SeeBeyond™ eBusiness Integration Suite

HIPAA Implementation Guide

Release 4.5.2

Version 20020430153314.

Contents

Chapter 3

Chapter 4

Chapter 5

Appendix A

Chapter 1

Introduction

This chapter introduces you to the HIPAA Implementation Guide.

HIPAA is a mandate that was developed specifically for the healthcare industry. For transactions related to health care, HIPAA uses a customization of X12. For pharmaceutical transactions, the HIPAA standard uses NCPDP (National Council for Prescription Drug Programs) transactions.

This book includes an overview of HIPAA, and then specific information relating to the installation and contents of SeeBeyond's HIPAA implementations.

1.1 Introduction to HIPAA

The Health Insurance Portability & Accountability Act of 1996 (HIPAA) amends the Internal Revenue Service Code of 1986. It requires:

- Improved efficiency in healthcare delivery by standardizing electronic data interchange
- Protection of confidentiality and security of health data through setting and enforcing standards.

More specifically, HIPAA calls for:

- Standardization of electronic patient health, administrative and financial data
- Unique health identifiers for individuals, employers, health plans and health care providers
- Security standards protecting the confidentiality and integrity of "individually identifiable health information," past, present or future

1.2 Intended Reader

The reader of this guide is presumed to be a developer or system administrator with responsibility for developing the e*Gate™ system or the SeeBeyond™ eBusiness Integration Suite, to be thoroughly familiar with Windows 2000 and Windows NT operations and administration, and also with Microsoft Windows graphical user interfaces.

1.3 Supporting Documents

The following SeeBeyond documents provide additional information that might prove useful to you.

- *HIPAA ETD Library User's Guide*
- *X12 ETD Library User's Guide*
- *NCPDP ETD Library User's Guide*
- *e*Gate Integrator Installation Guide*
- *e*Xchange Partner Manager Installation Guide*
- *e*Index User's Guide*
- *e*Index Administrator User's Guide*
- *e*Index Security User's Guide*

Chapter 2

HIPAA Overview

This chapter provides an overview of HIPAA, including general information, a list of the specific transactions that comprise the HIPAA standard, and the structure of HIPAA envelopes, data elements, and syntax.

2.1 Introduction to HIPAA

The following sections provide an introduction to HIPAA.

2.1.1 What Is HIPAA?

HIPAA is an acronym for the Health Insurance Portability and Accountability Act of 1996. This Act is designed to protect patients. Among other things, it makes specifications affecting standards of treatment and privacy rights. It provides a number of standardized transactions that can be used for such things as a healthcare eligibility inquiry or a healthcare claim. HIPAA legislates that all of the healthcare industry will be on the same implementation timetable. All institutions doing electronic healthcare insurance transactions must implement these standardized transactions by October 2003.

HIPAA has three primary goals:

- define standards for electronic transactions and code sets used for financial and clinical electronic data interchange (EDI).
- establish unique identifiers for the three participants in the provision of healthcare services: providers, payers, and employers.
- mandate security and privacy standards for the protection of individually identifiable healthcare information.

HIPAA regulations affect many organizations dealing with the medical industry, such as:

- providers
- health plans
- employers

For provider systems, HIPAA does not mandate they perform EDI and therefore many of the standards do not apply. However, if a provider elects to perform EDI, then their

EDI transactions are required to be in compliance with all of the HIPAA transaction requirements.

The impact of HIPAA on health plans is potentially far greater than the impact on provider systems. Where providers have the option to perform EDI, HIPAA requires health plans to support the nine standard EDI transactions.

2.1.2 HIPAA Goals

Electronic Health Transactions Standards

Historically, health providers and plans used many different electronic formats. Implementing a national standard means that everyone uses one format, thereby simplifying and improving transaction efficiency. HIPAA defines standards for nine healthcare transactions, and mandates that all providers, health plans, and employers performing EDI comply with the standards. The HIPAA transactions are:

- eligibility for a health plan
- claims or equivalent encounter information
- payment and remittance advice
- coordination of benefits
- health claims status
- referral certification and authorization
- first report of injury
- enrollment and disenrollment in a health plan
- health plan premium payments
- pending transaction - health claims attachments

For transactions relating to such things as health care claims, the HIPAA standard uses a range of customized X12 transactions as listed above. For transactions relating to prescriptions, HIPAA uses NCPDP (National Council for Prescription Drug Programs) transactions. For information on HIPAA X12, see **"HIPAA X12" on page 12**, and for information on NCPDP, see **"NCPDP" on page 15**.

Health organization must also adopt standards for the coding of information within the individual transactions. For example, coding systems that describe diseases, injuries, and other health problems, as well as their causes, symptoms, and actions taken must be uniform. HIPAA also establishes national standards for these code sets based on currently available standards (for example, ICD9, CPT4, and so on).

Unique Identifiers

As well as meeting the need for standard encoding of information within the transactions, HIPAA also establishes the requirement to uniquely identify the participants involved in the provision and payment for healthcare services. These participants include the provider, the payer (health plan), and the employer.

Security and Electronic Signatures

The security standards provide a level of protection for all health information that is housed or transmitted electronically and that pertains to an individual. Organizations that use electronic signatures also have to meet a standard ensuring message integrity, user authentication, and non-repudiation.

The security standard mandates safeguards for physical storage and maintenance, transmission, and access to individual health information. It applies not only to the transactions adopted under HIPAA, but to all individual health information that is maintained or transmitted.

The security standard does not require specific technologies to be used; solutions vary from business to business, depending on the needs and technologies in place.

No transactions adopted under HIPAA currently require an electronic signature.

Privacy and Confidentiality

In general, privacy is about who has the right to access personally identifiable health information. This covers all individually identifiable health information regardless of whether the information is, or has been, in electronic form.

The privacy standards:

- limit to non-consensual use and release of private health information
- give patients the right to access their medical records and to know who else has accessed them
- restrict most disclosure of health information to the minimum needed for the intended purpose
- establish new criminal and civil sanctions for improper use or disclosure
- establish new requirements for access to records by researchers and others

2.1.3 Trading Partner Agreements

Although the regulations mandated by HIPAA are very strict and specific, it is still important to have trading partner agreements for individual trading relationships.

Following the HIPAA standard ensures that transactions comply with the regulations mandated by the government. HIPAA requirements are completely described in the HIPAA implementation guide for each transaction, and must not be modified by a trading partner.

However, there is room for negotiation in terms of the specific processing of the transactions in each trading partner's individual system. The specifics might vary between sites. The trading partner agreement is a useful repository for this type of site-specific information.

There are three levels of information that guide the final format of a specific transaction. These three levels are:

- The HIPAA standard

HIPAA publishes a standard structure for each HIPAA transaction.

- Industry-specific Implementation Guides

  Specific industries, including healthcare, publish Implementation Guides customized for that industry. Normally, these are provided as recommendations only. However, in the case of HIPAA, it is extremely important to follow these guidelines since HIPAA regulations are law.

- Trading Partner Agreements

  It is normal for trading partners to have individual agreements that supplement the standard guides. The specific processing of the transactions in each trading partner's individual system might vary between sites. Because of this, additional documentation that provides information about the differences is helpful to the site's trading partners and simplifies implementation. For example, while a certain code might be valid in an implementation guide, a specific trading partner might not use that code in transactions. It would be important to include that information in a trading partner agreement.

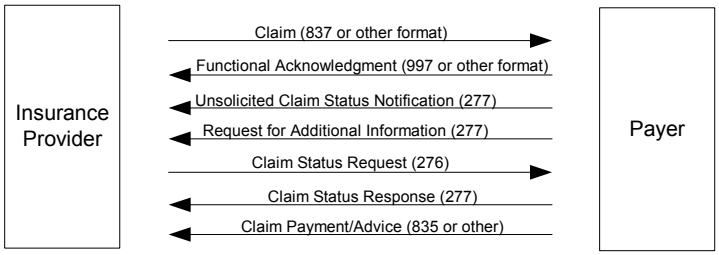
2.2 HIPAA X12

The following section provides an introduction to HIPAA X12, including information about HIPAA X12 transactions and message structures.

2.2.1 Sample Scenario

An example of a HIPAA X12 transaction exchange between a healthcare provider and a payer is shown in Figure 1.

Figure 1 Sample HIPAA Transaction Exchange



2.2.2 Batch and Real Time Transactions

The HIPAA standard supports the sending and receiving of messages in both batch and real time (interactive) modes.

Batch

In batch mode, transactions are grouped together and multiple transactions are sent in a single message. The batch can either go directly to the receiver or via a clearing house. The connection does not remain open while the receiver processes the messages. If there is an expected response transaction (for example, a 271 in response to a 270) the receiver creates the response transaction offline and then sends it.

Real Time

If a transaction is processed in real time, it is sent individually. Transactions that require an immediate response are normally sent in real time. In real time mode, the sender sends the request transaction, either directly or through a clearing house, and the connection is kept open while the receiver processes the transaction and returns a response transaction. Response times are typically no more than one minute, and often less.

