data, validates the EDI data, converts EDI to XML, and sends XML to PP, or directly sends EDI to PP. X12 protocol generates acknowledgment for Response data and uses “Batch Submission Request” mode to submit acknowledgment to Responder BC.

SOAP operations that are used to exchange CAQH batch transactions are pre-defined. You can find them in the CAQH samples of the SOAP protocol:

- CAQH/2.2.0/BatchSubmitTransaction
- CAQH/2.2.0/BatchSubmitAckRetrievalTransaction

- CAQH/2.2.0/BatchResultsRetrievalTransaction
- CAQH/2.2.0/BatchResultsAckSubmitTransaction
- CAQH/2.2.0/GenericBatchSubmissionTransaction
- CAQH/2.2.0/GenericBatchSubmissionAckRetrievalTransaction
- CAQH/2.2.0/GenericBatchRetrievalTransaction
- CAQH/2.2.0/GenericBatchReceiptConfirmationTransaction
- CAQH/4.0.0/BatchSubmitTransaction
- CAQH/4.0.0/BatchSubmitAckRetrievalTransaction
- CAQH/4.0.0/BatchResultsRetrievalTransaction
- CAQH/4.0.0/BatchResultsAckSubmitTransaction
- CAQH/4.0.0/GenericBatchSubmissionTransaction
- CAQH/4.0.0/GenericBatchSubmissionAckRetrievalTransaction
- CAQH/4.0.0/GenericBatchRetrievalTransaction
- CAQH/4.0.0/GenericBatchReceiptConfirmationTransaction



Initiator sends a batch response / acknowledgment retrieval request from PP with SOAP protocol.

HTTP MIME Multipart Envelope

This envelope feature uses only TIBCO BusinessConnect EDI protocol powered by Instream to transmit CAQH requests over HTTP/HTTPS transport. HTTP MIME Multipart envelope is defined based on HTTP transport, so you need to configure HTTP or HTTPS transport for trading partner and then specify HTTP/HTTPS transport as the primary transport in Business Agreement before you use HTTP MIME Multipart envelope to exchange CAQH message with your trading partner.

TIBCO BusinessConnect EDI protocol powered by Instream will decide whether the outbound CAQH message is packaged with HTTP MIME Multipart envelope or SOAP + WSDL envelope by identifying the query string provided in the outbound HTTP/HTTPS transport URL.

The query string takes the form:

package = x12-caqh-mime

For example:

server1.company.example.com:6700/X12?package=x12-caqh-mime

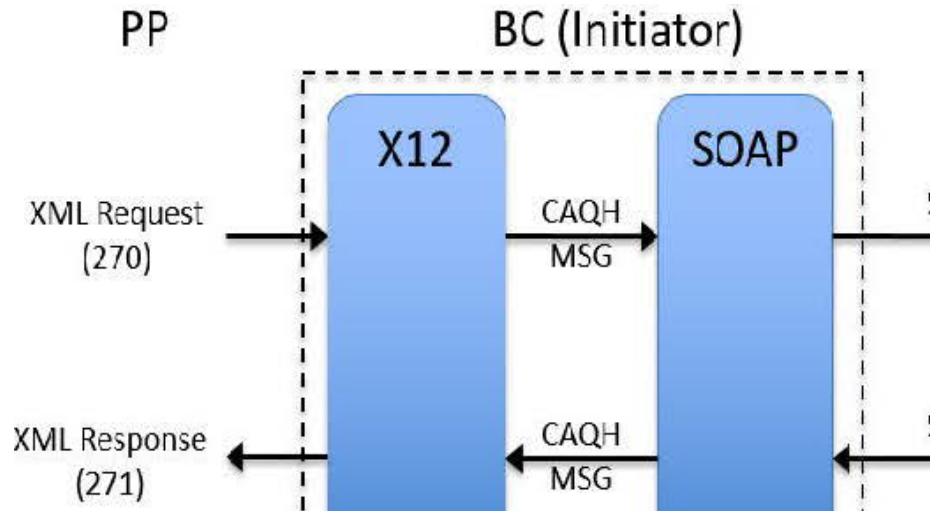


If the query string is not provided in the outbound HTTP URL, TIBCO BusinessConnect EDI protocol powered by Instream will package outbound CAQH message with SOAP + WSDL envelope.

CAQH Real-time Interaction

The process flow of a sample real-time interaction request is shown in [Figure 12](#).

Figure 12 CAQH Real-time Interaction



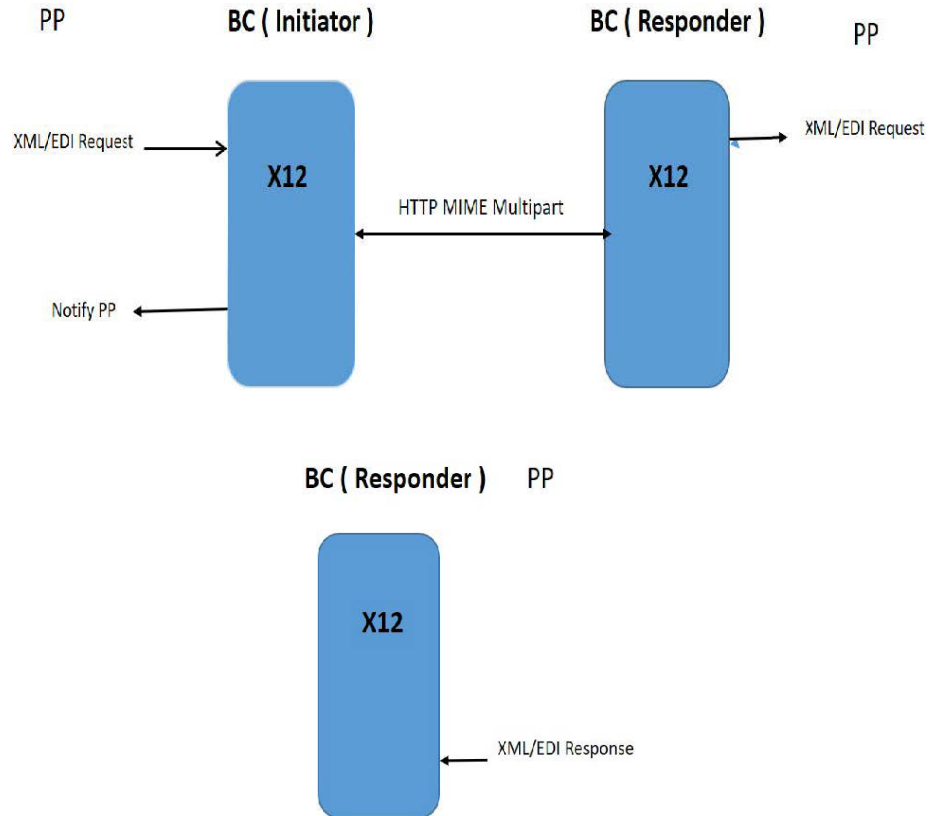
- Private Process (PP) sends an XML request to Initiator BC X12 protocol. X12 protocol converts the XML request to EDI, validates the EDI data, packages the EDI data and CAQH information as HTTP MIME Multipart message, and sends it to Responder BC.
- The Responder BC X12 unpacks the HTTP MIME Multipart message, validates EDI data and CAQH information, converts EDI data into XML request, and then sends it to PP.
- PP sends back an XML response to Responder BC. X12 protocol converts the XML response to EDI, validates the EDI data, packages the EDI data and CAQH information as HTTP MIME Multipart message, after that sends it back to Initiator BC via the same communication channel.
- The Initiator BC unpacks the HTTP MIME Multipart message, validates EDI data and CAQH information, converts EDI data to XML response, and then sends it to PP.

CAQH Batch Interaction

(A) Batch Submission Request

The process flow of a sample CAQH batch submission request is shown in [Figure 13](#).

Figure 13 Batch Submission Request



- PP sends an XML (single transaction) or batch EDI request to Initiator BC X12 protocol. X12 protocol converts XML to EDI, validates the EDI data, adds it to the batch or validates the batch EDI request (for directly sending the batch EDI case), packs the EDI data and CAQH information as HTTP MIME Multipart message and sends the Request to Responder BC.
- The Responder X12 protocol unpacks the MIME Multipart envelope, extracts the EDI batch data, sends back a CAQH response to Initiator BC to confirm that Responder BC received the request, and the Initiator BC sends it back to PP. The Responder X12 protocol validates EDI data, converts it to XML, and sends XML request to PP, or directly sends EDI data to PP.

- Before the Initiator BC retrieves a response, Responder PP needs to send response data to X12 protocol. X12 protocol will hibernate the response and wait for retrieval by the Initiator.



Generic Batch Submission Request uses the same request message as the Batch Submission Request message, with PayloadType values based on what is being submitted.

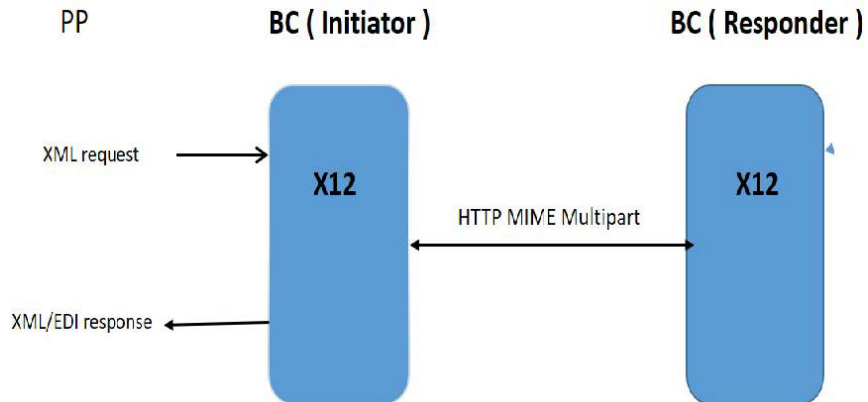
(B) Batch Retrieval Request

The process flow of a sample CAQH batch retrieval request is shown in [Figure 14](#).



The Initiator PP uses operation "EDI/Outbound/Interchange" to send CAQH retrieval request to Initiator BC.

Figure 14 Batch Retrieval Request



- PP sends a CAQH retrieval request to Initiator BC X12 protocol. The Initiator X12 protocol packages the CAQH information as HTTP MIME Multipart message and sends the retrieval request to Responder BC.
- The Responder X12 protocol retrieves the EDI data from hibernation according to the CAQH retrieval request, packages it with CAQH information as HTTP MIME Multipart message, and then sends it back to the Initiator BC.
- The Initiator X12 protocol unpacks the MIME Multipart envelope, extracts the EDI data, validates it, and converts EDI to XML, and sends XML to PP, or directly sends EDI to PP.
- X12 protocol generates acknowledgment for Response Data and uses "Batch Submission Request" mode to submit acknowledgment to Responder BC.



Generic Batch Retrieval Request uses the same request message as the Batch Results Retrieval Request message structure, with different PayloadType values based on what is being retrieved.

Process Flows and Transaction Types

The following is an example of how a TIBCO BusinessConnect EDI Protocol powered by Instream process flow might proceed:

1. A private process inside the Initiator company sends a TIBCO Rendezvous or JMS topic message to TIBCO BusinessConnect. The message contains all fields specified in [Initiator Outbound Request on page 103](#). It includes a document specifying the business operation.
2. TIBCO BusinessConnect EDI Protocol powered by Instream performs the following functions:
 - Validates the document.
 - Converts XML documents to EDI documents.
 - Looks up trading partner information to determine the address to which the document must be sent and the security required.
 - Establishes the necessary security for message transmission.
 - Enters transaction information for the audit and non-repudiation databases.
3. TIBCO BusinessConnect sends the document to the Responder. The Responder receives the message, performs validation as appropriate, and sends out the document as a TIBCO Rendezvous or JMS topic message to the private process. See [Responder Inbound Request on page 115](#).

What happens next depends on whether the transaction is defined as a notification transaction or a request-response transaction.

Notification Transaction

For notify activities, TIBCO BusinessConnect on the Initiator side considers the operation complete as soon as an acknowledgment is received from the Responder TIBCO BusinessConnect.

The Initiator TIBCO BusinessConnect creates a notify message that it sends back to its private process to indicate that a previously sent transaction has been successfully reconciled upon receiving an acknowledgment from the trading partner. See [Initiator Acknowledgment From Partner on page 113](#).

The Responder TIBCO BusinessConnect creates a notify message that it sends back to its private process to indicate the success or failure of the posting of an acknowledgment to the Initiator. See [Responder Acknowledgment to Partner on page 124](#).

Request-Response Transaction

For request-response activity, the Responder site processes the response as follows:

1. The private process must send a message with appropriate response information back to TIBCO BusinessConnect. See [Responder Outbound Synchronous Response on page 113](#).

If the Responder private process is unable to handle the document, it must publish an error response back to the server so that the server can inform the Initiator that the request has failed. See [Responder Outbound Synchronous Response on page 113](#).

2. When the message arrives, the Responder TIBCO BusinessConnect adds appropriate information and converts it to HTTP format, using the specified trading partner, enveloping, and security information. It sends the response to the Initiator TIBCO BusinessConnect.
3. The message arrives at the Initiator TIBCO BusinessConnect. The Initiator verifies and validates the incoming information and converts it to a response message, which it sends to the private process. The response may also be converted into XML documents depending on the configuration.

See [Initiator Inbound Response on page 98](#).

Timeout and Validation Alerts

If at any point an error occurs, TIBCO BusinessConnect EDI Protocol powered by Instream will post one of the following alert messages to the private process:

- [Timeout Alert on page 128](#)
- [Validation Alert on page 129](#)

Tuning for Improved Performance

You can improve performance by tuning memory usage and thread counts. This section provides more information about both.

Memory Tuning

Using the TIBCO BusinessConnect console, you can configure TIBCO BusinessConnect to leverage the file system to decrease memory usage. Depending on the EDI protocol, these features are available:

Table 4 Memory Tuning Features

Feature	Feature Availability				
	Gatew ay	X12	EDIFA CT	TEXT	TRADA COMS
The following features apply to inbound messages only:					
Publish XML and TEXT Request as File Reference		X	X		X
Output XML Request in UTF-8 and Remove Empty Elements		X	X		X
Use In-memory Compression During EDI to XML (or EDI to TEXT) Conversion		X	X		
Threshold to Use File-based Storage During EDI to XML (or EDI to TEXT) Conversion (bytes)		X	X		
The following feature applies to both inbound and outbound messages:					
Use File-based Storage Prior to EDI Processing		X	X		X

For more information about each of these TIBCO BusinessConnect EDI Protocol powered by Instream features, see the section for the host record, advanced properties, in the configuration guide for the appropriate EDI variant.

For information about the memory-tuning available features, see the *TIBCO BusinessConnect Server Administration Guide*.

Tuning Thread Count

For optimal performance, TIBCO BusinessConnect EDI Protocol powered by Instream requires a total thread count of at least 32 in the deployment.

The following EDI plug-in properties allow you to configure a thread count:

- **edi.LongRunningJob.ThreadPool.EDIToXML.FileBased.maxCount:** This property controls the maximum thread pool size used by file based EDI validation and conversion. The default is 8.
- **edi.LongRunningJob.ThreadPool.EDIToXML.InMemory.maxCount:** This property controls the maximum thread pool size used by in-memory based validation and conversion. The default is 8.
- **edi.LongRunningJob.ThreadPool.XMLToEDI.maxCount:** This property controls the maximum thread pool size used by XML to EDI validation and conversion. The default is 8.

Included in the minimum thread count of 32 are sixteen (16) threads used by TIBCO ActiveMatrix BusinessWorks for pollers, batching, and other uses.

See [Property Reference on page 142](#) for a complete list of EDI plugin properties and a procedure for modifying property values.

See the TIBCO ActiveMatrix BusinessWorks documentation for information about configuring thread pool size.

To Improve Performance

- To improve performance, change the property value `Interior_Server.tra` from the `.tra` file.
- The default value of the `java.property.bmh.doc.validation.threads` property is 16. This is done to allow for handling multiple transactions that are sent after processing when a document is received from a trading partner.

Tuning Flow Control

TIBCO BusinessConnect is built on top of TIBCO ActiveMatrix BusinessWorks, which flow control features you can use to throttle at the service configuration of the deployment. For example, when deployed in DMZ mode, you can set the flow control for Max Jobs from the **Advanced** tab of the Service Configuration of Interior Server.par, which is located in the bottom portion of the screen.

These are the TIBCO ActiveMatrix BusinessWorks processes that you can control using flow control:

- **Gateway/BMH/Outbound/From-Back-Office/RV/Receive Request** This is the RVCMQ listener.
 - You can limit the number of jobs and processes that BusinessConnect accepts from the private process at a time. The default is unlimited.
 - You can limit the number of jobs in memory for both the RVCMQ listener and the JMS process. A value of 16 is optimal depending on the number of TIBCO ActiveMatrix BusinessWorks engine threads set during deployment.
- **Gateway/MSH/Outbound/From-BMH/RV Receiver** This RVCMQ listener controls certain kinds of MSH service jobs that can be processed. This is mainly used in EDI for batching, message queue and scheduling. Setting Max jobs at 10 is optimal. By default, the relay processes between multiple BusinessConnect engines are turned off.
- **Gateway/BMH/MiscMsg Receiver** This RVCMQ listener controls the certain kinds of BMH service jobs that can be processed. This is mainly used in EDI for batching, message queue and scheduling. Setting Max jobs at **10** is optimal. The default is unlimited.
- **Gateway/MSH/Inbound/From-DMZ/RV Receiver** This RVCMQ listener controls the number of inbound messages that is received from the partner. The default is to receive unlimited messages. The proper flow control value for max jobs in memory depends on the TIBCO ActiveMatrix BusinessWorks engine ThreadCount: a value of **16** is recommended.
- **Gateway/BMH/Inbound/From-MSH/RV Receiver** This RVCMQ listener controls the number of inbound messages that is received from the MSH sub-component. The default is to receive unlimited messages. A proper flow control value for max jobs in memory should be set: a value of **16** is recommended.

Chapter 5 **Acknowledgments and Reconciliation**

This chapter describes acknowledgments and reconciliation for the X12 and EDIFACT protocols.

Topics

- [Acknowledgments, page 68](#)
- [Acknowledgment Timeouts, page 69](#)
- [Reconciling Acknowledgments, page 70](#)
- [Reconciling Acknowledgments After Timeout, page 72](#)

Acknowledgments

When an initiator using TIBCO BusinessConnect EDI Protocol powered by Instream sends an EDI document to a trading partner, it can request an acknowledgment from the trading partner to verify that the EDI document was accepted or rejected.

TIBCO BusinessConnect EDI Protocol powered by Instream offers the following levels of acknowledgment configurations:

- Interchange, group, and transaction acknowledgments
- Interchange and group acknowledgments
- Interchange acknowledgments only
- Group and transaction acknowledgments
- Group acknowledgments only

Types of Acknowledgments

There are following types of acknowledgments.

For EDIFACT:

- CONTRL Syntax and Service Report Message

For X12:

- TA1 Interchange Acknowledgment
- 997 Functional Group Acknowledgment
- 999 Functional Group Acknowledgment

Acknowledgment Timeouts

When an acknowledgment is requested, TIBCO BusinessConnect EDI Protocol powered by Instream records all outbound transactions that are sent to your trading partner. When an acknowledgment is received from your trading partner, the recorded transactions are reconciled and marked as completed.

If the acknowledgment is not received within a specified timeout value, an advisory is created to alert your backend systems and the transaction is marked with a status indicating a timeout waiting for trading partner acknowledgment. The timeout values can be specified based on the outbound transaction types.