In real time mode, the receiver must send a response; either the expected response transaction, such as a 271 in response to a 270, or a standard acknowledgment such as the 997.

2.2.3 Data Overview

HIPAA X12 transactions all use the standard components of the X12 standard, covered in **Appendix A**, **"ASC X12 Overview" on page 62**.

Specifically, it uses the following elements:

- Segments
- Data elements
- Looping structures

In addition, it mandates consistent use of these things across all HIPAA implementation guides.

The X12 portion of the HIPAA ETD Library provides Event Type Definitions for all nine standard X12 transactions that have been adopted by HIPAA, as listed in Table 1.

These transactions are based on the October 1997 X12 standard; that is, Version 4, Release 1, Sub-release 0 (004010) (version 4010).

Table 1 HIPAA X12 Transactions

Number	Name
270	Eligibility Coverage or Benefit Inquiry
271	Eligibility Coverage or Benefit Information
276	Health Care Claim Status Request
277	Health Care Claim Status Notification
278	Two versions: Health Care Services Review Information and Request for Review/Response to Request

Table 1 HIPAA X12 Transactions

Number	Name
820	Payment Order Remittance Advice
834	Benefit Enrollment and Maintenance
835	Health Care Claim Payment Advice
837	Health Care Claim (three versions: Professional, Dental, and Institutional)

The NCPDP portion of the HIPAA ETD Library provides request and response transactions for all the HIPAA-approved NCPDP transaction codes, as listed in Table 2.

Table 2 NCPDP Transaction Codes

Code	Transaction Name
E1	Eligibility Verification
B1	Billing
B2	Reversal
B3	Rebill
P1	Prior Authorization Request and Billing
P2	Prior Authorization Reversal
P3	Prior Authorization Inquiry
P4	Prior Authorization Request Only
N1	Information Reporting
N2	Information Reporting Reversal
N3	Information Reporting Rebill
C1	Controlled Substance Reporting
C2	Controlled Substance Reporting Reversal
C3	Controlled Substance Reporting Rebill

2.2.4 Acknowledgment

The HIPAA X12 transactions either have specific designated response transactions, or use the standard 997 Functional Acknowledgment.

The 997 is used by the following transactions:

- 837 (sent by the payer to acknowledge claim receipt)
- 277 (sent by the provider to acknowledge receipt of a Health Care Payer Unsolicited Claim Status request)
- 277 (sent by the provider to acknowledge receipt of a Health Care Claim Request for Additional Information)
- 835 (sent by the provider to acknowledge receipt of a Health Care Claim Payment/Advice notification)

2.3 NCPDP

The following section provides an introduction to NCPDP, including information about NCPDP transactions and message structures.

2.3.1 What Is NCPDP?

NCPDP (an acronym for the National Council for Prescription Drug Programs) is an organization, accredited by ANSI, that is tasked with standards development for the pharmaceutical industry.

The mission of NCPDP is twofold:

- To create and promote standards for data interchange in pharmaceutical services (including electronic data interchange)
- To provide educational information and resources to members

In following the above, NCPDP hopes to enhance the quality of healthcare by creating, and encouraging the use of, a high-quality data interchange standard.

2.3.2 History

Pharmacies started moving towards computerization in the late 1970s. By 1977, standardization of forms was seen as a need and NCPDP was formed to meet that need. The first NCPDP standardized form was released in 1978. By 1987, electronic claims were introduced. In 1988, version 1.0 of the NCPDP Telecommunications Standard was released. Since then, the standard has continued to be developed.

2.3.3 What Is the NCPDP Telecommunications Standard?

The NCPDP Telecommunications Standard (Telecom) is a data transmission standard specifically designed for the communication of prescription information between pharmacies and payers. It was developed to provide a consistent standard for pharmaceutical drug claims. This standard defines the structure for prescription claim transactions between providers (for example, pharmacies or doctors) and claims adjudicators. It provides for communications in both directions.

The HIPAA (Health Insurance Portability and Accountability Act of 1996) standard for electronic health care transactions and code sets adopts the following NCPDP standards for pharmacy claims:

- NCPDP Telecommunication Standard Format, Version 5.1
- NCPDP Batch Standard, Version 1 Release 0 (1.0)

Note: *Currently, the HIPAA standard uses NCPDP Batch Standard Version 1.0. However NCPDP has asked DSMO (Designated Standards Maintenance Organization) for revision to support Batch Standard Version 1.1 for usage with Telecommunication Standard Version 5.1. When the NPRM (Notices of Proposed*

Rulemaking) is issued for the correction, the HIPAA standard will use Batch Standard Version 1.1.

Health plans, health care clearinghouses, and health care providers who use electronic transactions are required to use these standards after October 2003.

2.3.4 Components of an NCPDP Envelope

NCPDP messages are all ASCII text, with the exception of the delimiters which are hexadecimal.

Structure of a Request Transaction

An NCPDP Business Request Transaction has the following main parts:

- An electronic envelope, including such items as sender ID, receiver ID, message type, password, and date/time.
- A prescriber section, including such items as prescriber identifier (for example, State License), prescriber name, business name, business address, and specialty code.
- A pharmacy section, including such items as NCPDP provider identifying code, pharmacy name, pharmacist name, pharmacy address, and pharmacy phone number.
- A patient section, including such items as patient name, date of birth, gender, address, and the pharmacy or prescriber's internal ID code for the patient.

Structure of a Response Transaction

An NCPDP Response Transaction includes:

- Electronic envelope
- Response status, which can be any one of the following:
  - An acknowledgment of receipt of the transaction
  - A "paired" response transaction (this might approve the request, deny it, or approve it but with changes)
  - An error acknowledgment

2.3.5 Batching in NCPDP

NCPDP supports batching of transactions.

An NCPDP batch file is comprised of three sections:

- Transaction header (one per batch)
- Data (one or many, to a maximum of 9,999,999,997) Each contains a Transaction Reference Number to uniquely identify the transaction within the file.
- Transaction trailer (one per batch)

2.3.6 Acknowledgment Types

The transactions defined within NCPDP are of two types: request transactions, and response transactions. There are no discrete acknowledgment transactions.

However, a "captured" response (one of the several types of response transactions) can be used when information transactions are sent and require nothing more than acknowledgment of their receipt at the processor or endpoint.

2.3.7 Transaction Codes

NCPDP uses Transaction Codes to indicate the type of transaction.

A list of NCPDP Transaction Codes is provided in Table 2.

Table 3 NCPDP Transaction Codes

Code	Transaction Name
E1	Eligibility Verification
B1	Billing
B2	Reversal
B3	Rebill
P1	Prior Authorization Request and Billing
P2	Prior Authorization Reversal
P3	Prior Authorization Inquiry
P4	Prior Authorization Request Only
N1	Information Reporting
N2	Information Reporting Reversal
N3	Information Reporting Rebill
C1	Controlled Substance Reporting
C2	Controlled Substance Reporting Reversal
C3	Controlled Substance Reporting Rebill

2.4 Additional Information

For more information on HIPAA, visit the following Web sites:

- **http://www.hcfa.gov/HIPAA/HIPAAHM.HTM**
- **http://www.hipaa-dsmo.org**
- **http://www.wedi.org/**
- **http://www.ehnac.org/**
- **http://aspe.os.dhhs.gov/admnsimp/**

For more information on NCPDP, visit the official NCPDP Web site at this address:

- **http://www.ncpdp.org/**

Note: *This information is correct at the time of going to press; however, SeeBeyond has no control over these sites. If you find the link is are no longer correct, use a search engine to search for* ***HIPAA*** *or* ***NCPDP****.*

Chapter 3

The SeeBeyond Solution

This chapter provides an overview of SeeBeyond's solution for HIPAA implementations.

3.1 Introduction

The SeeBeyond Business Integration Suite supports the translations and field mapping features needed to comply with nationally mandated code sets while preserving local autonomy. It also includes the pre-built message structures for all HIPAA transactions, and the ability to map proprietary, internal messaging formats to the appropriate HIPAA transactions.

3.1.1 e*Xchange Partner Manager

e*Xchange Partner Manager allows organizations to use technology for business-to-business (B2B) and business-to-consumer (B2C) e-commerce. In addition to the standard e*Xchange functionality, e*Xchange provides pre-built Monk validation rules for the nine standard X12 transactions for HIPAA that are EHNAC compliant, as well as HIPAA-compliant security for transmission over public networks, if desired.

3.1.2 e*Gate Integrator

e*Gate Integrator can be used used without e*Xchange Partner Manager to transform data from other formats to the a standard X12 format for HIPAA. It also provides connectivity with, and between, the diverse systems and applications that participate in the HIPAA transactions.

3.1.3 e*Index Global Identifier

e*Index Global Identifier provides the ability to maintain internal numbering for providers, health plans, and employers, and simply cross-index these internal numbers to the nationally assigned identifiers for external communication. This may become useful when introducing the HIPAA requirement of unique identifiers.

3.2 e*Xchange Partner Manager

e*Xchange Partner Manager provides functionality to receive, process, and routes inbound and outbound messages in batch, fast batch, and interactive transmission modes.

For HIPAA, e*Xchange provides pre-built validation rules for the nine standard transactions, as well as HIPAA compliant security for transmission over public networks, if desired. Specifically, e*Xchange provides the following:

- Validates messages based on Event Type Definitions and Collaboration scripts that conform to HIPAA regulations. These validations are EHNAC compliant.
- e*Xchange automatically generates and reconciles acknowledgments; providing the acknowledgment handling required by HIPAA.
- Stores trading partner information, messages, acknowledgments, and errors in a database. HIPAA requires that seven years of patient data is stored. This is handled by the e*Xchange database; and the e*Xchange Repository Manager allows management/archiving of data.
- e*Xchange allows users to view messages and supports security of data access via user ID/password verification (e*Xchange Web interface only; if HIPAA compliance is a requirement, you cannot use Client for Windows).
- Audit of who is viewing the data is also a HIPAA mandate; SeeBeyond support that via the Web interface Message Tracking audit feature.
- HIPAA requires that they are able to track transactions per trading partner -- this is supported in Message Tracking.

eSecurity Manager offers the following additional functionality that may be desired by HIPAA:

- Exchange content integrity
- Origin authentication via digital signatures
- Non-repudiation of transmission and receipt

3.2.1 e*Xchange Files for HIPAA Transactions

e*Xchange includes Monk Collaboration Rules (**.tsc**) and Monk Event Type Definition (**.ssc**) files for many of the standard HIPAA transactions. These files are in a format suitable for use with e*Xchange; they include only the transaction portion of the message, without the GS/GE and ISA/IEA enveloping.

There are HIPAA X12 transaction files for the following HIPAA Implementation Guide releases:

- May 1999
- May 2000

The files that match the HIPAA standard have “hipaa” appended to the file name. Examples are shown below.

- Collaboration Rules file for an X12 270, Eligibility Coverage Inquiry, May 1999 release: **X12_270EligibilityCoverageorBenefitInquiry_004010X092_hipaa.tsc**.
- Collaboration Rules file for an X12 270, Eligibility Coverage Inquiry, May 2000 release: **X12_270EligibilityCoverageorBenefitInquiry_004010X092_00_hipaa.tsc**.

In some cases, there are different versions of the HIPAA files. For example, for the May 1999 release, there are three versions for 837, as follows:

- X12_837HealthCareClaim004010X096_hipaa_q1 (professional)
- X12_837HealthCareClaim004010X096_hipaa_q2 (dental)
- X12_837HealthCareClaim004010X096_hipaa_q3 (institutional)

HIPAA files are automatically installed in the following location:

- **\eGate\server\registry\repository\default\monk_scripts\eXchange\HIPAA**

Note: *For HIPAA compliance, you must use the e*Xchange Web Interface to track access to data viewed through the Message Tracking facility, and limit e*Xchange Client for Windows access to Administrators only. e*Xchange Client for Windows does not audit data access.*

3.2.2 Complete HIPAA Transaction ETDs

In addition to the standard e*Xchange format files, installation also includes a version of the HIPAA ETD files which include the GS/GE and ISA/IEA enveloping. These are suitable for use outside e*Xchange when a complete Event structure is required; for example, when using e*Gate to translate from X12 to a business application's proprietary data format.

These files are stored in the same location as the standard e*Xchange format HIPAA files. The file names have "_xlate" (for May 1999 files) or "_xlat" (for May 2000 files) appended to the file name to indicate that these are the translation files and include the interchange control and functional group header and footer. For a complete list of files, see **"HIPAA e*Xchange Files for e*Gate" on page 78**.

Note: *These files use dynamic delimiters, and can only be used in translating from X12 to a proprietary format.*

3.3 e*Gate Integrator

The HIPAA ETD Library includes the pre-built Java message structures for all HIPAA transactions, and the ability to map proprietary, internal messaging formats to the appropriate HIPAA transactions. For more information on the HIPAA ETD Library, refer to the HIPAA ETD Library User's Guide.

Note: *Although you can use e*Gate to create EDI messages that conform to HIPAA standards, you also need to ensure that other HIPAA standards are also met; for*

*example, privacy and security. e*Xchange Partner Manager provides a more complete HIPAA solution.*

All the HIPAA X12 ETDs accept either standard ANSI X12 format or XML format as input. By default, output is ANSI. However, you can optionally define that the output is XML. Although the XML format does not meet the HIPAA requirements for EDI, this format is useful when displaying the data in a Web browser.

3.3.1 e*Gate Files for HIPAA Transactions

The X12 portion of the HIPAA ETD Library provides Java Event Type Definitions (**.xsc** and **.jar** files) for all nine standard X12 transactions that have been adopted by HIPAA. These ETDs are stored in the following locations:

<eGate>\server\registry\repository\default\etd\templates\Hipaa_1999

<eGate>\server\registry\repository\default\etd\templates\Hipaa_2000

These transactions are based on the October 1997 X12 standard; that is, Version 4, Release 1, Sub-release 0 (004010) (version 4010).

For a list of files, see **"e*Gate Files for HIPAA Transactions" on page 79**.

The NCPDP portion of the HIPAA ETD Library provides request and response transactions for all the HIPAA-approved NCPDP transaction codes. These ETDs are stored in:

<eGate>\server\registry\repository\default\etd\templates\NCPDP

For a list of NCPDP files, see **"e*Gate Files for HIPAA Transactions" on page 79**.

Chapter 4

e*Xchange Implementation

This chapter discusses the steps involved to create an e*Xchange implementation that transfers HIPAA X12 data.

4.1 Overview

An e*Xchange implementation makes use of the features designed to add and remove the EDI enveloping information for messages exchanged between trading partners.

In an e*Xchange implementation, use the e*Xchange Web Interface to set up trading partner information, and the e*Gate Enterprise Manager GUI to add user-defined e*Gate components to provide connectivity to the business application or trading partner. Once this is done, the pre-configured e*Xchange e*Gate Schema components handle enveloping and de-enveloping Events as they travel through the e*Xchange system.

The major steps for an e*Xchange implementation are as follows:

1. Create the trading partner profiles.
2. Configure the user-defined e*Ways that will connect the business application to e*Xchange and exchange messages with the trading partner.
3. Configure the e*Xchange e*Way.
4. Run and test the scenario.

4.1.1 Case Study: Sending a Health Care Claim

The case study discussed in this chapter illustrates one possible implementation of sending a health care claim to a trading partner.

In this example, a Health Care Claim (837) is sent to an external trading partner, the insurance provider. The enveloping is automatically added to the message by e*Xchange based on trading partner information retrieved from the e*Xchange database, and then it is sent to the external system. An acknowledgment message (997) is immediately returned by the insurance provider. Then the Health Care Payment (835) is sent from the insurance provider trading partner, and an acknowledgment message (997) is returned to complete the cycle. Figure 2 shows the message flow.