For more information on how to set up the acknowledgment timeout value for each transaction type, see the following:

- [Table 47, EDI Plugin Property Reference, on page 142](#), field `edi.ack.timeout.pollingInterval`
- Table 9, Outbound Action: Acknowledgement Tab in the *TIBCO BusinessConnect EDI Protocol powered by Instream, X12 Configuration*.
- Table 9, Outbound Action: Acknowledgement Tab in the *TIBCO BusinessConnect EDI Protocol powered by Instream, EDIFACT Configuration*.

How Acknowledgment Timeouts Work

TIBCO BusinessConnect EDI Protocol powered by Instream regularly polls the acknowledgment reconciliation table for any EDI transactions waiting to be reconciled. Transactions with the ACK PENDING status in the audit log require the receipt of an acknowledgment from the Responder. If the maximum wait time passes, ACK TIMEOUT ERROR will be logged for this transaction in the audit log. An alert is then sent on the ERROR.TIMEOUT.ACK subject.

TIBCO BusinessConnect EDI Protocol powered by Instream can be configured to try to reconcile an acknowledgment even when it arrives after the timeout. In this case when the maximum wait time passes, ACK PENDING will be kept in audit log. See [Reconcile After Timeout Check box on page 72](#).

This feature does not detect whether the status of an acknowledgment is rejected or accepted, but whether an acknowledgment has been received or not.

Reconciling Acknowledgments

TIBCO BusinessConnect EDI Protocol powered by Instream records all outbound transactions that are sent to your trading partner when an acknowledgment is requested.

When an acknowledgment is received from your trading partner, the recorded transactions are reconciled based on the detail information provided in the incoming acknowledgment message. The recorded transactions are marked as `COMPLETED` upon reconciliation.



When the payload file is a .dat file, acknowledgment reconciliation for the EDIFACT/X12 protocol takes place only if outbound validation is enabled under **Business Connect > Participants > Trading Partner > Protocols > EDIFACT|X12** general page.

When the EMAIL transport is requesting an MDN in cases when the acknowledgment comes before the receipt, reconciliation will take place at the point where this acknowledgment was received (*not* after it was received). This happens because in EMAIL is no control over the order in which the receipt and acknowledgment are received.

How Acknowledgments Are Reconciled

TIBCO BusinessConnect EDI Protocol powered by Instream regularly polls the acknowledgment reconciliation table for any EDI transactions waiting to be reconciled. Transactions with the `PENDING` status require the receipt of an acknowledgment from the Responder.

When an acknowledgment arrives from the Responder, the Initiator tries to match the values in the following fields in the acknowledgment with the values in those same fields in the request message, which was stored in the Initiator acknowledgment reconciliation table when it was sent:

- **tpName**
- **transaction ID**
- **control#** The control number values for the interchange control number, group control number, and transaction control number are automatically generated by the TIBCO BusinessConnect EDI Protocol powered by Instreamengine before the request is sent.

The control number values are stored in the internal acknowledgment reconciliation table for each transaction to enable reconciliation when an acknowledgment is received.

If the Initiator finds a match for all three values, it changes the status of the request message to `COMPLETED` and reconciles the transaction.

TIBCO BusinessConnect EDI Protocol powered by Instream can be configured to reconcile an acknowledgment even when it arrives after the timeout. See [Reconcile After Timeout Check box on page 72](#).

The acknowledgment status of each reconciled transaction displays in the Audit Log Viewer. Look at the acknowledgment contents to see the error cause when the acknowledgment status indicates that an error occurred for a transaction.

Reconciling Acknowledgments After Timeout

You can control how TIBCO BusinessConnect EDI Protocol powered by Instream handles acknowledgments that are received after the timeout period has passed by setting options on the **Acknowledgement** tab of a transaction action.

Reconcile After Timeout Check box

By default, the **Reconcile after Timeout** check box is not selected. In this case, if an acknowledgment does not arrive within the maximum acknowledgment wait time set in the field above, an audit log entry is created with an ACK TIMEOUT ERROR status and ACK_RESPONSE_TIMEOUT state. If the acknowledgment arrives after the timeout, the acknowledgment is not allowed to reconcile.

If the **Reconcile after Timeout** check box is selected, when an acknowledgment does not arrive within the Maximum Wait time, an audit log entry is created with an ACK PENDING status and an ACK_RESPONSE_TIMEOUT state. However, if the acknowledgment is received later, the acknowledgment is supported to reconcile and transaction completes.

See [Acknowledgments Display in the Audit Log on page 90](#) for more information.

Completing Transactions After Timeout

If the **Mark as Completed after Timeout** check box is selected for this type of transaction and an acknowledgment does not arrive within the Maximum Wait time, an audit log entry is created with a COMPLETED status and ACK_RESPONSE_TIMEOUT state. If the acknowledgment arrives after timeout, the acknowledgment is ignored and will not be reconciled. This is useful when your trading partner is sending an acknowledgment only when errors occurred in validating the previously transaction sent. After the reconciliation timeout expires, the transaction is assumed as accepted by the trading partner.

Mark as Completed after Timeout cannot be selected when **Reconcile after Timeout** is selected.

Chapter 6 **File Masks**

You can use file masks to configure how to retrieve files from and send files to a trading partner.

Topics

- [FTP Inbound File Name Masks, page 74](#)
- [FTP and FILE Outbound File Name Masks, page 76](#)
- [Ack File Name Masks, page 78](#)

FTP Inbound File Name Masks

You use inbound file name masks to control which files FTP-GET retrieves from an FTP server. The FTP-GET mask option is available in the FTP Edit Settings dialog, which is accessible in the **Transports** tab for a protocol binding.

See *TIBCO BusinessConnect Trading Partner Administration Guide* for more information.

File Name Mask Examples

If the trading partner name is receiver, and the host name is sender, to generate an inbound file named receiver.sender.12202004-TIBCO.edi on December 20, 2004, enter the following mask in the File Mask field:

```
#{TpName).#{HostName).#{MMDDYYYYY)-TIBCO.edi
```

Some other examples are as follows:

- MyFile.dat. This example specifies the exact name of the file to be retrieved.
- TIBCO_#(YYYYMMDD).dat. This example uses a standard predefined variable for retrieving the file.
- #{var1)_#{var2).dat. This example uses user-defined variables for retrieving the file.
- *. This example uses the asterisk wildcard character (*) to retrieve files. This retrieves all files.
- #{var1)_#{var2)*.dat. If var1 is defined as TIBCO and var2 is defined as X12, this would retrieve all files with names beginning with TIBCO_X12 and ending with .dat.

When wildcard characters are used in the mask, files are retrieved using mget. Not all FTP servers implement mget. Be sure to verify that your trading partner FTP server supports mget before defining file name masks which use the asterisk wildcard character (*).



The following characters are treated as special in EDI for masks: #, (,). Any value between # and first closing parenthesis ')' will be checked for a replacement and, if not found, will be made an empty string.

File Name Mask Syntax

Table 5 lists the mask syntax supported for FTP-GET.



You can use an asterisk (*) in an FTP-GET mask. This acts as a wildcard.

Masking applies only to file names. It does not apply to any directory name in the FTP URL. For example, if the FTP URL is `/local/tibco/bc/#(TPNAME)/REQUEST/`, and `/local/tibco/bc/SELLER-AXNT15/REQUEST/` exists, FTP fails.

Table 5 FTP-GET File Name Mask Syntax

Mask Name	Case Sensitivity	Description
\$(TpName)	Case-insensitive	Trading partner name
\$(HostName)	Case-insensitive	Host name
\$(DDD)	Case-insensitive	Day of the year
\$(MMDDYYYY)	Case-insensitive	MM = Month, DD = Day, YYYY = Year
\$(HHMISSNNN)	Case-insensitive	HH = hour, MI = minute, SS = second, NNN = milliseconds

FTP and FILE Outbound File Name Masks

Use outbound file name masks to control how TIBCO BusinessConnect names outbound files sent using FTP and FILE transport. The mask option is available in the **Transports** tab for each trading partner.

See "FTP Outbound and FILE Outbound" in *TIBCO BusinessConnect Trading Partner Administration Guide*.

If you leave the file mask field blank, TIBCO BusinessConnect constructs the file name according to the following rules:

- Files sent outbound from an Initiator are named *tpName-DocumnetID.request*.
- When acknowledgment files are sent outbound from a Responder, the files are named *tpName-GUID.ack.(1 or 2).response* or *tpName-GUID-TimeStamp.ack.response*. The suffix number depends on the number of acknowledgments.
 - For a single transaction X12 document, because TA1 and 997 are generated, the number is always 1 or 2.
 - For UN/EDIFACT, it is always 1.
- You can also specify file names by using file masks for generated acknowledgment data.

File Name Mask Examples

If the trading partner's name is *receiver*, and the host's name is *sender*, to generate an outbound file called *receiver.sender.12202002-TIBCO.edi* on December 20, 2002, enter the following mask in the Output File Mask field:

```
#(TpName).#(HostName).#(MMDDYYYY)-TIBCO.edi
```

If the trading partner's name is *receiver*, to generate an outbound file named *receiver-12202004-12000003.edi* on December 20, 2004 at 12:00:00.03, specify the mask

```
#(TpName)-#(MMDDYYYY)-#(HHMISSNNN).edi.
```

File Name Mask Syntax

You can use an asterisk (*) in an outbound FILE mask. If a wild card (*) is used in a File or in a FTP-Put outbound file mask, the outbound document will be sent as *hostname-GUID*.

Masking applies only to file names. It does not apply to any directory name in the FTP or FILE URL. For example, if the FTP or FILE URL is `/local/tibco/bc/#(TPNAME)/REQUEST/`, and `/local/tibco/bc/SELLER-AXNT15/REQUEST/` exists, FILE naming would fail.

Table 6 lists the mask syntax supported for FTP and FILE outbound.

Table 6 Outbound Transport File Name Mask Syntax

Mask	Case Sensitivity	Description
TxName	Case-insensitive	Name of the transaction This mask is not valid for use with batched transactions.
TpName	Case-insensitive	Receiving participant name
HostName	Case-insensitive	Sending participant name
DDD	Case-insensitive	Day in a year
YY	Case-insensitive	Last two digits of a year
YYYY	Case-insensitive	Year
MMM	Case-insensitive	Month abbreviated to three characters
MM	Case-insensitive	Month using two digits (1-base)
DD	Case-insensitive	Day of the month using two digits (1-base)
HH	Case-insensitive	Hour of the day (0-24)
MI	Case-insensitive	Minutes of the hour
SS	Case-insensitive	Seconds of the minute
NN	Case-insensitive	Milliseconds of the second using three digits
TransactionID	Case-insensitive	ID of the transaction
FileName	Case-insensitive	Name of the file
GUID	Case-insensitive	Globally unique identifier
DocumentID	Case-insensitive	ID of the sent document
OperationID	Case-insensitive	ID of the operation

Ack File Name Masks

You can also specify file names by using file masks for generated acknowledgment data.

File Name Mask Examples

If the trading partner name is receiver, to generate an acknowledgment file called receiver-Z8m2JKPdfRKJf-RgPGSKfJsUBic-20151020-12000003.ack.response on October 20, 2015 at 12:00:00.03, enter the following mask in the **Ack File Mask** field:

#(TpName)-#(DocID)-#(YYYYMMDDHHMISSNNN).ack.response

File Name Mask Syntax

Table 7 lists the mask syntax supported for generated acknowledgment data.

Table 7 Ack File Name Mask Syntax

Mask	Case Sensitivity	Description
TxName	Case-insensitive	Transaction name
TpName	Case-insensitive	Receiving participant name
DocID	Case-insensitive	ID of the sent document
YYYY	Case-insensitive	Year
MM	Case-insensitive	Month using two digits (1-base)
DD	Case-insensitive	Day of the month using two digits (1-base)
HH	Case-insensitive	Hour of the day (0-24)
MI	Case-insensitive	Minutes of the hour
SS	Case-insensitive	Seconds of the minute
NNN	Case-insensitive	Milliseconds of the second using three digits

Chapter 7 **Viewing Logs**

This chapter discusses how to view audit, non-repudiation, message queue, and resend logs after conducting business transactions.

Topics

- [Overview, page 80](#)
- [Audit Log, page 84](#)
- [Non-Repudiation Log, page 94](#)
- [Message Queue Log, page 95](#)
- [Resend Log, page 97](#)
- [Document Archiving, page 100](#)

Overview

TIBCO BusinessConnect provides several logs which are used to store messages processed through the system. These logs are:

- **Audit log** This log is used to store information about the messages and documents processed by TIBCO BusinessConnect. Using the audit log, one can follow the processing states of inbound or outbound documents.

The audit log also provides information for monitoring inbound messages which are received through the Gateway and redirected for EDI message validation and XML conversion if applicable.

- **Non-repudiation log** This log is used to provide proof of the delivery of messages. Non-repudiation depends on authentication using digital signatures. Incoming messages which have been digitally signed are authenticated and stored in the non-repudiation database.
- **Message queue log** This log is used to store those messages waiting to be transmitted to your trading partners. The message queue log is used in conjunction with transmission scheduling.
- **Resend log** This log displays the audit log transactions eligible to be resent.

Using the Log Viewer

To search for transaction records and access the log, perform the following steps:

1. Click the **BusinessConnect > Log Viewer** link in the left panel.
2. Click a protocol radio button to select which type of log to view:
 - **Audit** Select this to display the audit log search options. See [Audit Log on page 84](#). If you select the protocol link instead the protocol radio button, the audit log will be selected by default.
 - **Non-Repudiation** Select this to display the non-repudiation search options. See [Non-Repudiation Log on page 94](#).
 - **Message Queue** Select this to display the message queue search options. See [Message Queue Log on page 95](#). Available for X12, EDIFACT, and Gateway protocols.
 - **Resendable Transactions** Select this to display the resendable transactions log search options. See [Resend Log on page 97](#).
 - **Resend History** Select this to display the resendable transaction history. See [Resend Log on page 97](#).

3. Invoke a search by following the description provided in *TIBCO BusinessConnect Trading Partner Administration*.

Status Codes

Table 8 contains a list of the status codes that can appear in the transaction details for all log types, a description of what the status code means, and TIBCO BusinessConnect EDI Protocol powered by Instream that supports this status code.

Table 8 Status Codes (Sheet 1 of 3)

Status Code	Description	Protocols
ACK ACCEPTED	<p>The transaction that is sent out has received a positive business acknowledgment that was requested and was successfully completed with no EDI validation errors.</p> <p>This status can/cannot happen depending on the acknowledgment that was expected and the acknowledgment that was created by the Trading Partner.</p>	EDIFACT, X12
ACK ACCEPTED WITH ERRORS	<p>The transaction has received a negative business acknowledgment and was not successfully completed with no EDI validation errors by the partner. This status means that the transaction was accepted, but errors were noted in the transaction.</p> <p>This status can/cannot happen depending on the acknowledgment that was expected and the acknowledgment that was created by the trading partner. The actual error condition is defined in the acknowledgment description and in the ack code that was sent by the partner.</p>	EDIFACT, X12
ACK PENDING	<p>The transaction is waiting for a business acknowledgment from the partner.</p> <p>This status can/cannot happen depending on the acknowledgment that was expected and the acknowledgment that was created by the trading partner.</p>	EDIFACT, X12
ACK REJECTED ERROR	<p>The transaction has received a negative business acknowledgment and was rejected due to EDI validation errors by the partner.</p> <p>This status can or cannot happen, depending on the acknowledgment that was expected and the acknowledgment that was created by the trading partner. This transaction cannot be reconciled after this error.</p>	EDIFACT, X12

Table 8 Status Codes (Sheet 2 of 3)

Status Code	Description	Protocols
ACK TIMEOUT ERROR	The transaction is waiting for the business ack and has not received within the timeout interval. The transaction cannot be reconciled after this error.	EDIFACT, X12
ANY	Any transaction that occurred.	EDIFACT, X12, TEXT, TRADACOMS, Gateway
ANY ERROR	Any error that occurred.	EDIFACT, X12, TEXT, TRADACOMS, Gateway
ANY PENDING	Any transaction that is pending.	EDIFACT, X12, TEXT, TRADACOMS, Gateway
CANCELED	The transaction was scheduled and later deleted from the message queue log.	EDIFACT, X12, Gateway
COMPLETED	The individual summary rows are completed and no further audit trail is pending. It also means that the transaction has completed successfully if there were no acks were expected.	ALL
COMPLETED WITH ERRORS		ALL
DENIED	The transaction cannot be accepted by this partner due to partner/transaction level security being enabled and the message that was received does not follow the security agreement that was agreed between the partners.	EDIFACT, X12, Gateway
ERROR	There were general errors during the processing of the transaction.	ALL
ERROR ACK CREATION	There were errors encountered during the creation of the acknowledgment on inbound. This can be due to missing guidelines.	EDIFACT, X12
ERROR ACK RECONCILIATION	There were problems in reconciling the business acks with the transactions that were sent out. The problems can be several: ack codes are not correct, database problem in retrieving the original ack option that was expected, and so on.	EDIFACT, X12
ERROR CONFIGURATION	There were problems in identifying the partner and cannot retrieve the business agreement.	ALL

Table 8 Status Codes (Sheet 3 of 3)

Status Code	Description	Protocols
ERROR PARSING	There were problems encountered in parsing the EDI document due to syntactic errors in the document.	ALL
ERROR SECURITY	There were errors encountered in digitally signing/authenticating or encrypting/decrypting the EDI document due to bad/wrong public or private keys.	EDIFACT, X12, TEXT, TRADACOMS, Gateway
ERROR TRANSPORT	The transaction could not be sent due to problems contacting the partner through the transport that was configured to be sent.	ALL
ERROR VALIDATION	Indicates there were errors in the EDI validation. Can be used to search for transactions with only the ERROR VALIDATION status.	EDIFACT, X12, TEXT, TRADACOMS
INFO VALIDATION	Indicates that the EDI document has some informational category of errors which do not cause the transaction to be rejected. Can be used to search for transactions with only the INFO VALIDATION status. Used by HIPAA users only.	X12,
PENDING	The transaction has not yet completed all required states.	ALL
RECEIPT PENDING	Indicates that the transaction that was sent is waiting for a transport level receipt from the partner. Business acknowledgments cannot be reconciled even if they are received before the receipt is received. Only occurs when AS1 or AS2 transport is used.	EDIFACT, X12, TEXT, TRADACOMS, Gateway
SCHEDULED	The transactions are scheduled to be sent to the partner.	EDIFACT, X12, TEXT, TRADACOMS, Gateway
WARNING VALIDATION	Indicates that the category of a validation error is a warning. The warning does not cause the EDI transaction to be rejected. Used by HIPAA users only.	X12

Audit Log

The audit log is used to store information about the messages and documents processed by TIBCO BusinessConnect. You can use the audit log to follow the processing states of inbound or outbound documents. Some of the types of information stored in the audit log include:

- Sent and received documents
- Document originator
- Trading partner name
- Processing status
- Validation errors

Summary View

The audit log summary view depends on the protocol.

[Table 9, Common Audit Log Columns](#) lists the columns common to all protocols except Gateway.

The columns specific to each protocol are listed in the following tables:

- [Table 10, X12 and EDIFACT Audit Log Columns](#)
- [Table 11, TEXT Audit Log Columns](#)

Table 9 Common Audit Log Columns

Column	Description
Gateway Instance Information	Indicates which Gateway Instance transferred this message.
Operation ID	A protocol-specific value identifying the type of the transaction. Example: 00403/005010/850
Document ID	A unique value identifying the transaction.
Trading Partner	Trading partner name. Example: Space Sparrows, Inc.