Figure 2 HIPAA Message Flow

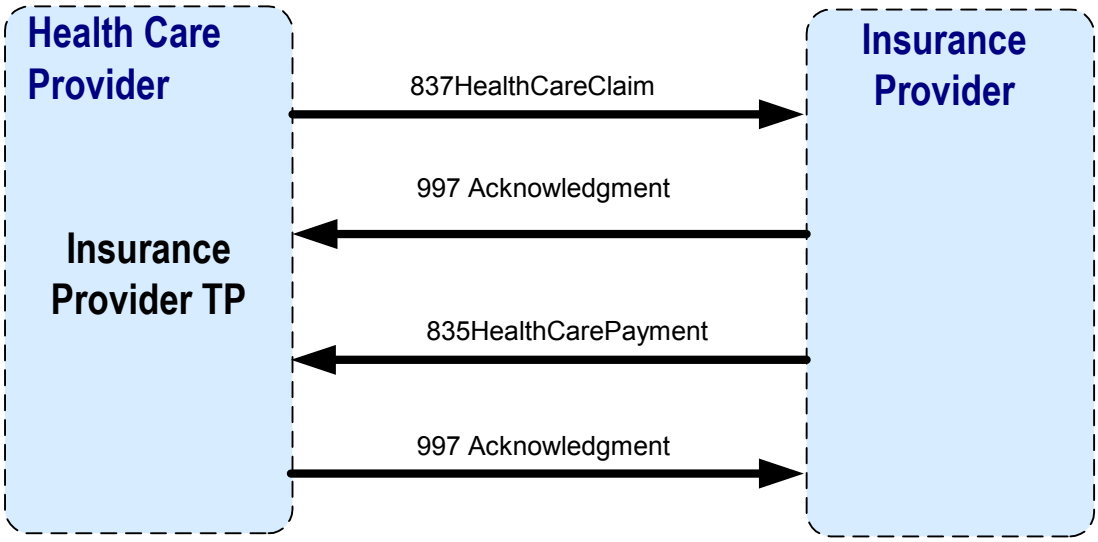


Figure 3 e*Xchange Scenario Data Flow

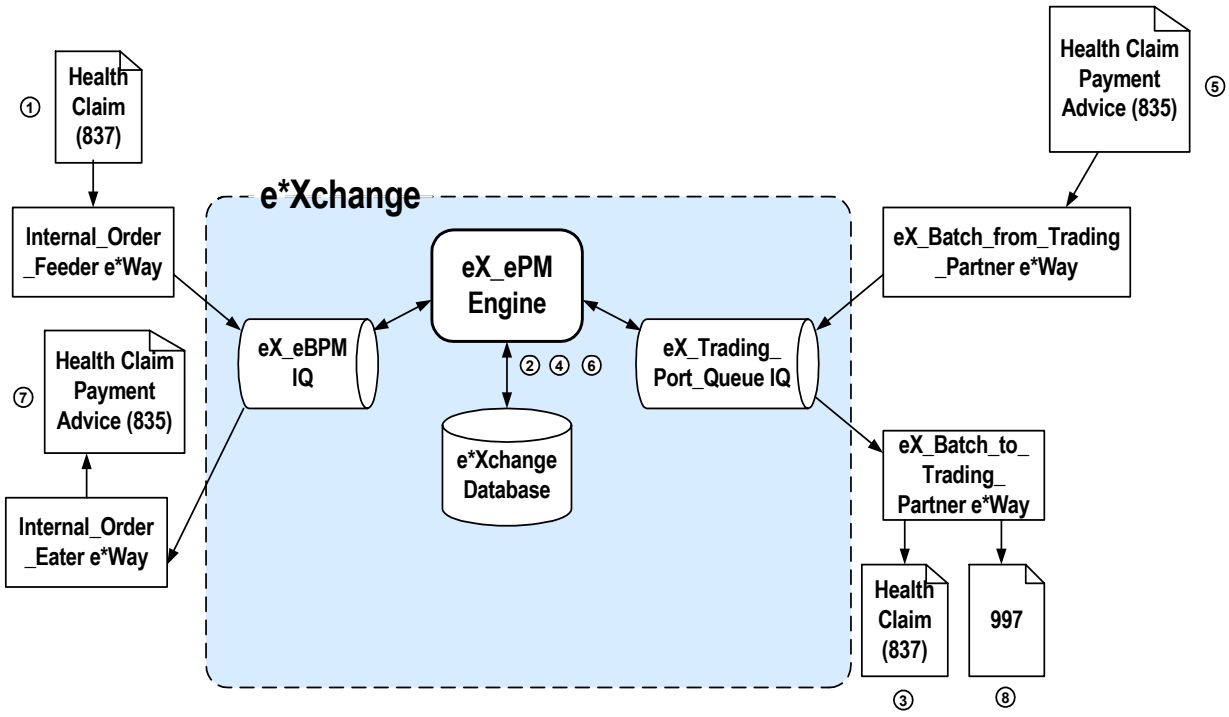


Figure 3 data flow description

1. The **Internal_Order_Feeder** e*Way picks up the health care claim message (837) and publishes it to the **eX_eBPM** IQ.
2. The e*Xchange engine picks it up from the IQ, validates it, saves it to the database, and publishes the message to the **eX_Trading_Port_Queue** IQ.
3. The **eX_Batch_to_Trading_Partner** e*Way sends out the message to the trading partner.
4. The **eX_Batch_from_Trading_Partner** e*Way sends the health care payment advice message (835) to the **eX_Trading_Port_Queue** IQ.
5. The e*Xchange engine picks it up from the IQ, validates it, saves it to the database, and publishes two messages:
   - an acknowledgment (997) to the **eX_Trading_Port_Queue** IQ.
   - the health care payment advice (835) to the **eX_eBPM** IQ.
6. The **Internal_Order_Eater** e*Way picks up the message from the **eX_eBPM** IQ and sends it to the internal system.

4.2 Create the Trading Partner Profiles

Trading partner profiles in e*Xchange act as repositories for the information necessary to send EDI messages back and forth between entities. They contain all of the information needed to properly envelope an Event and forward it to its correct destination.

Refer to the *e*Xchange Partner Manager User's Guide* for detailed assistance with the process of creating trading partner profiles.

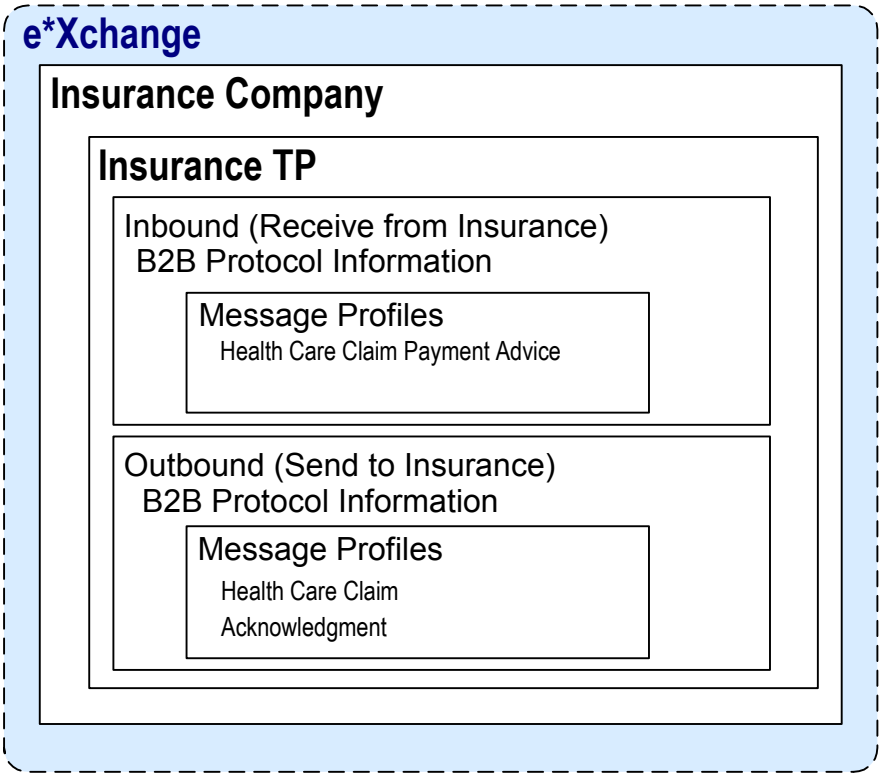
Trading Partner Information Hierarchy

e*Xchange stores trading partner information at various levels. The process of creating a trading partner profile proceeds from the most general inclusive level, that of a company with which you do business, to the most specific information regarding a message that you wish to send (the message profile).

Figure 4 shows an overview of the components that you need to create for this example, including:

- Company
- Trading Partner
- B2B Protocol Information
- Message Profiles

Figure 4 Insurance Overview



To configure the Insurance trading partner profile you must follow the steps listed below:

Step 1: Create the Company

1. Log in to the e*Xchange Web interface.
2. From the **Main** page, click **Profile Management**.
3. From the **Company** page, click **New**.
4. In the **Company - adding** page, enter the **Company** name, “Insurance”.
5. Click **Next**.

   This saves your changes and returns to the **Company** page.

Note: *The security information is automatically configured for the current user.*

Step 2: Create the Trading Partner

1. From the **Company** page, ensure that "Insurance" trading partner is selected, and then click **Continue: Trading Partner**.
2. From the **Trading Partner** page, click **New** to access the **Trading Partner - adding** page.
3. Enter the **Trading Partner Name**, "Insurance".
4. Click **Next**.

   This saves your changes and returns to the **Trading Partner** page.

The required security information defaults from the company level.

Step 3: Set up the Inbound B2B Protocol Information

To set up the inbound B2B Protocol Information

1. From the **Trading Partner** page, ensure that the "Insurance" is selected, and click **Continue: B2B Protocol**.
2. From the **B2B Protocol** page, click **New** to access the **B2B Protocol - adding** page.
3. Enter the information listed in Table 4.

   In an actual implementation, your local administrator can provide you with the B2B Protocol information. For an explanation of the B2B Protocol parameters, see the *e*Xchange Partner Manager User's Guide*.

Table 4 B2B Protocol Information

Parameter	Value
eBusiness Protocol	X12
Version	4010
Direction	Inbound

4. Click **Next** to save your changes and access the **General** section.
5. Enter the information listed in Table 5.

Table 5 B2B Protocol Information, General Page

Parameter	Value
Logical Name	Insurance
Status	Active
Communication Protocol	FTP(BATCH)

6. Click **Next** to save your changes and access the **Transport Component** section.
7. In the **File Name** window, enter **<egate>\data\hipaa\TP\input*.dat**.

8 Click **Next** to access the **Message Security** section.

9 No changes are required. Click **Finish** to save the information and return to the **B2B Protocol** page.

Step 4: Create the Inbound Message Profiles

For the purposes of this scenario, you must set up the following inbound message profile:

- Health Claim Payment Advice (X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa)

To set up the X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa Order inbound message profile

1 From the **B2B Protocol** page, click **Continue: Message Profile**.

2 From the **Message Profile** page, click the **New** button to access the **Message Profile - adding** page.

3 Enter the information listed in Table 6.

Note: *This table only lists the attributes required to make this scenario work.*

Table 6 General (X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa)

Name	Value
Name	X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa
Validation Collaboration	X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa
Transfer Mode	Interactive

4 Click **Next** to access the **Interchange Control Envelope** section. Enter the information listed in Table 7.

Table 7 Interchange Control Envelope (X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa)

Name	Value
ISA05 Interchange Sender Identification Qualifier	01
ISA06 Interchange Sender Identifier	6264712000
ISA07 Interchange Receiver Identification Qualifier	01
ISA08 Interchange Receiver Identifier	6264716000
ISA12 Interchange Version Number	00401

5 Click **Next** to access the **Functional Group Envelope** section. Enter the information listed in Table 8.

Note: *This table only lists the attributes required to make this scenario work.*

Table 8 Functional Group Envelope (X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa)

Name	Value
GS01 Functional Identification Code	HP
GS02 Application Sender Code	6264712000
GS03 Application Receiver Code	6264716000
GS06 Group Control Number	1
GS07 RESP Agency Code	X
GS08 Version/Release/Industry Identification Code	004010X091

6 Click **Next** to access the **Transaction Set Envelope** section. Enter the information listed in Table 9.

Table 9 Transaction Set Envelope (X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa)

Name	Value
ST01 Transaction Set Identification Code	835
ST02 TS Control Number	1

7 Click **Next** to access the **Return Messages** section.

8 No changes are required. Click **Finish** to save the information and return to the **Message Profile** page.

Step 5: Set Up Outbound B2B Protocol Information

To set up the outbound B2B Protocol information

As a shortcut, you can copy the Outbound B2B Protocol Information as a model for the Inbound B2B Protocol Information.

1 On the **B2B Protocol** page, select the X12-4010-Inbound protocol that you created in **"To set up the inbound B2B Protocol Information" on page 28**.

2 Click **Copy**.

The **Copy Type** page appears.

3 Clear the **Include Sub-components** check box and then click **OK**.

The **B2B Protocol - copying** page appears.

4 In the **Direction** field, ensure that **Outbound** is selected.

5 Click **Next**.

The **B2B Protocol - copying**, **General** page appears.

6 No changes are needed: click **Next** to accept the values and access the **Transport Component** page.

7 In the **File Name** window, enter **<egate>\data\TP\output\output%#.dat**.

8 Click **Next** to accept the values and access the **Message Security** page.

9 No changes are required. Click **Finish** to save the information and return to the **B2B Protocol** page.

Step 6: Create the Outbound Message Profiles

For the purposes of this scenario, you must set up the following outbound message profiles:

- Health Care Claim Message (X12_837HealthCareClaim_004010X098_00_hipaa_q1)
- Acknowledgment (X12-4010-997)

To set up the X12_837HealthCareClaim_004010X098_00_hipaa_q1 Order inbound message profile

1 From the **B2B Protocol** page, click **Continue: Message Profile**.

2 From the **Message Profile** page, click the **New** button to access the **Message Profile - adding** page.

3 Enter the information listed in Table 10.

Note: *This table only lists the attributes required to make this scenario work.*

Table 10 General (X12_837HealthCareClaim_004010X098_hipaa_q1)

Name	Value
Name	X12_837HealthCareClaim_004010X098_00_hipaa_q1
Validation Collaboration	X12_837HealthCareClaim_004010X098_00_hipaa_q1
Transfer Mode	Interactive

4 Click **Next** to access the **Interchange Control Envelope** section. Enter the information listed in Table 11.

Table 11 Interchange Control Envelope (X12_837HealthCareClaim_004010X098_00_hipaa_q1)

Name	Value
ISA06 Interchange Sender Identifier	6264716000
ISA08 Interchange Receiver Identifier	6264712000
ISA11 IC Standards Identifier	U
ISA13 IC Control Number	36
ISA15 Test Indicator	T

5 Click **Next** to access the **Functional Group Envelope** section. Enter the information listed in Table 12.

Note: *This table only lists the attributes required to make this scenario work.*

Table 12 Functional Group Envelope (X12_837HealthCareClaim_004010X098_00_hipaa_q1)

Name	Value
GS01 Functional Identification Code	HC
GS02 Application Sender Code	6264716000
GS03 Application Receiver Code	6264712000
GS06 Group Control Number	1209
GS08 Version/Release/Industry Identification Code	004010X098

6 Click **Next** to access the **Transaction Set Envelope** section. Enter the information listed in Table 13.

Table 13 Transaction Set Envelope (X12_837HealthCareClaim_004010X098_hipaa_q1)

Name	Value
ST01 Transaction Set Identification Code	837
ST02 TS Control Number	1

7 Click **Next** to access the **Return Messages** section.

8 Select the return message (select the **Include** check box), and enter the values, as shown in Table 14.

Table 14 Return Message Values: Outbound

Name	Response Time	Period	# Retries
X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa	10	Minutes	1

9 Click **Finish** to save the information and return to the **Message Profile** page.

To set up the X12-4010-997 inbound inner envelope

1 From the **Message Profile** page, click the **New** button to access the **Message Profile - adding** page.

2 Enter the information listed in Table 15.

Note: *This table only lists the attributes required to make this scenario work.*

Table 15 General (X12-4010-997)

Name	Value
Name	X12-4010-997
Transfer Mode	Interactive

3 Click **Next** to access the **Interchange Control Envelope** section. Enter the information listed in Table 16.

Table 16 Interchange Control Envelope (X12-4010-997)

Name	Value
ISA06 Interchange Sender Identifier	6264716000
ISA08 Interchange Receiver Identifier	6264712000
ISA11 IC Standards Identifier	U
ISA13 IC Control Number	36
ISA15 Test Indicator	T

4 Click **Next** to access the **Functional Group Envelope** section. Enter the information listed in Table 17.

Note: *This table only lists the attributes required to make this scenario work.*

Table 17 Functional Group Envelope (X12-4010-997)

Name	Value
GS01 Functional Identification Code	FA
GS02 Application Sender Code	6264716000
GS03 Application Receiver Code	6264712000
GS06 Group Control Number	1209
GS08 Version/Release/Industry Identification Code	004010X091

5 Click **Next** to access the **Transaction Set Envelope** section. Enter the information listed in Table 13.

Table 18 Transaction Set Envelope (X12-4010-997)

Name	Value
ST01 Transaction Set Identification Code	997
ST02 TS Control Number	36

6 Click **Next** to access the **Return Messages** section.

7 No changes are required. Click **Finish** to save the information and return to the **Message Profile** page.

Step 7: Configure Return Messages for Inbound

To set up the Return Message Profile values for Inbound

Once you have set up inbound and outbound message profiles, you can specify return messages.

1 From the **B2B Profile** page, select **X12-4010-Inbound**.

2 Click **Continue: Message Profile**.

3 From the **Message Profile** page, select **X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa** from the drop-down list.

4 Click the **Return Messages** link to access the **Return Messages** section.

5 Click **Edit**.

6 Select the return messages (select the check boxes), and enter the values, as shown in Table 19.

Table 19 Return Message Values: Inbound

Name	Response Time	Period	# Retries
X12-4010-997	3	Minutes	1

7 Click **Apply** to save the information and return to the **Message Profile** page.

8 Click **OK**.

4.3 Clone the eXSchema

The supplied schema named eXSchema contains the components required to run e*Xchange. Make a copy of this Schema and then configure the copy for this implementation.

To make a copy of eXSchema

1 Open eXSchema in the e*Gate Enterprise Manager GUI.

2 Export eXSchema.

3 Create a new Schema named **HealthClaim** using the exported file.

4.4 Configure the Internal_Order_Eater e*Way

The component (e*Way or BOB) sends the message to the internal system.

The e*Xchange Internal_Order_Eater e*Way

The e*Xchange example simulates the publication of the message to the internal system.

Configuration Steps

Follow these steps to configure the Internal_Order_Eater e*Way.

1 Create the configuration file.

2 Create the ETDs.

3 Create the Collaboration.

4.4.1 Step 1: Create and Configure the Internal_Order_Eater e*Way

1 Create an e*Way called **Internal_Order_Eater**.

2 In the **e*Way Properties** dialog box, **General** tab, in the **Executable file** area, browse for **stcewfile.exe**.