Table 10 X12 and EDIFACT Audit Log Columns

Column	Description
Initiated by Host	Values are TRUE or FALSE
Interchange Qlfr	Interchange qualifier used in the interchange envelope header for this request. Example: ZZ
Interchange ID	Interchange ID used in the interchange envelope header for this request. Example: 123456789
User Key	Values of the user keys delimited by colon (:); if specified in the .sef guideline as an EVALU record using EDISIM Standards Editor Example: PO:90392
Int Control Number	Example: 000000001
Group Control Number	Example: 1
Txn Control Number	Example: 0001

Table 11 TEXT Audit Log Columns

Column	Description
Domain	Domain used in the interchange envelope header for this request. Example: ZZ
Identity	Example: 123456789
Host Initiates	If a transaction is sent via TEXT, this column should be True in the sender audit log and False in the receiver audit log.
User Key	Values of the user keys delimited by colon (:); if specified in the .sef guideline as an EVALU record using EDISIM Standards Editor Example: PO:90392

Table 12 Gateway Audit Log Columns

Column	Definition
Protocol	X12, EDIFACT, TEXT, TRADACOMS


Table 12 Gateway Audit Log Columns (Cont'd)

Column	Definition
Transporting Trading Partner	If this batch is going through a VAN, the Gateway name. This value is the same for each transaction in the batch.
Transaction Set	The type of transaction in the batch. “Any” if this batch contains a mixture of different transaction sets.
Transmission ID	A unique value that identifies the scheduled transmission.
Host Initiates	If a transaction is sent via Gateway, this column should be <code>True</code> in the sender audit log and <code>False</code> in the receiver audit log.

Table 13 TRADACOMS Audit Log Columns

Column	Definition
Transmission Reference	Sender transmission reference generated by TIBCO Business Connect.
Interchange Name	Transmission name used in the interchange envelope header for this request. Example: TRANSMISSION SENDER
Interchange Code	Transmission code used in the interchange envelope header for this request. Example: 5013546000000
User Key	Values of the user keys delimited by colon (:); if specified in the .sef guideline as an EVALU record using EDISIM Standards Editor Example: <i>PO:90392</i>

Transaction Details View

To view the details of a transaction, click the document icon  in the left-most column of an audit log entry.

The Log Viewer first displays the general information for this entry, then a table with available information for each event in that transaction. The table columns are listed in [Table 14](#).

Table 14 Transaction Detail Columns

Column	Description
Time Stamp	Time when the transaction was processed

Table 14 Transaction Detail Columns (Cont'd)

Column	Description
Status	Status of the transaction. See Acknowledgment Messages on page 88 for a list of the possible values and a description of the meaning of the status codes.
State	The current state of the message.
Description	<p>Description of the last action recorded for the message. Example: An EDI file was received from the TP.</p> <p>The Description field may also display acknowledgment codes when the messaging event is for an acknowledgment. See Acknowledgments Display in the Audit Log on page 90.</p>
Transmission ID	<p>If scheduled transmissions have been enabled, this transaction will be bundled with a group of transactions prior to being transmitted.</p> <p>This column displays a unique value that identifies the group of transactions which will contain the transaction. A unique value that identifies the batch if the trading partner scheduler is enabled or the Gateway scheduler is enabled when Gateway is used.</p>
Transmission Time	<p>If scheduled transmissions have been enabled, this column will display the time this transaction was queued up to await transmission.</p> <p>When the scheduled time arrives, this column displays the actual transmission time. However, nothing displays if the scheduler is not enabled in the Scheduled Transmission tab.</p>

Transaction Detail View for the Gateway Protocol

The detail view of each scheduled transmission summary row displays transactions that are added or removed from the batch or scheduled transmission. It also shows the protocol, document ID, operation ID, and trading partner of the scheduled transaction and the scheduled transmission time.

Table 15 Transaction Detail Columns

Column	Definition
Time Stamp	Time when this transaction was processed.
Trading Partner	The destination trading partner the message will be sent to eventually. This could be different from the Transporting Trading Partner if it is used for batching.
Operation ID	Example: 00200/004020/101. For entries that do not reflect transactions, this field is EDI/Outbound.

Table 15 Transaction Detail Columns (Cont'd)

Column	Definition
Document ID	A unique value that identifies the transaction. For entries that do not reflect transactions, this field is the same as the Transmission ID field.
Transmission Time	This column displays the scheduled transmission time when the transaction is added to a batch or scheduled and is pending for transmission. When the scheduled time arrives, this column displays the actual transmission time.
Description	A description of what happened for each entry. Example: Transaction added to batch for scheduled transmission successfully.
Status	Status of the message, such as COMPLETED, PENDING, ERROR, COMPLETED WITH ERRORS, and so on.
State	The current state of the message.

Acknowledgment Messages

For information about how to see the error information contained in acknowledgments, see [State Details View on page 89](#).

EDIFACT

For CONTRL messages, the status of the last event for the CONTRL message will be COMPLETED, if the message contains only the following acknowledgment codes:


- 1 acknowledged
- 5 UNB/UNZ accepted
- 7 this level acknowledged, next lower level acknowledged if not explicitly rejected
- 8 interchange received

A code of 2 contained in the CONTRL message will be indicated by a status of ACK ACCEPTED WITH ERRORS. A code of 3, 4, or 6 in the CONTRL message will be indicated by a status of ACK REJECTED ERRORS. In this way it is possible to use the audit log to see which CONTRL messages report EDIFACT messages not accepted by your trading partner.

X12

For 997, 999 and TA1 acknowledgments, the status of the last event for each transaction will be COMPLETED, if the transactions contain an acknowledgment code of A for accepted. A status of ACK ACCEPTED WITH ERRORS will be indicated if the acknowledgment contains a code of E. A status of ACK REJECTED ERROR will be indicated if the acknowledgment contains any other code. In this way you can use the audit log to see which 997 or TA1 acknowledgments report X12 transactions or interchanges which were not accepted by your trading partner.

State Details View

To view the details of a specific state, click the document icon  at the left of a transaction detail entry. The Log Viewer displays in one window all of the information from the summary view and transaction details view for this current state.

From the state details view you can save the message associated with the processing state entry to a file. To information on this feature, see the Viewing Logs and Reports chapter in *TIBCO BusinessConnect Trading Partner Administration* You can also save the HTML format validation audit error report to a file in case of validation errors. You have to configure to enable this feature.

State Detail View for the Gateway Protocol

Table 16 State Detail Columns

Column	Definition
Time Stamp	Time when this message was processed
Trading Partner	Trading partner installation name.
Operation ID	Example: 00200/004020/101. For entries that do not reflect transactions, this field is EDI/Outbound.
Document ID	A unique value identifying this transaction.
Transmission Time	This column displays the scheduled transmission time when the transaction is added to a batch or scheduled and is pending for transmission. When the scheduled time arrives, this column displays the actual transmission time.
Description	Description of the last action recorded for the message. Example: Transaction added to batch for scheduled transmission successfully.
Status	Status of the message, such as COMPLETED, PENDING, ERROR, COMPLETED WITH ERRORS, and so on.
State	The current state of the message. Examples: TXN_ADDED_TO_BATCH, TXN_PACKAGED_FOR_BATCH.

Acknowledgments Display in the Audit Log

The following is an example of the audit log entry for previously transmitted transactions when an acknowledgment with acceptance status is received from your trading partner.

- **Status** ACK ACCEPTED
- **State** ACK_FROM_TP
- **Description** Acknowledgment received with the Ack code or Code name.

EDIFACT Acknowledgment Codes

The CONTRL Syntax and Service Report Message acknowledgment is available in EDIFACT. The CONTRL acknowledgment can have the action codes and code names listed in [Table 17](#).

Table 17 EDIFACT CONTRL Acknowledgment Action Codes

Ack Action Code	Code Name
1	Acknowledged (this level and all lower levels).
2	Acknowledged - errors detected and reported.
3	One or more rejected - next lower level.
4	This level and all lower levels rejected.
5	UNB/UNZ accepted.
6	UNB/UNZ rejected.
7	This level acknowledged, next lower level acknowledged if not explicitly rejected.
8	Interchange received.

If the code of a CONTRL acknowledgment is 1 or 7, the status is COMPLETED. A code of 2 contained in the CONTRL message will be indicated by a status of ACK ACCEPTED WITH ERRORS. A code of 3, 4, or 6 in the CONTRL message will be indicated by a status of ACK REJECTED ERRORS.

X12 Acknowledgment Codes

The following acknowledgments are available in X12:

- [997 and 999 X12 Functional Acknowledgment](#)
- [TA1 X12 Interchange Acknowledgment](#)

997 and 999 X12 Functional Acknowledgment

The 997 and 999 acknowledgments have codes and code names listed in [Table 18](#).

Table 18 997 X12 Functional Acknowledgment

Ack Code	Code Name
A	Accepted.

Table 18 997 X12 Functional Acknowledgment (Cont'd)

Ack Code	Code Name
E	Accepted but errors were noted.
M	Rejected, message authentication code (MAC) failed.
R	Rejected.
W	Rejected, assurance failed validity tests.
X	Rejected, content after decryption could not be analyzed.

If 997 and TA1 have code A, the status is COMPLETED. A status of ACK ACCEPTED WITH ERRORS will be indicated if the acknowledgment contains a code of E. The status of ACK REJECTED ERROR will be indicated if the acknowledgment contains any other code.

TA1 X12 Interchange Acknowledgment

The TA1 acknowledgment can have the codes and code names listed in Table 19.

Table 19 TA1 X12 Interchange Acknowledgment

Ack Code	Code Name
A	The transmitted interchange control structure header and trailer are received and have no errors.
E	The transmitted interchange control structure header and trailer are received and are accepted but errors are noted. This means the sender must not resend this data.
R	The transmitted interchange control structure header and trailer are rejected because of errors.

If 997 or 999 and TA1 have code A, the status is COMPLETED. For all other codes, the status is COMPLETED WITH ERRORS. These values come from the status field in certain tables.

User Keys

Users can choose certain elements in their guidelines to be shown in an audit log, and these elements are searchable in the audit log summary rows.

For example, in EDIFACT and X12 protocol you would search for purchase orders or order numbers instead of actually looking into the EDI document.

To achieve the user key functionality, perform the following steps:

- 1. Open a transaction, message guideline, or both in EDISIM Standards Editor.

2. Select the element you want to show in the TIBCO BusinessConnect audit log.
Select the DSR Mark/Unmark to mark the element. You would see a tick mark to indicate it is marked. This will define a user key for Reference Identification
3. Click **Apply**.
4. Click **Save** to save the guideline.
5. Load the guideline in the operation editor or in the operation binding for that specific business agreement.

When a transaction is received for the transaction type for which you requested the user key value to be shown, the BusinessConnect engine picks up the value for the field indicated in the guideline to be displayed in the User Key column, as well as the part of the userkey attribute of the ResponderRequest message.

You can add as many keys as you prefer. The key values will be shown in one column only and will be separated by a semicolon (;).



The user key feature is supported on inbound and outbound for the X12, TEXT, TRADACOMS, and EDIFACT protocols.

Retrieve the User Key

The user key will be retrieved after the documents have been converted from XML to EDI.

They can be retrieved only if outbound validation has been enabled for outbound documents, and inbound validation has been enabled for inbound documents.

Non-Repudiation Log

The non-repudiation logs are used to provide proof of the message delivery and depend on authentication and use of digital signatures. Digitally signed incoming messages are authenticated and stored in the non-repudiation database. Digitally signed outbound messages are also stored in the database. See also *TIBCO BusinessConnect Trading Partner Administration Guide*, Non-repudiation logs.

Summary View

Not all of the information in the non-repudiation log can be viewed; for example, digital signatures are stored in the log but cannot be displayed. The summary view of the non-repudiation log displays the information in [Table 20](#).

Table 20 Non-Repudiation Log Columns

Column	Description
Time Stamp	Time when this message was processed.
Operation ID	A protocol-specific value identifying the message.
Document ID	A unique value identifying this document.
Trading Partner	The trading partner name.
Interchange Qlfr	The interchange qualifier of the trading partner.
Interchange ID	The interchange ID of the trading partner.
User Key	Displays the value of certain fields in a transaction when the guideline includes a user key for the field; for example, the 997 and TAI guidelines are updated to include user keys which will cause the ack status of a 997 or TAI transaction to be displayed.
Interchange Control Number	The interchange control number.
Group Control Number	The group control number.
Txn Control Number	The transaction control number.
Initiated by Host	True for outbound documents. False for inbound documents

Message Queue Log

The message queue log is used to store those transactions not yet sent to your trading partners as they are awaiting their scheduled transmission time. These transactions may not have been transmitted for one of the following reasons:

- The transmission window for a trading partner is not open.
- A transmission frequency has been set for the trading partner and the next transmission time has not yet been reached.
- The transactions are accumulated until a certain number is reached.



Detail views are not provided for the message queue log.

Summary View

Table 21 displays the summary view of the message queue log.

Table 21 Message Queue Log Columns

Column	Description
Start Time	The start time of the transmission window.
End Time	The end time of the transmission window.
Time Stamp	Time when this message was entered into the queue.
Protocol	X12, EDIFACT, TEXT
Operation ID	A protocol-specific value identifying the message.
Trading Partner	The trading partner name.
Interchange Qlfr	The interchange qualifier of the trading partner.
Interchange ID	The interchange ID of the trading partner.
Interchange Control Number	batch
Group Control Number	batch
Txn Control Number	batch
Initiated by Host	True

Log Viewer Options

From the message queue log summary view, you can select transactions to send before their scheduled transmission time or you can cancel the transmission of transactions.

To send a transaction before its scheduled transmission time, select the check box in the left-most column of the summary view and click the **Send** button.

To cancel a transaction, select the check box in the left-most column of the summary view and click the **Cancel** button.

Transactions in the Message Queue Viewed in the Audit Log

Transactions stored in the message queue log have the following status and state when viewed in the audit log:

- Status: SCHEDULED
- State: TXN_ADDED_TO_BATCH

You can delete messages from the message queue log, cancelling their scheduled transmission. These cancelled transactions have the following status and state when viewed in the audit log:

- Status: CANCELED
- State: TXN_REMOVED_FROM_BATCH

You can send scheduled messages from the message queue log. TIBCO BusinessConnect removes them from the scheduled transmission and sends them instantly.

Resend Log

The resend log allows you to search for and view both resent transactions and resendable transactions. It also allows you to easily initiate a resend.

TIBCO BusinessConnect allows you to resend transactions based on their state. The resendable states vary by business protocol. [Table 22](#) provides detailed information.

Table 22 Resendable States

Transaction Type, State, and Description	Business Protocols
Business level Acks sent to partner: ACK_TO_TP This state is resendable if the partner requested a Business level Acknowledgment, for example, TA1, 997 for X12	EDIFACT X12
Message from partner: EDI_LOGGED_TO_FILE When the Log Raw EDI Request to File feature is enabled for inbound, you can resend messages previously sent from your partner to yourself as if it was resent by the partner. TIBCO BusinessConnect reprocesses the resent message. The Log Raw EDI segment should be enabled on the trading host for inbound.	EDIFACT, TEXT, TRADACOMS, X12
Message from partner: EDI_TXN_FROM_TP Same as EDI_LOGGED_TO_FILE except that you can send individual transactions instead of the whole interchange. To use this resendable state, you must enable Log Raw EDI Segments to File and Include Envelope Segments at the Transaction level, and provide a value for Store Location value on the trading partner side for the inbound.	EDIFACT X12
New request from back-end private process: RECEIVED_FROM_PP TIBCO BusinessConnect validates and converts payloads as appropriate and uses the primary transport to resend messages to the partner.	EDIFACT, TEXT, TRADACOMS, X12
InitiatorResponse to private process: RESPONSE_TO_PP TIBCO BusinessConnect can resend these private process messages to TIBCO Rendezvous or JMS. See <i>TIBCO BusinessConnect User's Guide</i> for information about how TIBCO Rendezvous and JMS messages are resent.	EDIFACT, TEXT, TRADACOMS, X12, Gateway

Table 22 Resendable States (Cont'd)

Transaction Type, State, and Description	Business Protocols
ResponderRequest to private process: REQUEST_TO_PP TIBCO BusinessConnect can resend these private process messages to TIBCO Rendezvous or JMS. See <i>TIBCO BusinessConnect User's Guide</i> for information about how TIBCO Rendezvous and JMS messages are resent.	EDIFACT, TEXT, TRADACOMS, X12
Batched or scheduled messages to partner: REQUEST_FOR_BATCH TIBCO BusinessConnect can resend messages to a partner that were sent by batch or scheduled.	GATEWAY

Click to resend a message that was not successfully sent the first time. You should *use caution* when using this feature. If the resend request is activated before the current transaction completes or returns an error, the outcome is beyond control of the protocol.

The **Resend** button is always available, except when a transaction is:

- A resent transaction itself. That is, not the original request message, but a message that was resent.
- A transaction that is currently queued for resending, but is not yet processed.

You can resend these messages:

- Initiator request
- an Initiator receiving an Initiator response
- a Responder receiving a message from the trading partner
- Responder request
- batch

Resending a message is identical to making a new message request, and can result in multiple transactions of the same request.

For outbound documents, selecting Resend retransmits this document to the trading partner. For inbound documents, selecting Resend will resend the message to the private process.

For more information on the Resend Log, see the chapter on Viewing Logs and Reports in *TIBCO BusinessConnect Trading Partner Administration Guide*.

Manual Resend Scenarios

The following scenarios are supported for outbound requests:

- Resend to reprocess the XML request from the private process. A new EDI document is regenerated for transmission to your trading partner.

This resend is available for the audit log entry with state RECEIVED_FROM_PP.

- Resend to republish the response to the private process. The previous response from the trading partner is published to the private process again.

This resend is available for the audit log entry with state RESPONSE_TO_PP.

The following scenarios are supported for inbound requests:

- Resend to republish the converted XML request for an incoming transaction to the private process. The previous converted XML request from the trading partner is published to the private process again. The original incoming EDI document will not be resubmitted for message validation and XML conversion.

This resend is available for the audit log entry with state REQUEST_TO_PP.

- Resend to reprocess the incoming EDI document received from your trading partner. The incoming EDI document will be resubmitted for message validation and XML validation again.

This resend option is available for the audit log entry with state EDI_LOGGED_TO_FILE when the option to Log Raw EDI Request is enabled for the trading partner.

Document Archiving

TIBCO BusinessConnect EDI Protocol powered by Instream provides the ability to archive raw requests. Document archiving is controlled through settings in the host participant **Logging** tab. Inbound and outbound raw EDI requests are permanently archived for the following reasons:

- To enable the ability to resend inbound files for re-processing as needed before forwarding to the local private process. If there were problems in processing an inbound file the first time, the file can be re-sent from the local storage site for re-processing before forwarding to the local private process.
- To provide a central location for archiving of inbound files and outbound raw EDI documents on the local file system.

If the **Log Raw EDI Request to File** check box is selected, the files retrieved from or sent to a trading partner are stored in a directory whose name is derived from the Store Location field as follows:

- A subdirectory named *In* or *Out* is appended to the store location. Inbound EDI documents are sorted into subdirectories under the *In* directory. Outbound EDI documents are sorted into subdirectories under the *Out* directory.
- The subdirectories are named according to the trading partner name:
StoreLocation\In\tpName and *StoreLocation\Out\tpName*

Selecting the **Include Date Folder** check box adds the date directory to the document path. If unchecked, the date directory is omitted and the document is stored under the *tpName* directory.

The new file name of the inbound or outbound file has the following syntax:

tpName-year-month-day--hour-h-minute-m-seconds-s-milliseconds-ms.edi

where timestamp has the format:

year-month-day--hour-h-minute-m-seconds-s-milliseconds-ms.edi

Example:

/tibco/bc/FILE/In/buy123/2002-02-14/buy123-2009-02-14--23-h-38-m-38-s-940-ms.edi

Chapter 8 **Private Messages**

This chapter describes private message formats in TIBCO BusinessConnect EDI Protocol powered by Instream transactions.

Topics

- [Overview, page 102](#)
- [Initiator Messages, page 103](#)
- [Responder Messages, page 115](#)
- [General Messages, page 127](#)
- [Objects Included in Private Messages, page 136](#)

Overview

The following sections describe the messages used for private request and response document exchange in TIBCO BusinessConnect EDI Protocol powered by Instream.

You can use either TIBCO Rendezvous or JMS as the transport method for private process messages. The message-packet format varies with these two transport methods.

With TIBCO Rendezvous, TIBCO BusinessConnect uses a subject name constructed as follows:

prefix.installation.standardID.message_category.message_type

For example:

AX.BC.BC-ACME.X12.INITIATOR.REQUEST

With JMS, TIBCO BusinessConnect uses a JMS queue for Initiator/Responder messages and a JMS topic for general messages. The topic name or queue name is constructed as follows:

prefix.installation.message_category.message_type

For example:

AX.BC.BC-ACME.INITIATOR.REQUEST

Topic and queue names do not include the standardID.

Initiator Messages

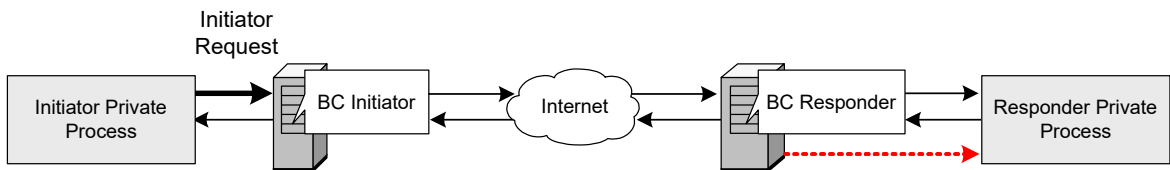
Initiator Outbound Request

The Initiator private process uses this message to handle outbound requests.



If the TpName and receiver qualifier ID are sent from the private process, TpName takes precedence over the receiver qualifier ID of the per message field. The TpName is selected first; if it is not present, the receiver qualifier ID is selected.

Figure 15 Initiator Outbound Request



Rendezvous Subject Name *prefix.installation.standardID.INITIATOR.REQUEST*
 Example: AX.BC.BC-ACME.X12.INITIATOR.REQUEST

JMS Queue Name *prefix.installation.INITIATOR.REQUEST*
 Example: AX.BC.BC-ACME.INITIATOR.REQUEST

Message Name InitiatorRequest

Table 23 Initiator Request (Sheet 1 of 6)

Field	Type	Required	Description
standardID	String	Yes	X12, EDIFACT, TEXT, TRADACOMS, Gateway.
operationID	String	Yes	ID of operation to be performed. Format: <i>interchange_version/group_version/transactionoperation</i> Example: 00403/005010/850
transactionID	String	Yes	An ID unique within Initiator private process's environment for this transaction. The private process creates this ID.
tpName	String	Yes	Trading partner name.

Table 23 Initiator Request (Sheet 2 of 6)

Field	Type	Required	Description
encoding	String	No	<p>The encoding used in the request.</p> <p>Example: UTF--16. When sending UTF-16 encoded request of EDI documents, this field should have a value of UTF-16.</p>
perMessage	Object	No	<p>Overrides the default parameters of the host and trading partner configuration on a per-message basis. For overriding parameters for the interchange header, the ID and qualifier are validated against the list of domain identities for the host and the trading partner. See EnvelopeInfo, page 136.</p> <p>X12 The parameters for X12 are:</p> <p>Interchange Header: Sender/Receiver ID and Qualifier (ISA05, ISA06, ISA07, ISA08)</p> <p>Group Header: Group Application Sender/Receiver ID (GS02, GS03)</p> <p>InterchangeHeaderInfo present in the class EDIInterchangeInfo has several overridable fields that are part of the participant Interchange Header tab. The following fields are overridable:</p> <p>ISA01Qualifier: Authorization Information Qualifier</p> <p>ISA02Information: Authorization Information</p> <p>ISA03Qualifier: Security Information Qualifier</p> <p>ISA04Information: Security Information</p> <p>ISA09Date: Interchange Date</p> <p>ISA10Time: Interchange Time</p> <p>ISA14AckRequested: Acknowledgment Requested</p> <p>ISA15UsageIndicator: Usage Indicator</p> <p>ISA16CompElemSep: Component Element Separator</p> <p>InterchangeHeaderInfo present in the class EDIInterchangeInfo has several overridable fields that are part of the participant Interchange Header tab.</p>

Table 23 Initiator Request (Sheet 3 of 6)

Field	Type	Required	Description
perMessage (cont)			<p>EDIFACT The parameters for EDIFACT are:</p> <p>Interchange Header: Sender/Recipient ID and Qualifier (UNB-S002/0004, UNB-S002/0007, UNB-S003/0010, UNB-S003/0007)</p> <p>Group Header: Group Application Sender/Recipient ID and Qualifier (UNG-S006/0007, UNG-S006/0040, UNG-S007/0007, UNG-S007/0044)</p> <p>InterchangeHeaderInfo present in the class EDIInterchangeInfo has several overridable fields that are part of the participant Interchange Header tab. The following fields are overridable:</p> <p>UNBS001Identifier: Syntax identifier</p> <p>UNBS001ServiceCodeList: Service code list directory version number</p> <p>UNBS001CharacterEncoding: Character encoding, coded</p> <p>UNBS002InternalID: Interchange sender internal identification. Identification (such as a division, branch, or computer system/process) specified by the sender of the interchange; to be included, if agreed, by the recipient in the response interchange to facilitate internal routing.</p> <p>UNBS002SubInternalID: Interchange sender internal sub-identification. Sub-level of the server internal identification when further sub-level identification is required.</p> <p>UNBS003InternalID: Interchange recipient internal identification/Routing Address</p> <p>UNBS003SubInternalID: Interchange recipient internal sub-identification</p> <p>UNBS004Date: Date</p> <p>UNBS004Time: Time</p> <p>UNBS005Reference: Recipient reference/password</p> <p>UNBS005Qualifier: Recipient reference/password qualifier</p> <p>UNB0026ApplicationRef: Application reference</p> <p>UNB0029ProcessingCode: Processing priority code</p> <p>UNB0031AckRequested: Acknowledgment request</p> <p>UNB0032AgreementIdentifier: Interchange agreement identifier</p> <p>UNB0035TestIndicator: Test Indicator</p> <p>Note: The values entered in the InterchangeHeaderInfo Node of the perMessage field in TIBCO BusinessConnect Palette are not validated when XML to EDI conversion happens for the X12 and EDIFACT protocol. Users should send appropriate valid values from the private process.</p>

Table 23 Initiator Request (Sheet 4 of 6)

Field	Type	Required	Description
closure	Any	Yes	<p>InterchangeHeaderInfo present in the class EDIInterchangeInfo has several overridable fields that are part of the participant Interchange Header tab. The following fields are overridable:</p> <p>UNBS001Identifier: Syntax identifier</p> <p>UNBS001ServiceCodeList: Service code list directory version number</p> <p>UNBS001CharacterEncoding: Character encoding, coded</p> <p>UNBS002InternalID: Interchange sender internal identification. Identification (such as a division, branch, or computer system/process) specified by the sender of the interchange; to be included, if agreed, by the recipient in the response interchange to facilitate internal routing.</p> <p>UNBS002SubInternalID: Interchange sender internal sub-identification. Sub-level of the server internal identification when further sub-level identification is required.</p> <p>UNBS003InternalID: Interchange recipient internal identification/Routing Address</p> <p>UNBS003SubInternalID: Interchange recipient internal sub-identification</p> <p>UNBS004Date: Date</p> <p>UNBS004Time: Time</p> <p>UNBS005Reference: Recipient reference/password</p> <p>UNBS005Qualifier: Recipient reference/password qualifier</p> <p>UNB0026ApplicationRef: Application reference</p> <p>UNB0029ProcessingCode: Processing priority code</p> <p>UNB0031AckRequested: Acknowledgment request</p> <p>UNB0032AgreementIdentifier: Interchange agreement identifier</p> <p>UNB0035TestIndicator: Test Indicator</p> <p>Note: The values entered in the InterchangeHeaderInfo Node of the perMessage field in BusinessConnect Palette are not validated when XML to EDI conversion happens for X12 and EDIFACT protocols. Users should send appropriate valid values from the private process.</p>
txnGroupingID	String	No	<p>This field is used only for batching transactions.</p> <p>Transactions sent from the private process with the same txnGroupingID will be grouped in a single batch and not in different batches.</p> <p>Since transactions batched based on this parameter are not processed by the scheduled task poller, they can be triggered only from the Message Queue using the GUI, or by using the Transaction Trigger service supplied in TIBCO BusinessConnect EDI Protocol powered by Instream.</p>

Table 23 Initiator Request (Sheet 5 of 6)

Field	Type	Required	Description
request	String	No	An string representing the message body or TIBCO Rendezvous/JMS representation of an XML file. The XML must conform to the .xsd generated by EDISIM for the transaction type.
inputFile	String	No	Optional. Specifies the full path of the file where the XML request is stored. If this is specified, the request XML is read from the file. The request field takes precedence.
destinationFileName	String	No	<p>All protocols except Service.</p> <p>The final destination file name is comprised of destinationFileName and a file extension. The file extension default value is .dat and is derived from the property edi.outbound.contentDisposition.fileExtension, which is located under System Settings > Activated Protocol Plug-ins and Properties > tibEDI.</p> <p>This property is not supported for the plain HTTP/HTTPS transport.</p>
filePathForScript	String	No	Allows to set the destination path, which is available as script variable for FTP/FTPS/SFTP/FILE transports.
mimeSubject	String	No	<p>Optional.</p> <p>Custom subject line to be constructed on each outbound request when a MIME encoded transport is used. A custom mask can also be used in this attribute for variable replacement at runtime.</p> <p>See also Setting the Transport Subject Dynamically on page 109 for more details.</p>
ediGateway	String	No	Overrides the default setting of the transporting Gateway to use for each outbound request.
controlNumInfo	Object	No	<p>Specifies custom control numbers for the interchange, group and transaction on each outbound request. If this is specified, the conversion engine will not generate the control numbers based on the seed values configured for the partner profile.</p> <p>Note: When user-defined control numbers are specified, the conversion engine will not ensure the uniqueness and does not validate the control numbers against the control number sequences that are maintained internally. However, transactions constructed with user-defined control numbers will be recorded for acknowledgment reconciliation. See ControlNumberInfo, page 138.</p>
httpAttributes			

Table 23 Initiator Request (Sheet 6 of 6)

Field	Type	Required	Description
name	String	No	Name of the httpAttribute. Note : It is supported by X12, EDIFACT, TEXT, and TRADCOMS protocols.
value	String	No	Value assigned to the httpAttribute. Note : It is supported by X12, EDIFACT, TEXT, and TRADCOMS protocols.
attachment	Sequence of Attachment	No	Supports Email and AS1 transports with attachments when sending and receiving messages. This feature is supported with the EDIFACT, TRADACOMS and X12 protocols. When using Email and AS1 transports to send attachments, batching is not supported. Note: Attachments are ignored if both the content and fileName elements are empty, if both elements are configured, the content element takes precedence. See Attachment, page 140 .
enableCAQHPackage	Boolean	No	This field specifies if it is a CAQH or non-CAQH package. You set the value to 1 to specify if it is a CAQH package or set the value to 0 to specify that it is not a CAQH package.
CAQHOperationID	String	No	This string field describes the pre-defined CAQH operation ID.
PayloadType	String	Yes	This field describes the type of document (request, response, confirmation, or acknowledgement) and the type of transaction. You must define this field in the private process message for CORE-compliant packaging.
PayloadID	String	No	This field is the ID for the payload of the message.
SenderID	String	No	This field is the ID of the sender of the message.
ReceiverID	String	No	This field is the ID of the receiver of the message.
authenticationToken	String	Yes	This field combines the user name and password fields.
userName	String	Yes	This field is the user name that is entered to validate at the responder side.
password	String	Yes	This string field is the password that is entered to validate at the responder side.

Setting the Transport Subject Dynamically

Based on the subject specified in the configuration of the transports EMAIL, AS1, or AS2-HTTP/S, the subject field is overridable by the private process `mimeSubject`. Users set macros that are case-insensitive either at the configuration or at private process `mimeSubject` and the macro field is substituted by the TIBCO BusinessConnect EDI Protocol powered by Instream runtime engine. The values pertain to the interchange level for X12 and EDIFACT protocols and non-interchange field values for Gateway protocol.

The TEXT protocol does not have `InterchangeCtrlNum` or `InterchangeVerNum` and, therefore, the transport subject field with macros

```
#(InterchangeCtrlNum)
and
#(InterchangeVerNum)
```

is not applicable for this protocol.

The interchange level fields supported as macros are:

- ReceiverID, ReceiverQual, SenderID, SenderQual, InterchangeCtrlNum, and InterchangeVerNum.

Other non interchange fields that are supported are:

- toName, DocID, hostname, guid, and TxName.

All date/time fields supported for FILE.FTP macros are also supported. See [Chapter 6, File Masks, on page 73](#).



`#(tpName)` will always be replaced with the Transporting Trading Partner name whether the outbound message goes via Gateway or not.



If an EDI document is sent via the Gateway protocol from the private process as a pass-through, only the following macros are valid:

`TpName`, `HostName`, `Date/Time` macro fields, `GUID` and `docID`.

Values specific to the EDI Interchange document level values, such as `InterchangeID/Qualifier` (`SenderID/ReceiverID`, `SenderQual/ ReceiverQual`), `InterchangeVerNum` and `InterchangeCtrlNum` will not get populated in this scenario.

The subject in transport configuration can be overridden with the `mimesubject` from the private process:

- **Non-batching only** (non-Gateway trading partner or Gateway trading partner)
 - When the private process `mimeSubject` field is set with a value that contains some macro fields to TIBCO BusinessConnect, this value should override the value set

in the Transport subject and should be dynamically updated if there are macros passed, such as in #(DocID)- #(tpName)- #(SenderID).

If there are no macros, then the value should be used.

- When mimeSubject is not set, the Transport subject value should be dynamically updated if there are macros passed, such as in #(DocID)- #(tpName)- #(SenderID).

If there are no macros, then the value should be used.

- **Batching** (Gateway or protocol trading partner)

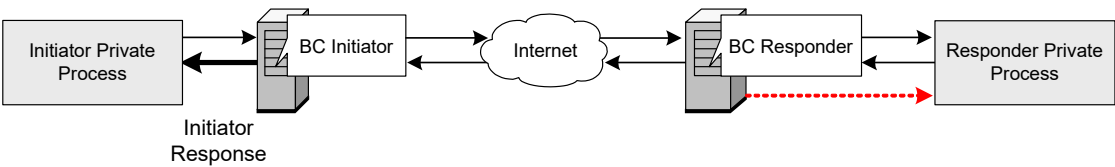
Overriding of the mimeSubject field is not supported for the batching.

The Transport subject value is used and the macro values of the first interchange in the document are set, if present.

Initiator Inbound Response

This message indicates if a InitiatorRequest has been successfully sent to your trading partner. If the requesting operation is a synchronous operation, this message contains the synchronous response EDI message received from the trading partner

Figure 16 Initiator Inbound Response



Rendezvous Subject Name *prefix.installation.standardID.INITIATOR.RESPONSE*
Example: AX.BC.BC-ACME.X12.INITIATOR.RESPONSE

JMS Queue Name *prefix.installation.INITIATOR.RESPONSE*
Example: AX.BC.BC-ACME.INITIATOR.RESPONSE

Message Name InitiatorResponse

Table 24 InitiatorResponse (Sheet 1 of 4)

Field	Type	Required	Description
standardID	String	Yes	X12, TEXT, TRADACOMS, EDIFACT, and Gateway.

Table 24 InitiatorResponse (Sheet 2 of 4)

Field	Type	Required	Description
transmissionID	String	No	The transmissionID property is provided in the InitiatorResponse message to indicate to the private process when it is scheduled for batching.
operationID	String	Yes	ID of operation to be performed. Format: <i>interchange_version/group_version/transactionoperation</i> Example: 00403/005010/850
transactionID	String	Yes	The response's transaction ID.
txName	String	Yes	Name of the transaction.
statusCode	Integer	Yes	200 success. Otherwise, one of the error codes. See Appendix B, Status and Error Codes , on page 147.
statusMsg	String	Yes	OK, or the string representing the cause of the error. In this case, the response might be an error document. See Appendix B, Status and Error Codes , on page 147.
envelopeInfo	Object	No	The interchange and group envelope information of this transaction. The interchange envelope identifies the sender and receiver interchange qualifier and ID. The envelope identifies the sender and receiver application ID. See EnvelopeInfo , page 136.
controlNumInfo	Object	No	The control number information of this transaction that identifies itself in group envelope and interchange envelope of the incoming document. It contains the control numbers for the interchange envelope, the group envelope and the transaction itself. See ControlNumberInfo , page 138. This field will have a value of batch in case of Batching. In case of messages sent from TEXT or Gateway, this field will not be populated with any values.
transactionInfo	Object	No	The raw EDI segments of this transaction and the header and trailer segment of its interchange envelope and group envelope. By default, transaction information is not included in the Initiator response. To include transaction information in the validation alert, select the Include Raw EDI Segments in Validation Error Advisory check box. If the Log Raw EDI segments To File check box for inbound transactions is enabled for the trading partner, the full path of the stored location is included in this message class. See TransactionInfo , page 138.
duplicate	Boolean	No	Set to true if the private process sent a duplicate document.

Table 24 InitiatorResponse (Sheet 3 of 4)

Field	Type	Required	Description
closure	Any	Yes	<p>The private process generates the closure message and sends it to TIBCO BusinessConnect.</p> <p>TIBCO BusinessConnect is required to return this closure contents back in the InitiatorResponse to ensure that the private process can match it with the original InitiatorRequest.</p>
response	Any	Yes	<p>XML string representing the message body or TIBCO Rendezvous/JMS representation of an XML file.</p> <p>For a notify operation, the XML response indicates if the outbound request was successfully sent to the trading partner.</p> <p>For a synchronous request-response operation, this XML string is the converted EDI information for either the rejection acknowledgment or the response that was received from the trading partner.</p>
archivedFileRef	String	No	<p>This field is populated with the location of the file where the raw EDI document is stored. It is populated only when the option Log raw EDI is enabled under Trading Host>Protocol with a valid location.</p>
auditReportFileRef	String	No	<p>This field is populated with the location of the XML file created during XML to EDI validation, when the option Enable Audit Report Logging is enabled for a participant available in the General tab for Outbound.</p> <p>This file is created only by the internal conversion engine and gives full reporting information of the transactions that are validated. A valid location must be entered for the trading host for the corresponding protocol. The protocol requires the parameter AuditReportLogging to be available to the TIBCO BusinessConnect Palette, which is available under the Advanced tab.</p> <p>This field is not populated for the Gateway or Service protocols, nor during the actual Batch Collation process that does not involve any internal conversion engine processing.</p>
CAQHOperationID	String	No	<p>This field describes the pre-defined CAQH operation ID.</p>
PayloadType	String	No	<p>This field describes the type of document (request, response, confirmation, or acknowledgement) and the type of transaction. For Initiator Response messages, the payload type is received from the trading partner as part of an actual request.</p>
ProcessingMode	String	No	<p>This field describes the type of transaction - real-time or batch.</p>

Table 24 InitiatorResponse (Sheet 4 of 4)

Field	Type	Required	Description
PayloadID	String	No	This field is the ID for the payload of the message received from a trading partner.
TimeStamp	String	No	This field indicates the time at which the response message is sent.
SenderID	String	No	This field is the ID of the sender of the message.
ReceiverID	String	No	This field is the ID of the receiver of the message.
ErrorCode	String	No	This field is the error code that is displayed.
ErrorMessages	String	No	This string field is the error message that is displayed against the corresponding error code.