3 In the **e*Way Properties** dialog box **General** tab, in the **Configuration file** area click **New**.

4 Configure the **Internal_Order_Eater** e*Way parameters using Table 20.

Table 20 Internal_Order_Eater e*Way Parameters

Screen	Parameter	Setting
General Settings	AllowIncoming	NO
	AllowOutgoing	YES
Outbound (send) settings	OutputDirectory	<eGate>\data\internal\eater
	OutputFileName	output_order%d.dat
	(All others)	(Default)
Poller (inbound) settings	(All)	(Default)
Performance Testing	(All)	(Default)

5 When finished editing the e*Way configuration file, save your work and close the e*Way editor.

6 Click **OK** to close the **e*Way Properties** dialog box.

4.4.2 Step 2: Create the Internal_Order_Eater Collaboration

The **Internal_Order_Eater** Collaboration must prepare the data leaving the e*Xchange system. How complicated this task is depends on the state of the data before the **Internal_Order_Eater** Collaboration processes it.

The **Internal_Order_Eater** Collaboration must do the following:

- put the data into the appropriate EDI format
- convert the data to raw data

The e*Xchange Internal_Order_Eater CRS

The **Internal_Order_Eater.tsc** CRS is used to convert the message to raw data, and copies it from the Payload node of the TP_EVENT section of the e*Xchange standard Event to the output ETD.

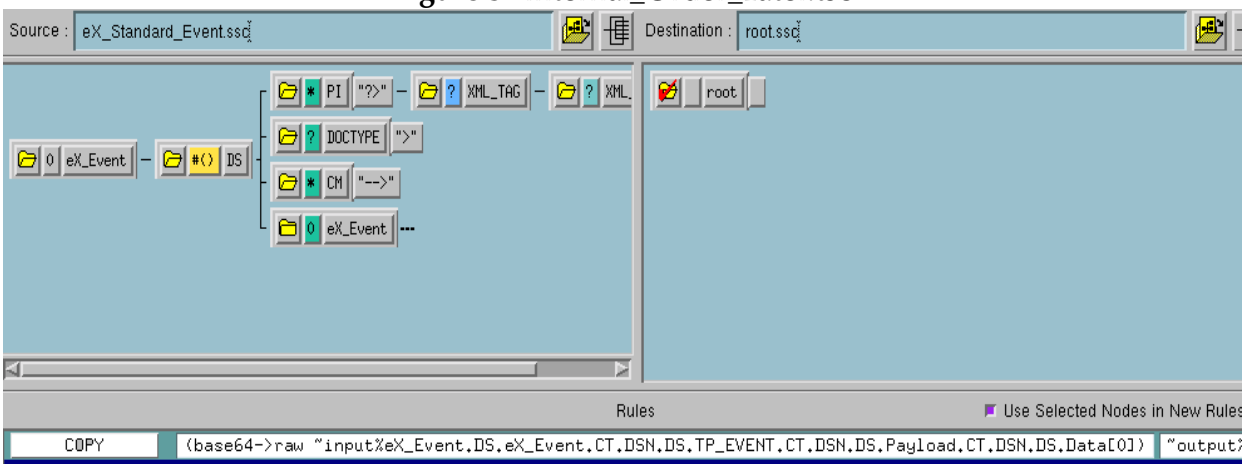
To create and configure the Internal_Order_Eater Collaboration Rule Script

1 Open the Collaboration Editor.

2 Create a new Collaboration Rules script named **Internal_Order_Eater.tsc**. The Source Event Type Definition is **eX_Standard_Event.ssc**. The Destination Event Type Definition is **root.ssc**.

3 Add the rule shown in Figure 5.

Figure 5 Internal_Order_Eater.tsc



Internal_Order_Eater Collaboration Properties Setup

Once the CRS has been created, you must set up the Collaboration and Collaboration Rules Properties for the **Internal_Order_Eater** Component in the Enterprise Manager GUI.

To create and configure the Internal_Order_Eater Collaboration Rule

1 Create a new Collaboration Rule named **Internal_Order_Eater**.

2 From Internal_Order_Eater Collaboration Rule properties, select the **General** tab. Configure as shown in Table 21.

Table 21 Internal_Order_Eater CR configuration - General Tab

Section	Value
Service	Monk
Collaboration Rule	Internal_Order_Eater
Initialization File	monk_scripts\common\load_ext

Important: *To use the Monk function **base64->raw**, you must make sure the file containing this function has been loaded.*

3 Select the **Subscriptions** tab. Select **eX_to_eBPM** and move to the right pane.

4 Select the **Publications** tab. Select **eX_External_Evt** and move to the right pane.

To create and configure the Internal_Order_Eater Collaboration

1 Select the **Internal_Order_Eater** e*Way.

2 Create a new Collaboration named **Internal_Order_Eater**.

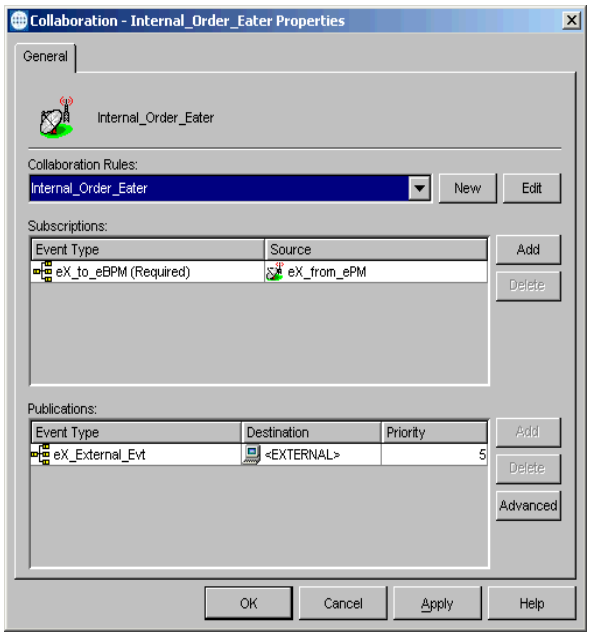
3 Configure the Internal_Order_Eater Collaboration properties using Table 22.

Table 22 Internal_Order_Eater Collaboration configuration

Section	Value
Collaboration Rule	Internal_Order_Eater
Subscriptions	Event Type: eX_to_eBPM Source: eX_from_ePM
Publications	Event Type: eX_External_Evt Destination: <EXTERNAL>

Verify the information in the **Collaboration Properties** dialog box as shown in Figure 6.

Figure 6 Internal_Order_Eater Collaboration Properties



4.5 Configure the Internal_Order_Feeder e*Way

The component (e*Way or BOB) that feeds data into e*Xchange must put the data into the appropriate business protocol format. It must also populate the required fields in the e*Xchange Event that is processed by e*Xchange.

This component is entirely user-defined and must be added to the eXSchema. The type of component to use depends on whether a connection to a system outside e*Gate must be made, and if so, what type of system. Typically, this component is an e*Way that connects to a business application such as SAP that sends out electronic messages. These messages may or may not be in the format required by the trading partner to which they are being sent. If the data is not in the correct format, the e*Way must translate the data into the required format before it is sent to the e*Xchange system for enveloping and forwarding to the trading partner.

The e*Xchange Internal_Order_Feeder e*Way

The e*Xchange example simulates sending the Health Claim message from the internal system.

Configuration Steps

Follow these steps to configure Internal_Order_Feeder e*Way.

1. Create the configuration file.
2. Create the ETDs.
3. Create the Collaboration.

4.5.1 Step 1: Create and Configure the Internal_Order_Feeder e*Way

1. Create a new e*Way named **Internal_Order_Feeder**.
2. In the **e*Way Properties** dialog box, **General** tab, in the **Executable file** area, browse for **stcewfile.exe**.
3. In the **e*Way Properties** dialog box **General** tab, in the **Configuration file** area click **New**.
4. Configure the **Internal_Order_Feeder** e*Way parameters using Table 23.

Table 23 Internal_Order_Feeder e*Way Parameters

Screen	Parameter	Setting
General Settings	(All)	(Default)
Outbound (send) settings	(All)	(Default)
Poller (inbound) settings	PollDirectory	<eGate>\data\internal\feeder
	MultipleRecordsPerFile	NO
	(All others)	(Default)
Performance Testing	(All)	(Default)

5. When finished editing the e*Way configuration file, save your work and close the e*Way editor.
6. Click **OK** to close the **e*Way Properties** dialog box.

4.5.2 Step 2: Create the Internal_Order_Feeder Collaboration

The **Internal_Order_Feeder** Collaboration must prepare the data coming into the e*Xchange system. How complicated this task is depends on the state of the data before the **Internal_Order_Feeder** Collaboration processes it.