Initiator Acknowledgment From Partner

The Initiator TIBCO BusinessConnect uses this message to create a notify message that it sends back to its private process to indicate that a previously sent transaction has been successfully reconciled upon receiving an acknowledgment from the trading partner. If the Responder TIBCO BusinessConnect failed to successfully post the acknowledgment to the Initiator, the `statusMsg` field indicates the cause of the error.

When configuring the smart routing rule, you should set operation "EDI/Inbound/Interchange" under BusinessConnect > System Settings > Private Process Smart Routing for initiator.ack to get the message.

Rendezvous Subject Name *prefix.installation.standardID.INITIATOR.ACK*
 Example: AX.BC.BC-ACME.X12.INITIATOR.ACK

JMS Queue Name *prefix.installation.INITIATOR.ACK*
 Example: AX.BC.BC-ACME.INITIATOR.ACK

Message Name TransactionAckFromTradingPartner

Table 25 TransactionAckFromTradingPartner

Field	Type	Required	Description
operationID	String	Yes	ID of operation to be performed. Format: <i>interchange_version/group_version/operation</i> Example: 00403/005010/997

Table 25 TransactionAckFromTradingPartner

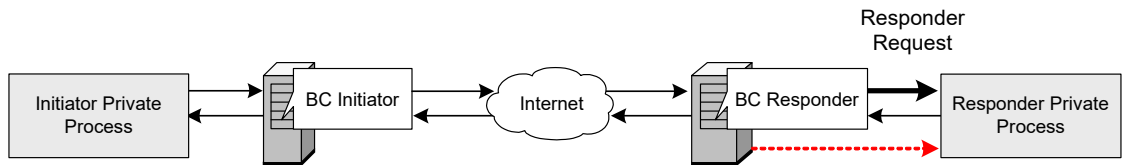
Field	Type	Required	Description
transactionID	String	Yes	The original transaction ID for the previously transmitted transaction.
tpName	String	Yes	Trading partner name.
status	String	Yes	Status of the transaction that is reconciled based on information of the acknowledgment received from the trading partner.
description	String	Yes	Details of the reconciled transaction based on information of the acknowledgment received from the trading partner.
receiver	Object	No	The interchange qualifier and ID of the trading partner who sent the acknowledgment that reconciled this transaction. See Identity , page 137.
controlNumInfo	Object	No	The control number information of the reconciled transaction. See ControlNumberInfo , page 138.

Responder Messages

Responder Inbound Request

The Responder private process uses this message to handle inbound requests. A ResponderRequest message is created for each transaction in each interchange from the received inbound EDI document. A ResponderInterchange message is created for each interchange from the received inbound EDI document.

Figure 17 Responder Inbound Request



Rendezvous Subject Name *prefix.installation.standardID.RESPONDER.REQUEST*
 Example: AX.BC.BC-ACME.X12.RESPONDER.REQUEST

JMS Queue Name *prefix.installation.RESPONDER.REQUEST*
 Example: AX.BC.BC-ACME.RESPONDER.REQUEST

Message Name ResponderRequest

Table 26 ResponderRequest (Sheet 1 of 5)

Field	Type	Required	Description
standardID	String	Yes	X12, TEXT, TRADACOMS, EDIFACT, and Gateway.
operationID	String	Yes	ID of operation to be performed. Format: <i>interchange_version/group_version/transactionoperation</i> Example: 00403/005010/850
operationType	String	Yes	Notify or synchronous request-response.
transactionID	String	Yes	Unique ID that the Responder TIBCO BusinessConnect generates to associate with an inbound message.
txName	String	Yes	Name of the transaction. Examples: 850, 270

Table 26 ResponderRequest (Sheet 2 of 5)

Field	Type	Required	Description
sourceTP	String	Yes	Name of the Initiator of this message.
destinationTP	String	Yes	Name of the Responder to this message.
tpExtraInfo	String	No	Extra information associated with the trading partner as configured to be included in this message request but not as part of the inbound document. Use this information to assist in routing this request to the backend private process. This is controlled by the Extra Info in Private Process Request field.
sourceGroup	String	Yes	Application sender's code for functional group.
destinationGroup	String	Yes	Application receiver's code for functional group.
controlNumInfo	Object	No	The control number information of this transaction that identifies itself in group envelope and interchange envelope of the incoming document. It contains the control numbers for the interchange envelope, the group envelope and the transaction itself. See ControlNumberInfo , page 138.
envelopeInfo	Object	No	The interchange and group envelope information of this transaction. The interchange envelope identifies the sender and receiver interchange qualifier and ID. The envelope identifies the sender and receiver application ID. See EnvelopeInfo , page 136.
transactionInfo	Object	No	The raw EDI segments of this transaction and the header and trailer segment of its interchange envelope and group envelope. By default, transaction information is not included in the Responder request. To include transaction information in the validation alert, select the Include in Validation Error Advisory check box. If the Log Raw EDI Segments to File check box for inbound transactions is enabled for the trading partner, the full path of the stored location is included in this message class. See TransactionInfo , page 138.

Table 26 ResponderRequest (Sheet 3 of 5)

Field	Type	Required	Description
duplicate	Boolean	No	<p>Indicates that this transaction of the interchange from the incoming document has previously been processed. The transaction is detected as a duplicate when the message digest created based on the message contents and the control number information matches a message digest that was created and persisted previously. By default, the duplicate detection for inbound transaction is disabled. When this check box is selected, all inbound transactions that are received from the trading partner are checked to see if a duplicate has been received and appropriately marked in the RESPONDER.REQUEST PP message. The duplicate attribute in ResponderRequest is marked as true if the same message is received from the trading partner. Following are the attributes that are used for comparison:</p> <p>X12 and EDIFACT have the following attributes in common: Trading Partner Name, operationID, transactionName, whole transaction (only the ST to SE segments for X12, UNH to UNT segments for EDIFACT), host initiates flag, interchange control number, group control number, and transaction control number.</p> <p>TEXT has the following attributes: Trading Partner Name, operationID, transactionName, whole transaction, host initiates flag.</p> <p>TRADACOMS has the following attributes: Trading Partner Name, operationID, transactionName, whole transaction, host initiates flag, interchange control number.</p>
resend	Boolean	No	Indicates if this request is generated based on re-processing of an inbound EDI document triggered by the resend link from the audit log viewer.
validated	Boolean	No	<p>Indicates whether or not this transaction of the interchange from the inbound document has been validated by the validation and conversion engine.</p> <p>Controlled by the Enable EDI Validation check box.</p>
closure	Any	No	<p>Used for the synchronous request-response operation.</p> <p>TIBCO BusinessConnect generates the closure message and sends it to the local private process. If this is a request of a synchronous request-response operation, the private process is required to return this closure contents back in the ResponderResponse message to ensure that TIBCO BusinessConnect can match it with the original Responder request message.</p>
rawRequest	Any	No	Used only for inbound Gateway request. The inbound raw EDI segments expressed as a byte array when this request is received from the trading partner through the Gateway.

Table 26 ResponderRequest (Sheet 4 of 5)

Field	Type	Required	Description
request	String	No	A string representing the message body or TIBCO Rendezvous/JMS representation of an XML file. This string is the converted EDI information for one transaction in the EDI document that was received from the trading partner. If the option to disable XML conversion for the sending trading partner is disabled, this field is not populated with the XML string.
requestFile	String	No	If the request message is passed by file reference, holds the name of the file. Used when Publish XML Request as File Reference is enabled and the size of the request exceeds the specified threshold value. For more information about publishing requests as file references, see Advanced Tab in the X12 Configuration.
receivedTime	Data Time	No	The original GMT timestamp of the inbound request received by TIBCO BusinessConnect.
infoWarnList	Object	No	Details validation informational and warning messages associated with this transaction. It is applicable to inbound X12, EDIFACT and HIPAA transaction when error severity configuration is defined for the transaction type. See ConversionEngineInfo , page 139.
sourceFile	String	No	This field contains the filename of the original request from trading partner; for example, for the FILE and FTP transports, this is the actual filename that is in the directory or on the FTP server.
CAQHOperationID	String	No	This field describes the pre-defined CAQH operation ID.
PayloadType	String	No	This field describes the type of document (request, response, confirmation, or acknowledgement) and the type of transaction. For Responder Request messages, the payload type is received from the trading partner as part of an actual request.
ProcessingMode	String	No	This field describes the type of transaction - real-time or batch.
PayloadID	String	No	This field is the ID for the payload of the message received from a trading partner.
TimeStamp	String	No	This field indicates the time at which the message is sent.
SenderID	String	No	This field is the ID of the sender of the message.
ReceiverID	String	No	This field is the ID of the receiver of the message.

Table 26 ResponderRequest (Sheet 5 of 5)

Field	Type	Required	Description
ErrorCode	String	No	This field is the error code that is displayed.
ErrorMessages	String	No	This string field is the error message that is displayed against the corresponding error code.

Responder Inbound Interchange

TIBCO BusinessConnect sends the `ResponderInterchange` message to the private process when it receives an interchange from the trading partner.



When the inbound document is TRADACOMS, the interchange is the entire message.

Rendezvous Subject Name

prefix.installation.standardID.RESPONDER.INTERCHANGE

Example: AX.BC.BC-ACME.X12.RESPONDER.INTERCHANGE

JMS Queue Name

prefix.installation.RESPONDER.INTERCHANGE

Example: AX.BC.BC-ACME.RESPONDER.INTERCHANGE

Message Name

`InboundInterchangeLoggedToFile`

Table 27 ResponderInterchange

Field	Type	Required	Description
standardID	String	Yes	X12, TEXT, TRADACOMS, and EDIFACT
operationID	String	Yes	Contains a constant value: EDI/Inbound/Interchange
transactionID	String	Yes	Unique ID that the Responder TIBCO BusinessConnect generates to associate with an inbound message.
sourceTP	String	Yes	Name of the Initiator of this message.
destinationTP	String	Yes	Name of the Responder to this message.
tpExtraInfo	String	No	Extra information associated with the trading partner as configured to be included in this message request but not as part of the inbound document. Use this information to assist in routing this request to the private process. Controlled by the Extra Info in Private Process Request field.

Table 27 ResponderInterchange (Cont'd)

Field	Type	Required	Description
controlNumInfo	Object	No	The control number information of this transaction that identifies itself in the group envelope and interchange envelope of the inbound document. It contains the control numbers for the interchange envelope. See ControlNumberInfo , page 138.
interchangeInfo	Object	No	The interchange envelope information of this interchange. The interchange envelope identifies the sender and receiver interchange qualifier and ID. The envelope identifies the sender and receiver application ID. See InterchangeInfo , page 137.
requestFile	String	No	If the request message is passed by a file reference, holds the name of the file. Controlled by Log Raw Request to File field.
receivedTime	DateTime	No	The original timestamp of the inbound request received by TIBCO BusinessConnect.
sourceFile	String	No	This field contains the filename of the original request from trading partner; for example, for the FILE and FTP transports this is the actual filename that is in the directory or on the FTP server.

ResponderInterchangeComplete

TIBCO BusinessConnect sends the message `ResponderInterchangeComplete` to the private process upon receiving an interchange from the trading partner when the option under General>Inbound>Enable Interchange Complete Message is enabled.

This message also marks the completion of this interchange validation and conversion.

Rendezvous Subject Name	<code>prefix.installation.standardID.RESPONDER.INTERCHANGE.COMPLETE</code> Example: AX.BC.BC-ACME.X12.RESPONDER.INTERCHANGE.COMPLETE
JMS Queue Name	<code>prefix.installation.RESPONDER.INTERCHANGE.COMPLETE</code> Example: AX.BC.BC-ACME.RESPONDER.INTERCHANGE.COMPLETE

Message Name InboundInterchangeComplete*Table 28 ResponderInterchangeComplete*

Field	Type	Required	Description
standardID	String	Yes	X12, EDIFACT, TRADACOMS
operationID	String	Yes	Contains a constant value: EDI/Inbound/Interchange
transactionID	String	Yes	Unique ID that the Responder TIBCO BusinessConnect generates to associate with an inbound message.
sourceTP	String	Yes	Name of the Initiator of this message.
destinationTP	String	Yes	Name of the Responder to this message.
tpExtraInfo	String	No	Extra information associated with the trading partner as configured to be included in this message request but not as part of the inbound document. Use this information to assist in routing this request to the private process. Controlled by the Extra Info in Private Process Request field.
controlNumInfo	Object	No	The control number information of this transaction that identifies itself in the group envelope and interchange envelope of the inbound document. It contains the control numbers for the interchange envelope. See ControlNumberInfo , page 138.
interchangeInfo	Object	No	The interchange envelope information of this interchange. The interchange envelope identifies the sender and receiver interchange qualifier and ID. The envelope identifies the sender and receiver application ID. See InterchangeInfo , page 137.
requestFile	String	No	If the request message is passed by a file reference, holds the name of the file. Controlled by Log Raw Request to File field.
receivedTime	DateTime	No	The original timestamp of the inbound request received by TIBCO BusinessConnect.
sourceFile	String	No	This field contains the filename of the original request from trading partner; for example, for the FILE and FTP transports this is the actual filename that is in the directory or on the FTP server.

Table 28 ResponderInterchangeComplete (Cont'd)

Field	Type	Required	Description
auditReportFileRef	String	No	<p>This field is populated with the location of the XML file created during EDI to XML conversion validation when the option Enable Audit Report Logging is enabled for a participant in the General tab for Inbound.</p> <p>This file is created only by the internal conversion engine and gives full reporting information of the transactions that are validated. A valid location must be entered for the trading host for the corresponding protocol. The protocol requires the parameter AuditReportLogging to be available to the TIBCO BusinessConnect Palette, which is available under the Advanced tab.</p> <p>This field will not be populated for Gateway or Service Protocols.</p>

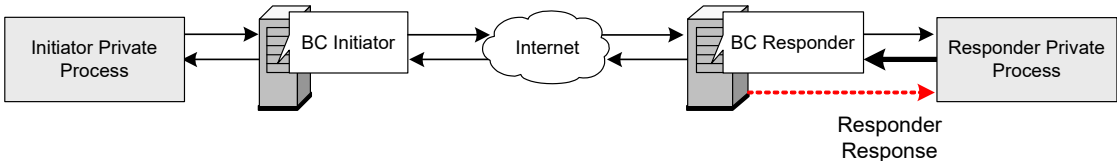
Responder Outbound Synchronous Response

The Responder private process sends this message as a response to an incoming synchronous request.



Synchronous request-response operations are supported in X12, including X12 HIPAA Edition.

Figure 18 Responder Outbound Synchronous Response



Rendezvous Subject Name *prefix.installation.standardID.RESPONDER.RESPONSE*
Example: AX.BC.BC-ACME.X12.RESPONDER.RESPONSE

JMS Queue Name *prefix.installation.RESPONDER.RESPONSE*
Example: AX.BC.BC-ACME.RESPONDER.RESPONSE

Message Name ResponderResponseTable 29 *ResponderResponse*

Field	Type	Required	Description
standardID	String	Yes	X12.
operationID	String	Yes	ID of operation to be performed. Format: <i>interchange_version/group_version/operation</i> Example: 00403/005010/850R
transactionID	String	Yes	The transaction ID of the original request. The private process is required to return the original transaction ID.
txName	String	Yes	Name of the transaction.
encoding	String	Yes	The encoding used in the response XML string. Example: UTF-8
statusCode	Integer	Yes	One of the error codes. See Appendix B, Status and Error Codes, on page 147 .
statusMsg	String	Yes	Represents the cause of the error. In this case, the response might be an error document. See Appendix B, Status and Error Codes, on page 147 .
closure	Any	Yes	TIBCO BusinessConnect generates the closure message, and sends it to the local private process in the ResponderRequest upon receiving a request for a synchronous request-response operation. The private process must return this closure contents in the ResponderResponse message to ensure that TIBCO BusinessConnect can match it with the original Responder request message.
response	Any	Yes	XML string representing the response message body or TIBCO Rendezvous/JMS representation of an XML file. The XML must conform to the .xsd generated by EDISIM for the transaction type. If a Responder uses this message class to send a HIPAA 275 Patient Information message, the HL7 message itself is embedded in this field. See the <i>TIBCO HL7 Integration Server Guide</i> for more information.

Table 29 ResponderResponse (Cont'd)

Field	Type	Required	Description
perMessage	Sequence	No	Overwrites the default parameters of the host and trading partner configuration on a per-message basis. The parameters for ASC X12 are: Group Header: Group Application Sender/Receiver ID (GS02, GS03) For overwriting parameters for the interchange header, the ID and Qualifier are validated against the list of domain identities for the host and the trading partner. See EnvelopeInfo , page 136.
responseFile	String	No	When the reponseFile is very large, this field is populated by the private process in place of the response. It must be a valid XML file.
CAQHOperationID	String	No	This field describes the pre-defined CAQH operation ID.
PayloadType	String	No	This field describes the type of the response file that is sent back to BusinessConnect.
PayloadID	String	No	This field is the ID for the payload of the message.
SenderID	String	No	This field is the ID of the sender of the message.
ReceiverID	String	No	This field is the ID of the receiver of the message.

Responder Acknowledgment to Partner

The Responder TIBCO BusinessConnect sends this notify message to the private process to indicate the success or failure of the posting of an acknowledgment to the Initiator.

If the Responder TIBCO BusinessConnect failed to successfully post the acknowledgment to the Initiator, the statusMsg field indicates the cause of the error.

Rendezvous Subject Name	<i>prefix.installation.standardID</i> .RESPONDER.ACK Example: AX.BC.BC-ACME.X12.RESPONDER.ACK
JMS Queue Name	<i>prefix.installation</i> .RESPONDER.ACK Example: AX.BC.BC-ACME.RESPONDER.ACK
Message Name	TransactionAckToTradingPartner

Table 30 ResponderAck

Field	Type	Required	Description
standardID	String	Yes	X12, EDIFACT
transmissionID	string	No	The transmissionID property is provided in the ResponderAck messages to indicate to the private process when it is scheduled for batching.
operationID	String	Yes	ID of operation to be performed. Format: <i>interchange_version/group_version/operation</i> Example: 00404/005010/997
transactionID	String	Yes	The acknowledgment response transaction ID for which the notify message is generated.
statusCode	Integer	Yes	One of the error codes. See Appendix B, Status and Error Codes, on page 147 .
statusMsg	String	Yes	Represents the cause of the error. In this case, the response might be an error document. See Appendix B, Status and Error Codes, on page 147 .
operationType	String	Yes	Notify or synchronous request-response.
controlNumInfo	Object	No	The control number information of the reconciled transaction. See ControlNumberInfo, page 138 .
responseFile	String	No	If the response message is to be passed by file reference, this field holds the name of the file. Note The value of this field is configuration for the outbound transport: it is the URL (or URI) pointing to the location where the acknowledgment was sent out.
resend	Boolean	No	Indicates if this request is generated based on re-processing of an inbound EDI document triggered by the resend link from the audit log viewer.
archivedFileRef	String	No	This field is populated with the location of the file where the raw EDI document is stored for acknowledgments. It is populated only when the option Log raw EDI is enabled under Trading Host > protocol with a valid location.

Table 30 ResponderAck (Cont'd)

Field	Type	Required	Description
auditReportFileRef	String	No	<p>This field is populated with the location of the XML file created during acknowledgment creation when the option Enable Audit Report Logging is enabled for a Participant under the Acknowledgement tab for outbound.</p> <p>This file is created only by the internal conversion engine and gives full reporting information of the transactions that are validated. A valid location must be entered for the trading host for the corresponding protocol. The protocol requires the parameter AuditReportLogging to be available to the TIBCO BusinessConnect Palette, which is available under the Advanced tab.</p> <p>This field will be populated only for the X12 and EDIFACT protocols.</p>

General Messages

CAQH Message

The Responder TIBCO BusinessConnect sends this notify message to the private process to indicate CAQH informational, warning, and error messages for both inbound and outbound transactions.

Rendezvous Subject Name *prefix.installation.standardID.CAQH*
 Example: AX.BC.BC-ACME.X12.CAQH

JMS Topic Name *prefix.installation.CAQH*
 Example: AX.BC.BC-ACME.CAQH

Message Name CAQHNotify

Table 31 CAQHMessage (Sheet 1 of 2)

Field	Type	Required	Description
standardID	String	Yes	Protocol name.
operationID	String	Yes	A three-part ID of the form: category/version_number/operation_Name.
transactionID	String	Yes	An ID unique within Initiator private processes environment for this transaction.
statusCode	Integer	Yes	Code indicating the status of the message; 200 for success. Otherwise, a code that represents the type of error.
statusMsg	String	Yes	OK, or the string representing the cause of the error. In this case, the response might be an error document.
CAQHOperatlonID	String	No	The pre-defined CAQH operation ID.
PayloadType	String	No	Payload Type specifies the type of payload included within a request, (e.g. HIPAA X12 transaction set 270, 276, 278, etc.).
ProcessingMode	String	No	Processing Mode indicates Batch or Real time processing mode (as defined by CORE).
PayloadID	String	No	This field is the ID for the payload of the message.
TimeStamp	String	No	This field indicates the time at which the response message is sent.

Table 31 CAQHMessage (Cont'd) (Sheet 2 of 2)

Field	Type	Required	Description
SenderID	String	No	This field is the ID of the sender of the message.
ReceiverID	String	No	The ID of the receiver of the message.
ErrorCode	String	No	The error code that is displayed.
ErrorMessages	String	No	The error message that is displayed against the corresponding error code.

Timeout Alert

The Responder TIBCO BusinessConnect sends this notify message to the private process to indicate timeout informational, warning, and error messages for both inbound and outbound transactions.

Rendezvous Subject Name	For ReceiptTimeoutErrorAdvisor: <i>prefix.installation.standardID.ERROR.TIMEOUT.RECEIPT</i> For AckTimeoutErrorAdvisory: <i>prefix.installation.standardID.ERROR.TIMEOUT.ACK</i>
JMS Topic Name	For ReceiptTimeoutErrorAdvisor: <i>prefix.installation.ERROR.TIMEOUT.RECEIPT</i> For AckTimeoutErrorAdvisory: <i>prefix.installation.ERROR.TIMEOUT.ACK</i>
Message Name	ReceiptTimeoutErrorAdvisory, AckTimeoutErrorAdvisory

Table 32 TimeoutAlert

Field	Type	Required	Description
tpName	String	No	Trading partner name
hostName	String	No	Host name
msgType	String	No	The type of timeout information, warning or error
standardID	String	Yes	X12, EDIFACT, TRADACOMS, and TEXT
operationID	String	No	ID of operation to be performed. Format: <i>interchange_version/group_version/operation</i> Example: 00403/005010/850
transactionID	String	Yes	The transaction ID for which the message is generated

Table 32 TimeoutAlert (Cont'd)

Field	Type	Required	Description
envelopeInfo	Object	No	The interchange and group envelope information of this transaction. The interchange envelope identifies the sender and receiver interchange qualifier and ID. The envelope identifies the sender and receiver application ID. See EnvelopeInfo , page 136.
controlNumInfo	Object	No	The control number information of this transaction that identifies itself in group envelope and interchange envelope of the incoming document. It contains the control numbers for the interchange envelope, the group envelope and the transaction itself. See ControlNumberInfo , page 138.

Validation Alert

The Responder TIBCO BusinessConnect sends this notify message to the private process to indicate validation informational, warning, and error messages for both inbound and outbound transactions.

Rendezvous Subject Name

prefix.installation.standardID.ERROR.VALIDATION

Example: AX.BC.BC-ACME.X12.ERROR.VALIDATION

JMS Topic Name

prefix.installation.ERROR.VALIDATION

Example: AX.BC.BC-ACME.ERROR.VALIDATION

Message Name

ValidationErrorAdvisory

Table 33 ValidationAlert

Field	Type	Required	Description
tpName	String	No	Trading partner name.
tpExtraInfo	String	No	Extra information associated with the trading partner as configured to be included in this message request but not as part of the inbound document. Use this information to assist in routing this request to the private process. Controlled by the Extra Info in Private Process Request field.
hostName	String	No	Host name.
msgType	String	No	The type of validation information, warning or error. Example: TRANSACTION_ERROR, INTERCHANGE_ERROR, GROUP_ERROR

Table 33 ValidationAlert (Cont'd)

Field	Type	Required	Description
hostInitiates	Boolean	No	Indicates if transaction is outbound or inbound.
standardID	String	Yes	X12, EDIFACT, TRADACOMS, and TEXT
operationID	String	No	ID of operation to be performed. Format: <i>interchange_version/group_version/operation</i> . Example: 00403/005010/850
transactionID	String	Yes	Transaction ID for which the message is generated.
sourceFile	String	No	Applicable only to inbound EDI document received from the FILE inbound transport. This is the full source path of the inbound EDI document.
receivedTime	String	No	Applicable only to inbound EDI document. The original GMT timestamp of the inbound request received by TIBCO BusinessConnect.
envelopeInfo	Object	No	The interchange (identifies the sender and receiver interchange qualifier and ID) and group envelope (identifies the sender and receiver application ID) information of this transaction. See EnvelopeInfo , page 136.
transactionInfo	Object	No	The raw EDI segments of this transaction and the header and trailer segment of its interchange envelope and group envelope. By default, transaction information is not included in the Responder request. To include transaction information in the Responder request, check Include Segments in Request to Private Process in the Operations Editor for this operation. TransactionInfo , page 138.
controlNumInfo	Object	No	The control number information of this transaction that identifies itself in group envelope and interchange envelope of the incoming document. It contains the control numbers for the interchange envelope, the group envelope and the transaction itself. See ControlNumberInfo , page 138.
errorInfoList	Sequence		Detail list of validation info and warning messages for with this transaction. See ConversionEngineInfo .

Error Message

An error message contains status information.

Rendezvous Subject Name *prefix.installation.standardID.ERROR*
 Example: AX.BC.BC-ACME.X12.ERROR

JMS Topic Name *prefix.installation.ERROR*
 Example: AX.BC.BC-ACME.ERROR

Message Name *GeneralErrorAdvisory*

Table 34 Error

Field	Type	Required	Description
operationID	String	Yes	ID of operation to be performed. Format: <i>interchange_version/group_version/operation</i> Example: 00403/005010/850
transactionID	String	Yes	Request-response ID for which the error message is generated.
tpName	String	No	Trading partner name.
statusCode	Integer	Yes	One of the status or error codes. See Appendix B, Status and Error Codes, on page 147 .
statusMsg	String	Yes	Represents the cause of the status or error. In this case, the response might be an error document. See Appendix B, Status and Error Codes, on page 147 .
details	String	Yes	More detailed description of the error. This can be an analysis of the problem.
extraInfo	Any	No	Details of the error.
standardID	String	Yes	Represents the Standard (such as X12) for which the error is reported.

Insight Report Message

Insight Report message is available for X12 and EDIFACT standards. It is enabled at the trading partner level and either published on RCVN, or sent to the queue during inbound, outbound, batching, and non-batching scenarios, including Synchronous RequestResponse cases that apply only for the X12 protocol. This message will publish a report based on the Instream results file along with EDI data file and other information required by the Insight reporting. The report of XML files contains:

- Messages received from the trading partner (both synchronous and asynchronous)
- Messages sent to the trading partner (synchronous response as well as regular non-batch messages)

- Batched messages sent to the trading partner

Rendezvous Subject Name

prefix.installation.standardID.INSIGHT.REPORT
Example: AX.BC.BC-ACME.X12.INSIGHT.REPORT

JMS Queue Name

prefix.installation.INSIGHT.REPORT
Example: AX.BC.BC-ACME.INSIGHT.REPORT



Insight Report Message should be consumed only by the BusinessConnect Insight or Transaction Insight Product.

Message Name

InsightReportNotify

Table 35 *Insight Report*

Field	Type	Required	Description
standardID	String	Yes	X12, EDIFACT
operationID	String	Yes	EDI/Inbound/Interchange for inbound or EDI/Outbound/Interchange for outbound
transactionID	String	No	Currently processed Interchange: either the documentID, or transactionID from the private process (for non-batch)
ediDataFile	String	Yes	EDI data file used by the Instream conversion engine
instreamResultsFile	String	Yes	Instream results file that processed the ediDataFile
originalSourceFile	String	No	File that is received from Trading Partner and only populated during Inbound

Push Real-Time Partner and Guideline Updates Message

Push real-time partner and guideline update messages are supported only for X12 and EDIFACT standards. A message to the private process to be consumed by TIBCO BusinessConnect Insight or Foresight Transaction Insight product is sent for synchronization of key partner information and guideline information between the two systems.

The message is sent when the updates are made in the TIBCO BusinessConnect administrator GUI in the information for partners, operation editor, or at the business agreements protocol binding level.

Rendezvous Subject Name

prefix.installation.standardID.CSUPDATE

Example: AX.BC.BC-ACME.X12.CSUPDATE

JMS Queue Name *prefix.installation.CSUPDATE*
 Example: AX.BC.BC-ACME.CSUPDATE

Message Name CSUpdate

Table 36 CSUpdateNotify Object

Field	Type	Required	Description
standardID	String	Yes	X12 or EDIFACT
operationID	String	No	When a particular operation is edited in the operations editor and saved. The value of the operation that is edited will be published; for example 00403/005010/850 for X12.
participantName	String	No	Name of the Participant being edited. This could be a host or a partner type.
participantType	String	No	Type of Participant: host or partner
agreementName	String	No	When the guideline is being updated from the business agreement operation bindings override, the name of the business agreement is published in this field.
operationBindingType	String	No	The value indicates whether the update is for the "Host can initiate" or "Partner can initiate" setting in the business agreement's operation binding.
actionType	String	Yes	The value indicates whether the UI action is create or update on either the participant, business agreement, or operations editor level.
operationTime	String	Yes	The value indicates the time of update that happened.
objectID	String	Yes	Object Identifier Value for the type of object that was changed.
propertyList	CSUpdatedetailInfo	No	This is a list of property list values that are changed by the user.

Table 37 CSUpdateDetailInfo Object

Field	Type	Required	Description
propertyName	String	No	<p>Name of the property that is being modified in the UI in Administrator. Following are the list of values that can be published:</p> <p>X12:</p> <p>For elements of ISA Segment</p> <p>ISA01Qualifier</p> <p>ISA02Information</p> <p>ISA03Qualifier</p> <p>ISA04Information</p> <p>ISA05SenderQualifier</p> <p>ISA06SenderID</p> <p>ISA07ReceiverQualifier</p> <p>ISA08ReceiverID</p> <p>ISA09Date</p> <p>ISA10Time</p> <p>ISA11ControlStandard</p> <p>ISA11RepeatingSeparator</p> <p>ISA14AckRequested</p> <p>ISA15UsageIndicator</p> <p>ISA16CompElemSep</p> <p>For elements of GS Segment</p> <p>GS02Sender</p> <p>GS03Receiver</p> <p>GS04Date</p> <p>GS05Time</p> <p>GS08GrpVersion</p>

Table 37 CSUpdateDetailInfo Object (Cont'd)

Field	Type	Required	Description
propertyName (continued)	String	No	EDIFACT: For elements of UNB Segment UNBS001Identifier UNBS001ServiceCodeList UNBS001CharacterEncoding UNBS002SenderID UNBS002SenderQualifier UNBS002InternalID UNBS002SubInternalID UNBS003ReceiverID UNBS003ReceiverQualifier UNBS003InternalID UNBS003SubInternalID UNBS004Date UNBS004Time UNBS005Reference UNBS005Qualifier UNB0026ApplicationRef UNB0029ProcessingCode UNB0031AckRequested UNB0032AgreementIdentifier UNB0035TestIndicator For elements of UNG Segment UNGS006Qualifier UNGS006ID UNGS007Qualifier UNGS007ID UNGS004Date UNGS004Time UNGS008Code UNG0058Password
oldValue	String	No	Value that was before the change, if any.
newValue	String	No	New value that was successfully updated.

Objects Included in Private Messages

The following objects can be included in private messages:

- [TPName, page 136](#)
- [EnvelopeInfo, page 136](#)
- [InterchangeInfo, page 137](#)
- [GroupInfo, page 137](#)
- [Identity, page 137](#)
- [ControlNumberInfo, page 138](#)
- [TransactionInfo, page 138](#)
- [ConversionEngineInfo, page 139](#)
- [ConversionEngineInfo, page 139](#)
- [Attachment, page 140](#)

TPName

The TPName object (ae/BC/TPName) has the following field:

Table 38 TPName Object

Field	Type	Description
TPName	String	Trading partner name. Example: CarCo.

EnvelopeInfo

The EnvelopeInfo object (ae/EDI/EnvelopeInfo) has the following fields:

Table 39 EnvelopeInfo Object

Field	Type	Description
interchange	ae/EDI/InterchangeInfo. See InterchangeInfo on page 137 .	Identities of the interchange sender and receiver.
group	ae/EDI/GroupInfo. See GroupInfo on page 137 .	Identities of the group sender and receiver application.

InterchangeInfo

The InterchangeInfo object (ae/EDI/InterchangeInfo) has the following fields:

Table 40 InterchangeInfo Object

Field	Type	Description
sender	ae/EDI/Identity. See Identity on page 137 .	Identity of the sender in the interchange header.
receiver	ae/EDI/Identity. See Identity on page 137 .	Identity of the receiver in the interchange header.
interchangeHeaderInfo	ae/EDI/InterchangeHeaderInfo Available for X12, TEXT, and EDIFACT	Interchange header information that can be passed from a private process on InitiatorRequest only to TIBCO BusinessConnect to overwrite interchange header values on a perMessage value. See Table 23, Initiator Request , field perMessage.

GroupInfo

The GroupInfo object (ae/EDI/GroupInfo) has the following fields:

Table 41 GroupInfo Object

Field	Type	Description
senderApp	ae/EDI/Identity. See Identity on page 137 .	Identity of the sender application in the group header.
receiverApp	ae/EDI/Identity. See Identity on page 137 .	Identity of the receiver application in the group header.

Identity

The Identity object (ae/EDI/Identity) has the following fields:

Table 42 Identity Object

Field	Type	Description
qualifier	String	Qualifier of the sender or receiver.
ID	String	Qualifier identifier of the sender or receiver.

ControlNumberInfo

The `ControlNumberInfo` object (`ae/EDI/ControlNumberInfo`) has the following fields:

Table 43 *ControlNumberInfo Object*

Field	Type	Description
interchange	String	Control number of the interchange envelope.
group	String	Control number of the group envelope.
transaction	String	Control number of the transaction within the interchange or group envelope.

TransactionInfo

The `TransactionInfo` object (`ae/EDI/TransactionInfo`) has the following fields:

Table 44 *TransactionInfo Object*

Field	Type	Description
interchange Header	String	The reference interchange header segment of the transaction.
groupHeader	String	The reference group header segment of the transaction. ASC X12 example: GS segment.
transaction Body	String	The raw EDI segments of the transaction. ASC X12 example: All segments between a pair of ST/SE segments.
groupTrailer	String	The reference group trailer segment of the transaction. ASC X12 example: GE segment.
interchange Trailer	String	The reference interchange trailer segment of the transaction. ASC X12 example: ISE segment.
userKey	String	Values of the user keys delimited by colon (:) as defined by the EDISIM Standards Editor DSR Mark.
userKeyName	String	Names of the user keys delimited by colon (:) as defined by the EDISIM Standards Editor DSR Mark for the guideline.
txExtraInfo	String	Extra information to be included with the transaction. It is not part of the raw EDI document. Use this to assist in routing the transaction to the backend private process. Example: Development, Production. Controlled by the field Extra Transaction Info.

Table 44 *TransactionInfo Object (Cont'd)*

Field	Type	Description
requestFile	String	The full path of the raw EDI segments for the transaction that are stored on file. This attribute is populated when the Log Raw EDI Segments to File check box for the transaction level is selected.

ConversionEngineInfo

The `ConversionEngineInfo` object (ae/EDI/ConversionEngineInfo) describes detail information of an EDI validation information, warning or error that contains the following fields:

Table 45 *ConversionEngineInfo Object*

Field	Type	Description
errorId	String	Unique identifier for the error.
errorCategory	String	Category code. Example: Rejecting
errorSeverity	String:	Severity level. Example: Normal
errorType	String	Not Used. Reserved for future.
errorMsg	String	Detailed description of the validation info, warning, or error. Example: Element BGN01 is a coded list element. Code 0X is not allowed. Segment BGN is defined in the guideline at position 0200. This error was detected at: Segment Count: 2 Element Position: 1 Character: 193 through 195.
errorNode	String	Not Used. Reserved for future.
errorContext	String	Context of the error node where the error occurred that describes the location of the error node as specified in the guideline. Example: Transaction-ORDERS Loop-Group_25 Segment-MOA Composite-C516 Element=5004
errorData	String	The primary data at the error node. This is typically the data of an element in a segment that are invalid. Example: 2192,42
errorDataParent	String	The parent data of the error node. This is typically the data of the segment that contains the invalid element. Example: MOA+203,42

Table 45 ConversionEngineInfo Object (Cont'd)

Field	Type	Description
errorSegmentCount	Integer	Segment count of the transaction where the error occurred. Example: 21
errorElementCount	Integer	Element count of the segment where the error occurred. Example: 1
errorCompositeCount	Integer	Composite count of the segment where the error occurred. Example: 1
errorSubElementCount	Integer	Sub-element count of the composite where the error occurred. Example: 2
errorCode String	String	Not Used. Reserved for future.

Attachment

The attachment object (ae/EDI/attachment) has the following fields:

Table 46 Attachment Object

Field	Type	Description
name	String	Name of the attachment.
content-type	String	Content type of attachment.
content-id	String	The unique identifier used to identify an attachment in outgoing document.
content	Base64 Binary	Content of the attachment.
fileName	String	A file reference can be sent as an attachment. Note: Attachments are ignored if both the content and fileName elements are empty, if both elements are configured, the content element takes precedence.