The **Internal_Order_Feeder** Collaboration must do the following:

- put the data into the appropriate EDI format

- convert the data to base 64 encoding
- populate the required nodes in the e*Xchange Event sent to e*Xchange for processing

The e*Xchange Internal_Order_Feeder CRS

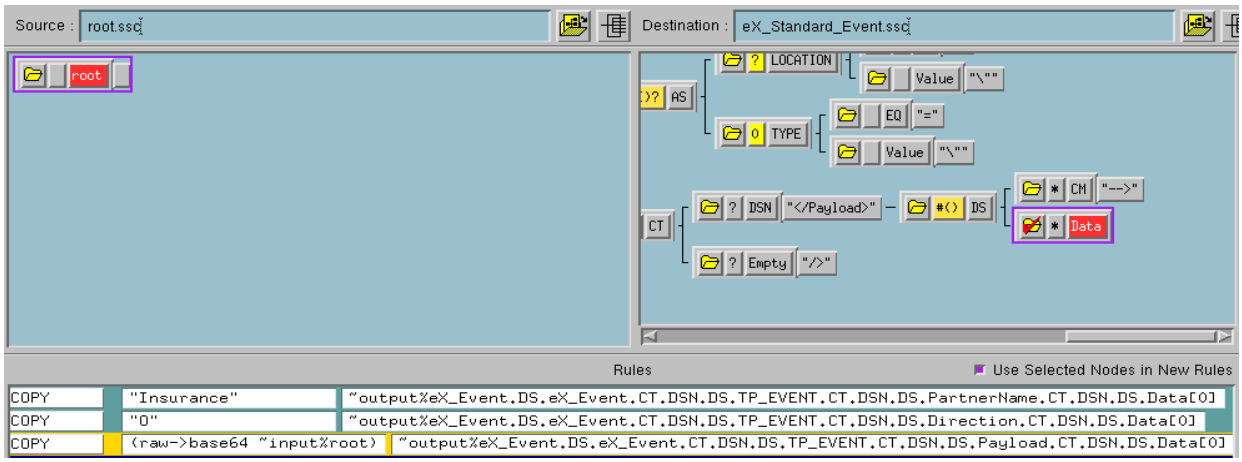
The **Internal_Order_Feeder.tsc** CRS does the following:

- Converts the message to base 64 encoding, and copies it to the Payload node of the TP_EVENT section of the e*Xchange standard Event.
- Copies "O" for outbound to the direction node of the TP_EVENT section.
- Copies the trading partner logical name "Insurance" to the PartnerName node of the TP_EVENT section.

To create and configure the Internal_Order_Feeder Collaboration Rule Script

1. Open the Collaboration Editor.
2. Create a new Collaboration Rules script named **Internal_Order_Feeder.tsc**. The Source Event Type Definition is **root.ssc**. The Destination Event Type Definition is **eX_Standard_Event.ssc**.
3. Add the rules shown in Figure 7.

Figure 7 Internal_Order_Feeder.tsc



Internal_Order_Feeder Collaboration Properties Setup

Once the CRS has been created, you must set up the Collaboration and Collaboration Rules Properties for the **Internal_Order_Feeder** Component in the Enterprise Manager GUI.

To create and configure the Internal_Order_Feeder Collaboration Rule

1. Create a new Collaboration Rule named **Internal_Order_Feeder**.
2. From Internal_Order_Feeder Collaboration Rule properties, select the **General** tab. Configure as shown in Table 24.

Table 24 Internal_Order_Feeder CR configuration - General Tab

Section	Value
Service	Monk
Collaboration Rule	Internal_Order_Feeder
Initialization File	monk_scripts\common\load_ext

Important: *To use the Monk function **raw->base64**, you must make sure the file containing this function has been loaded.*

3 Select the Subscriptions tab. Select **eX_External_Evt** and move it to the right pane.

4 Select the Publications tab. Select **eX_to_ePM** and move it to the right pane.

To create and configure the Internal_Order_Feeder Collaboration

1 Select the **Internal_Order_Feeder** e*Way.

2 Create a new Collaboration named **Internal_Order_Feeder**.

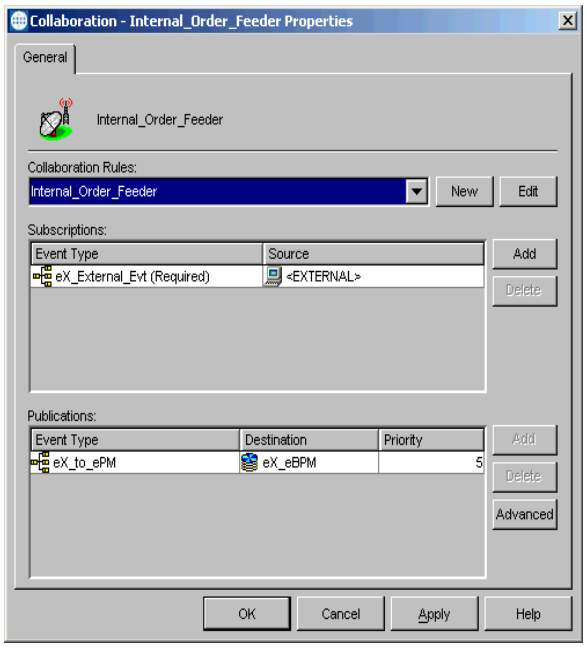
3 Configure the **Internal_Order_Feeder** Collaboration properties using Table 25.

Table 25 Internal_Order_Feeder Collaboration configuration

Section	Value
Collaboration Rule	Internal_Order_Feeder
Subscriptions	Event Type: eX_External_Evt Source: <EXTERNAL>
Publications	Event Type: eX_to_ePM Destination: eX_eBPM

Verify the information in the **Collaboration Properties** dialog box as shown in Figure 8.

Figure 8 Internal_Order_Feeder Collaboration Properties



4.6 Configure the eX_ePM e*Way

The **eX_ePM** e*Way requires only minimal configuration. You must give it the logon information for the e*Xchange database.

To configure the eX_ePM configuration file

1. In the **eX_ePM** e*Way properties, select the **General** tab.
2. In the **Configuration File** area, click **Edit**.
3. Configure the parameters as shown in Table 26.

Table 26 eX_ePM e*Way Parameters

Screen	Parameter	Setting
General Settings	(All)	(Default)
Communication Setup	(All)	(Default)
Monk Configuration	(All)	(Default)
Database Setup	Database Name	(service name of the e*Xchange database)
	User name	ex_admin
	Password	ex_admin
	(All others)	(Default)

4.7 Configure the eX_Poll_Receive_FTP e*Way

Although the **eX_Poll_Receive_FTP** e*Way does not appear in Figure 3, it is used to send information to the **eX_Batch_from_Trading_Partner** e*Way.

The **eX_Poll_Receive_FTP** e*Way requires only minimal configuration. You must give it the logon information for the e*Xchange database.

To configure the eX_Poll_Receive_FTP configuration file

1. In the **eX_Poll_Receive_FTP** e*Way properties, select the **General** tab.
2. In the **Configuration File** area, click **Edit**.
3. Configure the parameters as shown in Table 27.

Table 27 eX_Poll_Receive_FTP e*Way Parameters

Screen	Parameter	Setting
General Settings	(All)	(Default)
Communication Setup	(All)	(Default)
Monk Configuration	(All)	(Default)
Database Setup	Database Name	(service name of the e*Xchange database)
	User name	ex_admin
	Password	ex_admin
	(All others)	(Default)

4.8 Running the Scenario

There are two parts to running the scenario:

1. Sending the health claim to the trading partner
2. Processing the health claim payment sent from the trading partner

To process the Health Claim message

1. Start the Control Broker. At the command line, enter:

```
stccb.exe -rh localhost -rs HealthClaim -ln localhost_cb -un
Administrator -up STC
```

2. Open the e*Gate Monitor. Select the HealthClaim schema.
3. Start the **Internal_Order_Feeder** e*Way.

   This e*Way retrieves the health care claim message and sends it to e*Xchange.
4. Rename **<eGate>\data\hipaa\internal\input\hipaa-837.~in** to **hipaa-837.fin**.

   The file is renamed **hipaa-837.~in** as it is picked up.

5 Start the **eX_Batch_to_Trading_Partner** e*Way.

This e*Way sends the message to the trading partner.

6 Look in the **<egate>\data\hipaa\TP\output** folder. The file **output0.dat** appears.

7 Start the **eX_Poll_Receive_FTP** e*Way.

This e*Way sends configuration information to the **eX_Batch_from_Trading_Partner** e*Way.

8 Start the **eX_Batch_from_Trading_Partner** e*Way.

This e*Way retrieves messages from the trading partner.

9 Start the **Internal_Order_Eater** e*Way.

This e*Way sends the message to the internal system.

10 Rename **<egate>\data\hipaa\TP\input\hipaa-835.dat.backup** to **hipaa-835.dat**.

This sends the health claim payment advice message from the trading partner. The file is renamed **hipaa-837.~in** as it is picked up.