Appendix A **Property Reference**

This section lists and describes the EDI plugin properties, which are accessible through TIBCO Administrator.

Topics

- [Property Reference, page 142](#)

Property Reference

Using TIBCO Administrator you can access and modify a large set of properties for TIBCO BusinessConnect EDI Protocol powered by Instream. To access these properties, perform these steps:

1. In the TIBCO BusinessConnect console, select **System Settings > Activated Protocol Plug-ins and Properties**.
2. Click **tibEDI**.

Table 47 EDI Plugin Property Reference

Property	Explanation
edi.auditErrorReport. ignoreWarnInfo	(HIPAA only) Suppress the audit logging and error advisory of validation warnings. Default is false.
edi.data.encoding	Controls the output encoding of the resulting outbound EDI document and the encoding TIBCO BusinessConnect EDI Protocol powered by Instream uses to interpret the inbound EDI document. Default is set to the default encoding of the running system.
edi.inbound.xDataFile.encoding	Controls the output encoding of the converted XML request when it is written to file for inbound EDI to XML conversion. Default is UTF-16.
edi.edifact.interchangeControlNumMinLen	Minimal length of the Interchange control number for EDIFACT
edi.LongRunningJob.ThreadPool.EDIToXML.FileBased. maxCount	Controls the maximum thread pool size used by file based EDI validation and conversion. Restart the TIBCO BusinessConnect engine for the changes to take effect.
edi.LongRunningJob.ThreadPool.EDIToXML.InMemory. maxCount	Controls the maximum thread pool size used by in-memory based validation and conversion. Restart the TIBCO BusinessConnect engine for the changes to take effect.
edi.LongRunningJob.ThreadPool.XMLToEDI.maxCount	Controls the maximum thread pool size used by XML to EDI validation and conversion. Restart the TIBCO BusinessConnect engine for the changes to take effect.

Table 47 EDI Plugin Property Reference (Cont'd)

Property	Explanation
edi.outbound.contentDisposition.fileExtension	<p>Default file extension to use on the Content-Disposition MIME header field if TIBCO BusinessConnect generates the fileName.</p> <p>Note: Specify the period before the extension if you needs a different default.</p>
edi.ack.timeout.pollingInterval	<p>Controls the polling interval in seconds for the acknowledgment timeout management.</p> <p>Note: Even if the timeout period is set to a very small value, the minimum timeout value will always be based on the property edi.ack.timeout.pollingInterval.</p>
edi.inbound.xmlStreamBuf	<p>Buffer size in bytes to control the amount of XML to be stored in memory for each transaction before flushing to the interchange buffer. Default: 10K.</p>
edi.append.UserKeyName.to.audit.log	<p>When the value of this boolean property is set to "true", TIBCO BusinessConnect EDI Protocol powered by Instream shows both the User Key name and corresponding value in Audit logs.</p> <p>Default: false.</p>
edi.x12.enable.adjustIEACountForTA1.<TPName>	<p>When the value of this boolean property is set to "false", IEA01 value of "0" is expected in TA1 response.</p> <p>When the value is set to "true", IEA01 value of "1" is expected in the TA1 response.</p> <p>The changes will affect only those trading partners that are mentioned in <TP NAME> of the property.</p> <p>Default: false</p>
edi.x12.enable.adjustIEACountForTA1	<p>When the value of this boolean property is set to "false", IEA01 value of "0" is expected in TA1 response.</p> <p>When the value is set to "true", IEA01 value of "1" is expected in the TA1 response.</p> <p>The changes will affect all the trading partners.</p> <p>Default: false</p>

Table 47 EDI Plugin Property Reference (Cont'd)

Property	Explanation
edi.add.engine.name.to.logging.file	<p>This boolean property controls whether to append the engine name to the logged file when host logging is used.</p> <p>When the value of this property is set to "false", engine name will not be appended to the logged file name of the host logging feature.</p> <p>Default: false</p>
edi.skip.default.encoding.conversion	<p>When the value of this boolean property is set to "true", the default encoding set in the Inbound EDI Data Encoding field in EDI configuration is skipped. The changes to this property will affect all the trading partners.</p> <p>Default: false</p>
edi.skip.default.encoding.conversion.<TPname>	<p>When the value of this boolean property is set to "true", the default encoding set in the Inbound EDI Data Encoding field in EDI configuration is skipped. The changes to this property will affect only those trading partners that are mentioned in <TP NAME> of the property.</p> <p>Default: false</p>
edi.x12.outbound.ack.creation.fromTP	<p>Enabling this property will override the interchange and group header details for an outbound acknowledge from TP configuration for X12 protocol.</p> <p>Default: True</p>
edi.edifact.outbound.ack.creation.fromTP	<p>Enabling this property will override the interchange and group header details for an outbound ack from TP configuration for EDIFACT protocol</p>
edi.Translation.ThreadPool.maxCount	<p>Controls the maximum thread pool size used by EDI to XML transaction worker. The default is 16.</p>
edi.Translation.ThreadPool.minCount	<p>Controls the minimum thread pool size used by EDI to XML transaction worker. The default is 8</p>
edi.translator.forceThreadLocking	<p>When the value of this boolean property is set to "true", thread locking is enabled and the translator handles conversion task in a sequential order.</p> <p>Default: false</p>

Appendix B **Status and Error Codes**

This appendix describes different kinds of status and error codes that can occur in TIBCO BusinessConnect EDI Protocol powered by Instream.

The information in this appendix applies to TIBCO BusinessConnect EDI Protocol powered by Instream and TIBCO BusinessConnect EDI Protocol HIPAA Edition powered by Instream.

Topics


- [HTTP Status Codes: 200-510, page 148](#)
- [Status Codes: 600-699, page 149](#)

HTTP Status Codes: 200-510

This section describes status codes that are used in TIBCO BusinessConnect EDI Protocol powered by Instream. The 200-510 status codes are standard HTTP codes.

These status codes may display in the following areas:

- The `statusCode` field. See [Initiator Inbound Response on page 110](#) for an example of this.
- The `statusMsg` field. See [Initiator Inbound Response on page 110](#) for an example of this.
- The `Description` field in the logs. The contents of the `statusCode` and `statusMsg` fields may or may not display in this field, depending on the transaction. See [Transaction Details View on page 86](#).



In the `statusMsg` field, the text explains the meaning of the code, but does not specify the value that the `statusMsg` field can contain.

Table 48 HTTP Status Codes

Code (statusCode field)	Description (statusMsg field)	Role	Category	Resolution
200	OK			Transport-level acknowledgment.
201-299	HTTP(S) OK codes			Transport-level acknowledgments.
300 - 499	HTTP(S) error codes	Error		
510	No valid HTTP response. The most likely cause is that the Responder did not send a proper HTTP response. If the Responder is TIBCO BusinessConnect, then the Initiator may have sent a trading partner identity that was not recognized by the Responder TIBCO BusinessConnect.	Error	Configurat ion	

Status Codes: 600-699

This section describes the status codes which are specific to TIBCO BusinessConnect EDI Protocol powered by Instream.

These status codes may display in the following areas:

- The `statusCode` field in private messages. See [Initiator Inbound Response on page 110](#) for an example of this.
- The `statusMsg` field. See [Initiator Inbound Response on page 110](#) for an example of this.
- The `Description` field in the logs. The contents of the `statusCode` and `statusMsg` fields may or may not display in this field, depending on the transaction. See [Transaction Details View on page 86](#).



In the `statusMsg` field, the text explains the meaning of the code, but does not specify the value that the `statusMsg` field can contain.

Table 49 TIBCO BusinessConnect EDI Protocol powered by Instream Status Codes

Code (statusCode Field)	Description (statusMsg Field)	Role	Category
601	FTP error	Error	
605	Local posting error	Error	
620	No HTTP reply after waiting for HTTP acknowledgment wait time.	Error	
630	Mime encoding error	Error	Security
631	Encryption error	Error	Security
632	Decryption error ERROR_SMIME_DECRYPT	Error	Security
633	Signature error	Error	Security
634	Signature verification error ERROR_SMIME_VERIFY	Error	Security
635	Compression error	Error	
636	Decompression error	Error	
637	Receipt request error	Error	

Table 49 TIBCO BusinessConnect EDI Protocol powered by Instream Status Codes (Cont'd)

Code (statusCode Field)	Description (statusMsg Field)	Role	Category
638	Receipt match error	Error	
639	MDN receipt request store error	Error	
640	Receipt disposition error	Error	
641	HTTP headers error	Error	
660	Configuration error	Error	Configuration
661	Bad configuration - No matching TP email address ERROR_TP_CONFIG	Error	Configuration
662	Missing decryption key	Error	Configuration
663	Missing authentication certificate	Error	Configuration
699	Internal system error	Error	System
1000 - 1999	Private-party defined codes		

Appendix C **EDI Migration and Operations Updater Tools**

This appendix explains how to use the EDI Migration and Operations Updater tools.

Topics

- [EDI Migration Tool, page 152](#)
- [Operations Updater Tool, page 155](#)

EDI Migration Tool

The EDI Migration tool is a command line tool used to migrate TIBCO BusinessConnect EDI Protocol powered by Instream configuration repository data from the release 5.x to the release 6.x.



- You cannot use this tool to migrate between TIBCO BusinessConnect EDI Protocol powered by Instream 6.x, such as from version 6.4 to 6.6. Use the normal TIBCO BusinessConnect configuration import from the GUI or from the tool BAppManage.
- To migrate TIBCO BusinessConnect EDI Protocol powered by Instream 5.4.0 installed with TIBCO BusinessConnect 5.3.3 to TIBCO BusinessConnect EDI Protocol powered by Instream 6.x installed with TIBCO BusinessConnect 6.x, you must first migrate it to version 6.x installed with TIBCO BusinessConnect 5.3.3, and then migrate it to version 6.x installed with TIBCO BusinessConnect 6.x.

To perform migration using the EDI Migration tool, the .csx file exported from the release 5.x is required. In addition, you have to configure the file path for this .csx file and the JDBC information for the target repository.



Only complete configuration store migrations are supported using the EDI Migrator tool; participants only and operations only migrations are not supported.

The EDI Migration tool is located in the `BC_HOME\protocols\tibedi\tools\migration` directory consists of the following files:

- **edimigrator.exe** Executable file to perform the migration
- **edimigrator.tra** This file contains various environmental and other variables to be set for using this tool.
- **tibedimigration.jar** .This is the executable JAR file for this tool.
- **edimigrator.log** This file is generated after migration is performed and contains the migration logs.

Steps for migrating TIBCO BusinessConnect EDI Protocol powered by Instream using this tool are explained in the Migration and Compatibility section in *TIBCO BusinessConnect EDI Protocol powered by Instream Release Notes*.

Using the EDI Migration Tool

After you run `edimigrator.exe` by using a command line, the following information is displayed:

```
usage: edimigrator.exe [-help] [-csxPassword <csx password>] -targetUser <jdbc user> -csxFilepath <csx file path> -targetUrl <jdbc url>
-targetPassword <jdbc
password> -targetDriver <jdbc driver>
```

```
-----
-help                print help information
-csxPassword <csx password>    password specified in export process
-targetUser <jdbc user>        target bc database username
-csxFilepath <csx file path>    complete file path of the csx that needs to be migrated
-targetUrl <jdbc url>          target bc database jdbc url
-targetPassword <jdbc password> target bc database password
-targetDriver <jdbc driver>     target bc database jdbc driver
```

Changing URI Information when Migrating with the EDI Migration Tool

Any transport URL information that was configured under **Trading Partner > Enabled Protocol** from the previous version of TIBCO BusinessConnect EDI Protocol powered by Instream will not be changed. This URL is the outbound URL that the protocol will communicate to the trading partner. If users require this information to be updated, they must use TIBCO BusinessConnect ConfigStore Management Interface Protocol to modify the URLs.

The inbound URL for TIBCO BusinessConnect EDI Protocol powered by Instream requires the appropriate URI change for X12, Gateway, TEXT, TRADACOMS, and EDIFACT as mentioned in [Exchanging URI Definitions on page 38](#), and this information must be indicated to their trading partners as a part of the URI information exchange.

Migrating User-Specific Properties

User-specific properties defined under **System Settings> Activated Protocol Plug-ins and Properties> tibEDI** will not be migrated to TIBCO BusinessConnect EDI Protocol powered by Instream version 6.x unless they are given by TIBCO Support, or are pertinent to the new conversion engine.

To continue using these properties, users have to enter them manually under **System Settings> Activated Protocol Plug-ins and Properties> tibEDI**.

Migrating from the Version 5.x to 6.x and Later

Certain manual changes are required from users migrating from TIBCO BusinessConnect EDI Protocol 5.x to TIBCO BusinessConnect EDI Protocol powered by Instream Version 6.x and later:

- Users who had previously used the Service protocol have to manually change the following information:

- The package name of Service protocol interfaces has been changed to `com.tibco.ax.tibedi.runtime.service` and implementers of these interfaces need to refactor their code. The new protocol name is `Service`, rather than `Service`.
- TIBCO BusinessConnect EDI Protocol 5.x runtime table names have been changed from `EDI_` to `TIBEDI_` in TIBCO BusinessConnect EDI Protocol powered by Instream version 6.x and above. Users who depend on these table names for the Service protocol must change these names appropriately in their implementation.
- Users migrating from TIBCO BusinessConnect EDI version 5.x who had previously used custom scripts should change them manually to adapt to TIBCO BusinessConnect EDI Protocol powered by Instream version 6.x and later as follows:
 - The package name of EDI-specific custom script object interfaces have been changed to `com.tibco.ax.tibedi.runtime.transport` and implementers of these interfaces need to refactor their code if they are using TIBCO BusinessConnect EDI Protocol powered by Instream version 6.x and later.
- Guidelines, schemas etc files under operations in the operations editor and under the operation bindings in business agreements are not migrated from 5.x to 6.x by using the EDI migration tool. Instead, users have to run explicitly the operations updater tool to upload these files after migration.

Operations Updater Tool

The Operations Updater tool is used to export xml files to the partner and business agreement configurations.

After migrating from TIBCO BusinessConnect EDI Protocol 5.x to TIBCO BusinessConnect EDI Protocol powered by Instream 6.x, all guidelines, schemas, and validator profiles are not migrated. Some new fields also have to be updated and set after migration, such as root element and the map files that are pertaining to the new Instream engine.

The typical flow of steps after migration is as follows:

1. Note that no files are under operations or operations editor after migration.
2. Run the Operations Updater tool with the `export` option, so that an XML file is created with all operations and operation bindings information.
3. Generate all the required files: guidelines, validator profiles (if needed), XSD schemas, and map files as per 6.x standards.
4. Update the exported XML file with all files information.
5. Run `import` again so that all the files information is imported back into the BusinessConnect configuration. After the import, all migrated operations are ready to be used for runtime.

The Operations Updater tool is located in the `BC_HOME\protocols\tibedi\tools\operationsupdater` directory and consists of the following files:

- **operationsupdater.exe** Executable file to perform the update
- **operationsupdater.tra** Operations Updater .tra file



The Operations Updater tool now supports the import of EDIFACT to TEXT guideline files for EDIFACT protocol.

Using the Operations Updater Tool

This procedure describes how to use the Operations Updater tool.

1. Make a new guideline file using the EDISIM program.
2. Use the EDISIM export options to export the XSD and the map files.
3. Run the `operationsupdater` tool in export mode to export the current configuration into a readable format.
4. Modify the exported XML file to map the new guideline, XSD and map files. You can also add the appropriate Root Element name based on the XSD Root Element name.

5. Import the modified XML file.

After you run `operationsupdater.exe` by using the command line, the following information is displayed.

```
usage: operationsupdater.exe [-jdbcPassword <db password>] [-export] [-help]
-overwrite [-jdbcUser <db user name>] [-import] [-toFile <filepath>] [-fromFile
  <filepath>] [-jdbcUrl <jdbc url>] [-jdbcDriver <jdbc driver class>]
-----
-jdbcPassword <db password>      jdbc user password
-export                          export operation related information
-help                            print help information
-overwrite                       whether overwrite the existing files
-jdbcUser <db user name>         jdbc user name
-import                          import operation related information
-toFile <filepath>               file path will be used to put the exported
data
-fromFile <filepath>             file path will be used to retrieve the xml
content
-jdbcUrl <jdbc url>              jdbc URL e.g. jdbc:mysql://localhost:3306/
dbname
-jdbcDriver <jdbc driver class>  jdbc driver e.g. com.mysql.jdbc.Driver
```

- note:
- 1. for import/export jdbc options are required, because we have to create database connection by these options
 - 2. for import option [fromFile] is required, because we need a file path to retrieve the xml content to do import action
 - 3. for export option [toFile] is optional, if there is no option [toFile] we will use %BC_HOME%/protocols/tibedi/tools/migration/migration_data.xml instead
 - 4. [overwrite] option only works for command [import]

Appendix D Database Schema Definition

This appendix provides details about Runtime schemas to create your own archiving strategy to replace the deprecated Archiver Tool from the previous releases.

Topics

- [EDI Batch Schema Details, page 157](#)

EDI Batch Schema Details

EDI batch schemas hold all information for EDI batch transactions. The EDI batch schema details of the EDI tables are explained in [Table 50](#).

Table 50 Runtime Schema Details

Name	Documentation
TIBEDI_BATCH_TX_BIN	This table stores the EDI data that has been batched.
TIBEDI_BATCH_TX	This table stores the metadata for each transaction that is triggered when batching is enabled for any of the protocols. This records data such as the trading partner name, time stamp, interchange header, and group related data. It also has a reference to the record in TIBEDI_BATCH_TX_BIN (BININDEX) which has the actual transaction EDI document and TIBEDI_BATCH_TASK (BATCHID).
TIBEDI_BATCH_TASK	This table links the BatchID with Schd_TaskID and references to the record in the BC_SCHEDULED_TASK table (TASKID). This helps in scheduling the batching of documents depending on the transaction count or time interval.

TIBEDI_BATCH_TASK

Details for the table TIBEDI_BATCH_TASK are explained in [Table 51 on page 158](#)

Table 51 TIBEDI_BATCH_TASK: Details

Name	Value
Data Model	Physical
Documentation	This table links the BatchID with Schd_TaskID. This helps in scheduling the batching of documents depending on the transaction count or time interval.
User IDLast Numeric Value	0
DDLClause	BATCHID VARCHAR2(255) PRIMARY KEY, SCHD_TASKID VARCHAR2(255), PP_BATCH_ID VARCHAR2(255)

Columns summary for the table TIBEDI_BATCH_TASK is shown in [Table 52 on page 158](#)

Table 52 TIBEDI_BATCH_TASK: Columns Summary (Sheet 1 of 3)

Name	Data Type	Constraints	Nullable	Documentation
BATCHID	varchar2(255)	PK	No	Uniquely generated by TIBCO BusinessConnect.
SCHD_TASKID	varchar2(255)		No	Corresponds to column in table bc_scheduled_task.
PP_BATCH_ID	varchar2(255)		No	The value of "txnGroupingID" specified from Private Process Send Request/Notification palette.

TIBEDI_BATCH_TX

Details for the table TIBEDI_BATCH_TX are explained in [Table 53 on page 159](#)

Table 53 (Sheet 1 of 2)TIBEDI_BATCH_TX

Data Model	Physical
Documentation	<p>This table stores the metadata for each transaction that is triggered when batching is enabled for any of the protocols. This records data such as the trading partner name, time stamp, interchange header, and group related data. It also has a reference to the record in TIBEDI_BATCH_TX_BIN (BININDEX) which has the actual transaction EDI document and TIBEDI_BATCH_TASK (BATCHID).</p> <p>Note: This table will grow based on the exchanged transactions. This table should be added as a part of the archiving process.</p> <p>A typical sample archive query for an Oracle database is shown below:</p> <pre>Select * from TIBEDI_BATCH_TX where STATUS like '%COMPLETED%' OR '%ERROR%' AND TS <= TO_TIMESTAMP(<from date>,'YYYY-MM-DD HH24:MI:SS.FF') AND TS >= TO_TIMESTAMP(<to date>,'YYYY-MM-DD HH24:MI:SS.FF')</pre> <p>A typical sample archive query for Microsoft SQL database is shown below:</p> <pre>Select * from TIBEDI_BATCH_TX where STATUS like '%COMPLETED%' or '%ERROR%' and PROTOCOLNAME = '<protocol>' AND TS <= '<to date>' and TS >= '<from date>'</pre>
User IDLast Numeric Value	0

Table 53 (Cont'd) (Sheet 2 of 2)TIBEDI_BATCH_TX

DDLClause	BININDEX NUMBER(18) PRIMARY KEY, TRANSPORTINGTP VARCHAR2(128), INTERCHANGEHEADER VARCHAR2(256), GROUPHEADER VARCHAR2(256), BATCHID VARCHAR2(255) NULL, STATUS VARCHAR2(64) NULL, PROTOCOLNAME VARCHAR2(32) NULL, INSTALLATIONNAME VARCHAR2(32) NULL, PROTOCOLVERSION VARCHAR2(32) NULL, OPERATIONID VARCHAR2(512) NULL, DOCUMENTID VARCHAR2(512) NULL, DOCUMENTTYPE VARCHAR2(32) NULL, TPNAME VARCHAR2(128) NULL, TPDOMAIN VARCHAR2(32) NULL, TPID VARCHAR2(32) NULL, HOSTNAME VARCHAR2(128) NULL, HOSTDOMAIN VARCHAR2(32) NULL, HOSTID VARCHAR2(32) NULL, HOSTINITIATES VARCHAR2(5) NULL, CAQHOBJ BLOB NULL, TS TIMESTAMP DEFAULT SYSTIMESTAMP, LOGCONTEXTID VARCHAR2(512) NULL, CONSTRAINT TIBEDI_BCH_TX_FKEY FOREIGN KEY (BATCHID) REFERENCES TIBEDI_BATCH_TASK(BATCHID) ON DELETE CASCADE

Columns summary for the table TIBEDI_BATCH_TX is shown in [Table 54 on page 160](#)

Table 54 TIBEDI_BATCH_TX: Columns Summary (Sheet 1 of 3)

Name	Data Type	Constraints	Nullable	Documentation
BININDEX	number(18)	PK	No	Uniquely generated by TIBCO BusinessConnect.
TRANSPORTINGTP	varchar2(128)		No	Transporting Trading Partner Name.
INTERCHANGEHEA DER	varchar2(256)		No	Has basic interchange header information.

Table 54 TIBEDI_BATCH_TX: Columns Summary (Cont'd) (Sheet 2 of 3)

Name	Data Type	Constraints	Nullable	Documentation
GROUPHEADER	varchar2(256)		No	Has basic group header information.
BATCHID	varchar2(255)		Yes	Refers to the BATCHID in TIBEDI_BATCH_TASK table.
STATUS	varchar2(64)		Yes	Status of the batch and transaction.
PROTOCOLNAME	varchar2(32)		Yes	Protocol name.
INSTALLATIONNAME	varchar2(32)		Yes	BusinessConnect Admin Installation name.
PROTOCOLVERSION	varchar2(32)		Yes	Protocol version.
OPERATIONID	varchar2(512)		Yes	Basic operation ID.
DOCUMENTID	varchar2(512)		Yes	Document ID generated for each batching transaction.
DOCUMENTTYPE	varchar2(32)		Yes	Similar to Operation ID.
TPNAME	varchar2(128)		Yes	Name of the Trading Partner.
TPDOMAIN	varchar2(32)		Yes	Domain name of the Trading Partner.
TPID	varchar2(32)		Yes	ID of the Trading Partner.
HOSTNAME	varchar2(32)		Yes	Name of the Host.
HOSTDOMAIN	varchar2(32)		Yes	Domain name of the Host.
HOSTID	varchar2(32)		Yes	ID of the HOST.
HOSTINITIATES	varchar2(5)		Yes	Becomes true if generated from the machine where BC is installed.
CAQHOBJ	BLOB		No	Store CAQH Object when the batch transaction require CAQH package.

Table 54 TIBEDI_BATCH_TX: Columns Summary (Cont'd) (Sheet 3 of 3)

Name	Data Type	Constraints	Nullable	Documentation
TS	TIMESTAMP		No	The time at which the message is sent.
LOGCONTEXTID	varchar2(512)		Yes	Save the value of BC_TRANS_ID column value of BC_TRANSACTIONS table.

TIBEDI_BATCH_TX_BIN

Details for the table TIBEDI_BATCH_TX_BIN are explained in [Table 55 on page 163](#)

Table 55 TIBEDI_BATCH_TX_BIN: Details

Name	Value
Data Model	Physical
Documentation	<p>This table stores the EDI data that has been batched.</p> <p>Note: This table will grow based on the exchanged transactions. This table should be added as a part of the archiving process.</p> <p>A typical sample archive query for an Oracle database is shown below:</p> <pre>Select * from TIBEDI_BATCH_TX_BIN where BININDEX in (select BININDEX from TIBEDI_BATCH_TX where STATUS like '%COMPLETED%' OR '%ERROR%' AND TS <= TO_TIMESTAMP(<from date>, 'YYYY-MM-DD HH24:MI:SS.FF') AND TS >= TO_TIMESTAMP(<to date>, 'YYYY-MM-DD HH24:MI:SS.FF'))</pre> <p>A typical sample archive query for Microsoft SQL database is shown below:</p> <pre>Select * from TIBEDI_BATCH_TX_BIN where BININDEX in (select BININDEX from TIBEDI_BATCH_TX where STATUS like '%COMPLETED%' or '%ERROR%' and PROTOCOLNAME = '<protocol>' AND TS <= '<to date>' and TS >= '<from date>')</pre>
User IDLast Numeric Value	0
Primary Key Constraint Name	TIBEDI_BCH_TX_BIN
DDLClause	<pre>BININDEX NUMBER(18), COMPRESSED VARCHAR2(2) NULL, BINVAL BLOB NULL, CONSTRAINT TIBEDI_BCH_TX_BIN FOREIGN KEY (BININDEX) REFERENCES TIBEDI_BATCH_TX(BININDEX) ON DELETE CASCADE</pre>

Columns summary for the table TIBEDI_BATCH_TX_BIN is shown in [Table 56 on page 164](#)

Table 56 TIBEDI_BATCH_TX_BIN: Columns Summary

Name	Data Type	Constraints	Nullable	Documentation
BININDEX	number(18)		No	Foreign key reference.
COMPRESSED	varchar2(2)		Yes	The data in this column is compressed or not.
BINVAL	BLOB		Yes	The actual EDI data that has to be batched.

Appendix E **Troubleshooting**

This appendix describes some common problems and how to solve them.

Topics

- [Troubleshooting Tips, page 166](#)

Troubleshooting Tips

Senders and Receivers Falling out of Sync with Incremental Control Numbers

When the sender and receiver have all control number checking options set to incremental, it is possible for a sender to increment their control numbers and fail to send to the receiver. In this case, the receiver would have received some transactions, and then miss some, and then start receiving them again. This leads to a gap in the control numbers received which causes control number validation to fail on the receiving end.

Example:

1. Sender sends transaction 1 with control numbers 1,1,1.
2. Receiver receives transaction 1.
3. Sender increments control numbers for transaction 2: 2,1,1, but fails to actually send transaction 2.
4. Sender sends transaction 3 with control numbers 3,1,1.
5. Receiver receives transaction 3. Control number validation fails because the receiver never received transaction 2.

If this occurs, perform the following:

Execute the following SQL statement at the receiver side, assuming that the sender's domain identity is 01, Interchange Sen:

```
delete from tibedi_control_number where partner_qual='01' and partner_id='Interchange Sen' and  
direction = 'I'
```

Index

A

accessing audit log 80
 Ack file name masks 78
 acknowledgments 68
 997 and 999 X12 functional acknowledgment 91
 messages 88
 reconciliation 51
 reconciling 70
 timeouts 69
 alphanumeric control numbers and validation 49
 Analyzer 25
 archivedFileRef field
 InitiatorResponse 112
 ResponderRequest 125
 attachment 140
 audit log 79, 84
 access 80
 auditReportFileRef field
 InitiatorResponse 112
 ResponderInterchangeComplete 122
 ResponderRequest 126
 authoring and configuring EDI guidelines 34

B

batching 52

C

closure field
 InitiatorReponse 112
 InitiatorRequest 106
 ResponderRequest 117
 ResponderResponse 123

compatibility 13
 components 4
 configuring guidelines 25
 control numbers
 ControlNumberInfo object 138
 controlNumInfo fields
 InitiatorAck 114
 InitiatorReponse 111
 InitiatorRequest 107
 ResponderInterchange 120
 ResponderInterchangeComplete 121
 ResponderRequest 116
 ResponderResponse 125
 TimeoutAlert 129
 ValidationAlert 130
 management 45
 out of sync 166
 validation 45, 49
 conversion engine 27, 27
 ConversionEngineInfo object 139
 creating private processes 35
 customer support xviii

D

data formats 11
 database
 guideline 20
 description field
 InitiatorAck 114
 destinationGroup field
 ResponderRequest 116
 destinationTP field
 ResponderInterchange 119
 ResponderInterchangeComplete 121
 ResponderRequest 116

- details field
 - Error [131](#)
- directory, temporary [32](#)
- document archiving [100](#)
- duplicate field
 - InitiatorReponse [111](#)
 - ResponderRequest [117](#)

E

- EdiConvert utility [28](#)
- ediDataFile
 - Insight Report [132](#)
- ediGateway field
 - InitiatorRequest [107](#)
- edit .ecs guideline file [25](#)
- encoding field
 - InitiatorRequest [104](#)
 - ResponderResponse [123](#)
- engine
 - conversion [27](#)
- ENV_HOME [xvi](#)
- envelopeInfo field
 - InitiatorReponse [111](#)
 - ResponderRequest [116](#)
 - TimeoutAlert [129](#)
 - ValidationAlert [130](#)
- EnvelopeInfo object [136](#)
- error
 - codes [147](#)
 - fields
 - ConversionEngineInfo object [139](#)
 - errorInfoList [130](#)
 - messages [130](#)
- exchanging
 - identity information [39](#)
 - URI definitions [38](#)
- extraInfo field
 - Error [131](#)

F

- file name mask [74](#)
- file name mask examples [76](#)
- file name mask syntax [76](#)
- flow control [65](#)
- formats, data [11](#)
- FTP and FILE outbound file name masks [76](#)
- FTP inbound file name masks [74](#)

G

- general messages [127](#)
 - TimeoutAlert [127](#), [128](#)
 - ValidationAlert [129](#)
- group control number sequence check [48](#)
- group control numbers increment across inbound interchanges [47](#)
- group field
 - ControlNumberInfo object [138](#)
 - EnvelopeInfo object [136](#)
- groupHeader field
 - TransactionInfo object [138](#)
- GroupInfo object [137](#)
- groupTrailer field
 - TransactionInfo object [138](#)
- guideline
 - database [20](#)
 - files [16](#), [24](#)
 - XSDs [24](#)
- guidelines [34](#)
 - testing [25](#)

H

- hostInitiates field
 - ValidationAlert [130](#)
- hostName field
 - TimeoutAlert [128](#)
 - ValidationAlert [129](#)
- how acknowledgment timeouts work [69](#)

how acknowledgments are reconciled [70](#)
 how acknowledgments display in the audit log [90, 90](#)
 HTTP Status Codes
 200-510 [148](#)

I

ID field
 Identity object [137](#)
 identity object [137](#)
 importing a guideline file into the configuration store [25](#)
 importing a schema file into the configuration store [25](#)
 increment group control numbers across outbound
 interchanges [46](#)
 infoWarnList field
 ResponderRequest [118](#)
 initiator messages [103](#)
 Initiator Inbound Response [110, 127](#)
 Initiator Outbound Request [103](#)
 inputFile field
 InitiatorRequest [107](#)
 Insight Report
 Error [132](#)
 messages [131](#)
 instreamResultsFile
 Insight Report [132](#)
 integration [12](#)
 interchange control number seed [48](#)
 interchange control number sequence check [47](#)
 interchange field
 ControlNumberInfo object [138](#)
 EnvelopeInfo object [136](#)
 interchangeHeader field
 TransactionInfo object [138](#)
 interchangeInfo field
 ResponderInterchange [120](#)
 ResponderInterchangeComplete [121](#)
 InterchangeInfo object [137](#)
 interchangeTrailer field
 TransactionInfo object [138](#)

L

log viewer [79](#)
 log viewer options [96](#)

M

manual resend scenarios [98](#)
 message guidelines [17](#)
 message queue log [79, 95](#)
 messages [13](#)
 initiator [103](#)
 mimeType field
 InitiatorRequest [107](#)
 msgType field
 TimeoutAlert [128](#)
 ValidationAlert [129](#)

N

non-repudiation log [79, 94](#)
 notify message from initiator TIBCO BusinessConnect to
 the local private process [124](#)

O

object fields
 ControlNumberInfo [138](#)
 ConversionEngineInfo [139](#)
 EnvelopeInfo [136](#)
 GroupInfo [137](#)
 Identity [137](#)
 InterchangeInfo [137](#)
 TPName [136](#)
 TransactionInfo [138](#)

operationID field

- Error [131](#)
- InitiatorAck [113](#)
- InitiatorReponse [111](#)
- InitiatorRequest [103](#)
- Insight Report [132](#)
- ResponderInterchange [119](#)
- ResponderInterchangeComplete [121](#)
- ResponderRequest [115](#)
- ResponderResponse [123, 125](#)
- TimeoutAlert [128](#)
- ValidationAlert [130](#)

operationType field

- ResponderRequest [115](#)
- ResponderResponse [125](#)

originalSourceFile

- Insight Report [132](#)

P

perMessage field

- InitiatorRequest [104](#)
- ResponderResponse [124](#)

plugin properties [141](#)private messages and processes [13](#)private processes [13](#)private processes, creating [35](#)properties, EDI plugin [141](#)property reference [142](#)public messages [14](#)push real-time partner and guideline updates message [132](#)**Q**

qualifier field

- Identity object [137](#)

R

rawRequest field

- ResponderRequest [117](#)

receivedTime field

- ResponderInterchange [120](#)
- ResponderInterchangeComplete [121](#)
- ResponderRequest [118](#)
- ValidationAlert [130](#)

receiver field

- InitiatorAck [114](#)
- InterchangeInfo object [137](#)

receiverApp field

- GroupInfo object [137](#)

Reconcile After Timeout check box [72](#)reconciling acknowledgments [70](#)

request field

- InitiatorRequest [107](#)
- ResponderRequest [118](#)

requestFile field

- ResponderInterchange [120](#)
- ResponderInterchangeComplete [121](#)
- ResponderRequest [118](#)
- TransactionInfo object [139](#)

resend field

- ResponderRequest [117, 125](#)

resend log [97](#)responder messages [115](#)

- ResponderAck [125](#)
- ResponderInterchange [119](#)
- ResponderRequest [115](#)
- ResponderResponse [123](#)
- ResponderInterchangeComplete [120](#)

response field

- InitiatorReponse [112](#)
- ResponderResponse [123](#)

responseFile field

- ResponderResponse [124, 125](#)

Sscheduling [52](#)searching the audit log for X12 records [84](#)

- sender field
 - InterchangeInfo object [137](#)
- senderApp field
 - GroupInfo object [137](#)
- setting up trading partners [34](#)
- shared temporary directory [32](#)
- sourceFile field
 - ResponderInterchange [120](#)
 - ResponderInterchangeComplete [121](#)
 - ResponderRequest [118](#)
 - ValidationAlert [130](#)
- sourceGroup field
 - ResponderRequest [116](#)
- sourceTP field
 - ResponderInterchange [119](#)
 - ResponderInterchangeComplete [121](#)
 - ResponderRequest [116](#)
- standardID
 - Error [131](#)
 - Insight Report [132](#)
- standardID field
 - InitiatorReponse [110](#)
 - InitiatorRequest [103](#)
 - ResponderInterchange [119](#)
 - ResponderInterchangeComplete [121](#)
 - ResponderRequest [115](#)
 - ResponderResponse [123, 125](#)
 - TimeoutAlert [128](#)
 - ValidationAlert [130](#)
- state details view [89](#)
- status codes [81, 88, 147](#)
- status field
 - InitiatorAck [114](#)
- statuscode and statusmsg field reference [148](#)
- statusCode field
 - Error [131](#)
 - InitiatorReponse [111](#)
 - ResponderResponse [123, 125](#)
- statusMsg field
 - Error [131](#)
 - InitiatorReponse [111](#)
 - ResponderResponse [123, 125](#)
- summary view [84, 94, 95](#)
- support, contacting [xviii](#)

T

- TA1 X12 Interchange Acknowledgment [92](#)
- technical support [xviii](#)
- temporary directory [32](#)
- testing your configuration [26](#)
- thread count [65](#)
- TIBCO support
 - TIBCommunity [xviii](#)
 - TIBCO_HOME [xvi](#)
- timeouts
 - alert notify message [128](#)
 - completing transactions [72](#)
- tpExtraInfo field
 - ResponderInterchange [119](#)
 - ResponderInterchangeComplete [121](#)
 - ResponderRequest [116](#)
 - ValidationAlert [129](#)
- TPName field
 - TPName Object [136](#)
- tpName field
 - Error [131](#)
 - InitiatorAck [114](#)
 - InitiatorRequest [103](#)
 - TimeoutAlert [128](#)
 - ValidationAlert [129](#)
- TPName object [136](#)
- tradingpartner object [139](#)
- transaction field
 - ControlNumberInfo object [138](#)
- transactionBody field
 - TransactionInfo object [138](#)
- transactionID field
 - Error [131](#)
 - InitiatorAck [114](#)
 - InitiatorReponse [111](#)
 - InitiatorRequest [103](#)
 - Insight Report [132](#)
 - ResponderInterchange [119](#)
 - ResponderInterchangeComplete [121](#)
 - ResponderRequest [115](#)
 - ResponderResponse [123, 125](#)
 - TimeoutAlert [128](#)
 - ValidationAlert [130](#)

transactionInfo field

- InitiatorReponse 111
- ResponderRequest 116
- ValidationAlert 130

transactions

- batching and scheduling 52
- completing after timeout 72
- TransactionInfo object 138
- view details 86
- viewing in the audit log 96

troubleshooting 165

tuning

- flow control 65

txExtraInfo field

- TransactionInfo object 138

txName field

- InitiatorReponse 111
- ResponderRequest 115
- ResponderResponse 123

txnGroupingID

- txnGroupingIDfields
 - InitiatorRequest 106

X

X12 89, 89

X12 acknowledgment codes 91

XML conversion 43

U

userKey field

- TransactionInfo object 138

userKeyName field

- TransactionInfo object 138

utilities

- EdiConvert 28

V

validated field

- ResponderRequest 117

validation

- alert notify message 129
- and conversion 9
- and XML conversion 43
- error reporting 50