11 Look in the **<egate>\data\hipaa\TP\output** folder. The file **output0.dat** appears.

That completes the first part of the exercise. You can view the results in Message Tracking, in the e*Xchange Partner Manager Web Interface.

Viewing the Results in Message Tracking

You can view the results of the message processing by using the Message Tracking feature of e*Xchange.

Message Tracking shows two entries for the incoming message. This is because an adknowledgment can be sent out immediately, and a response message is sent out later. These two responses to the trading partner are tracked separately.

To view the inbound message in Message Tracking

1 From the e*Xchange Web interface, **Main** page, select **Message Tracking**.

The **TP Profile Selection** page appears.

2 In the **Company Profile** field, select **Insurance**.

3 In the **Trading Partner Profile** field, select **Insurance**.

4 In the **eBusiness Protocol** field, select **X12**.

5 In the **Direction** field, select **Outbound**.

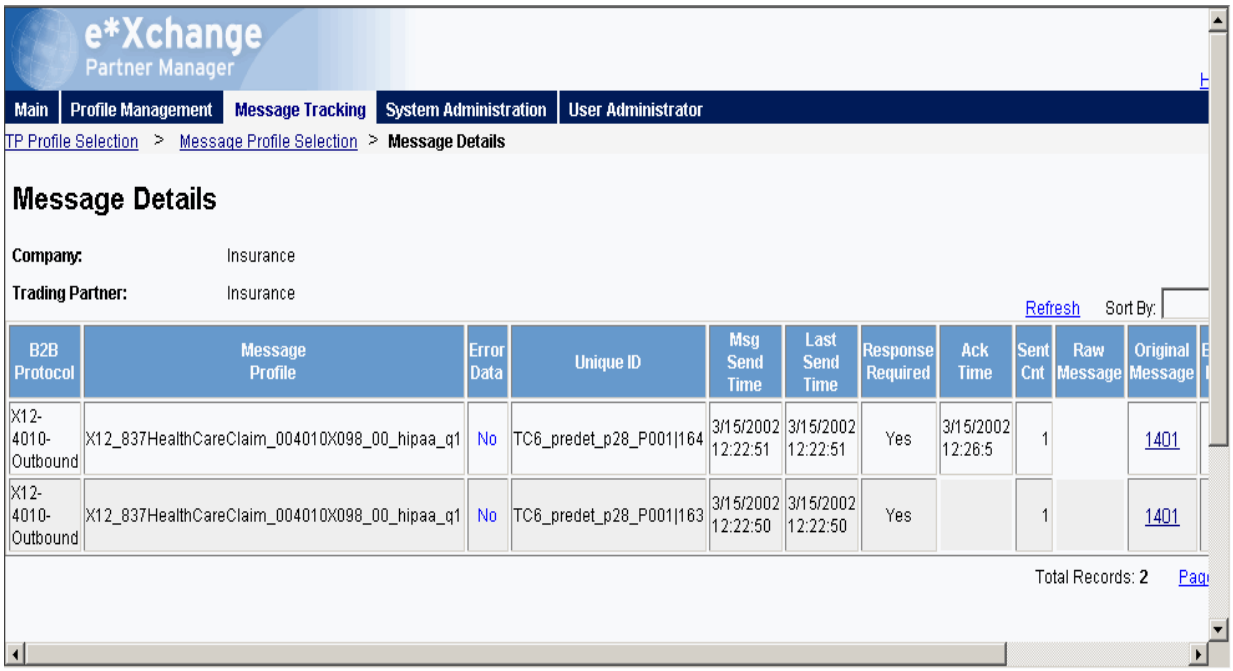
6 Click the **Message Profile Selection**.

7 Select the **X12_837HealthCareClaim_004010X098_00_hipaa_q1** message.

8 Click the **Message Details** link to view the resulting list.

The results are shown in Figure 9.

Figure 9 Message Tracking: Outbound



As shown in Figure 9, e*Xchange records two entries for the message. The top entry is for the original message, for which a response message will be sent. The second entry is for the acknowledgment message.

For one entry, the **Ack Message** column has a link to the message information. Click it to view the acknowledgment message.

Chapter 5

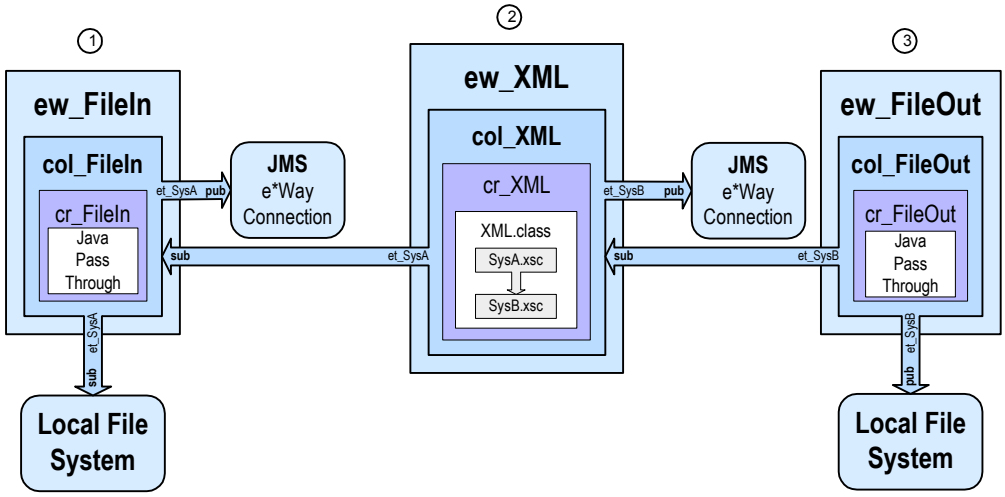
e*Gate Implementation

This chapter discusses the steps involved to create an implementation that converts HIPAA X12 data to or from XML.

e*Gate Solution

The proposed e*Gate solution makes use of the e*Gate Java Collaboration Service to transform the data from the System A format to the System B format. e*Gate is very flexible about where the actual transformation processing can occur as the data moves from System A to System B. The solution uses the Multi-Mode e*Way as the main transformation component and two Java Pass Through file e*Ways to bring data into and send data out from the e*Gate system. Figure 10 shows all the components and their relationships to one another in the complete e*Gate Schema.

Figure 10 XML Scenario Overview



Notes on the XML Scenario Overview

① **ew_FileIn** brings data from System A into e*Gate.

The **col_FileIn** Collaboration in the **ew_FileIn** e*Way subscribes to a location on the local file system. It polls this location for a text file with extension **".fin"** containing data from System A. Then it reads the message, packages the data as an **et_SysA** Event, and publishes the Event to the JMS e*Way Connection.

② **ew_XML** changes the data format.

The **col_XML** Collaboration in the **ew_XML** e*Way subscribes to **et_SysA** Events published by **col_FileIn**. It uses the Java Collaboration Rule **cr_XML** to convert to XML. This rule uses the **XML.class** which implements the transformation. Finally, **col_XML** publishes the **et_SysB** Event to the JMS e*Way Connection.

③ **ew_FileOut** writes the transformed data out to a local file system.

The **col_FileOut** Collaboration in the **ew_FileOut** e*Way subscribes to **et_SysB** Events published by a JMS e*Way Connection. The **cr_FileOut** Collaboration Rule uses the Java Pass Through service to move the data without modifying it. When an **et_SysB** Event is retrieved, the e*Way packages it as a text file and writes it to the specified location on the local file system, completing the end-to-end scenario.

Road Map For Setting Up the Scenario

The basic steps are:

1 Verify the e*Gate installation.

2 Create a new Schema.

3 Create the Event Types and Java ETDs.

4 Create the Collaboration Rules.

5 Add and configure the e*Ways and the JMS e*Way Connection.

6 Add and define the Collaborations that route the data.

7 Review the complete Schema.

8 Start the Schema.

9 Test the Schema.

10 Troubleshoot any problems.

By examining Figure 10 you notice that the road map works from the inside out when creating components. That is, the Event Types and Collaborations are created before creating the e*Ways that use them. This method has the advantage of letting you create all the components of the same type at the same time. It also ensures that the required components are available when you need them.

5.1 Verify the e*Gate Installation

This end-to-end scenario is designed to run on a single machine. Before beginning the configuration process, you must verify that you have all the required software installed on the target machine. Refer to the *e*Gate Integrator Installation Guide* for e*Gate system requirements and instructions to install the e*Gate components.

5.2 Create a New Schema

To create a new Schema

1 Start the e*Gate Enterprise Manager and log in as **Administrator** (or another user with administrator privileges) to the appropriate Registry Host.

2 In the **Open Schema on Registry Host** dialog box, click **New**.

3 In the **Enter New Schema Name** box, type **HIPAA**, and then click **Open**.

The Enterprise Manager opens and displays the new **HIPAA** Schema.

4 At the bottom of the navigator (left) pane, click the **Components** tab.

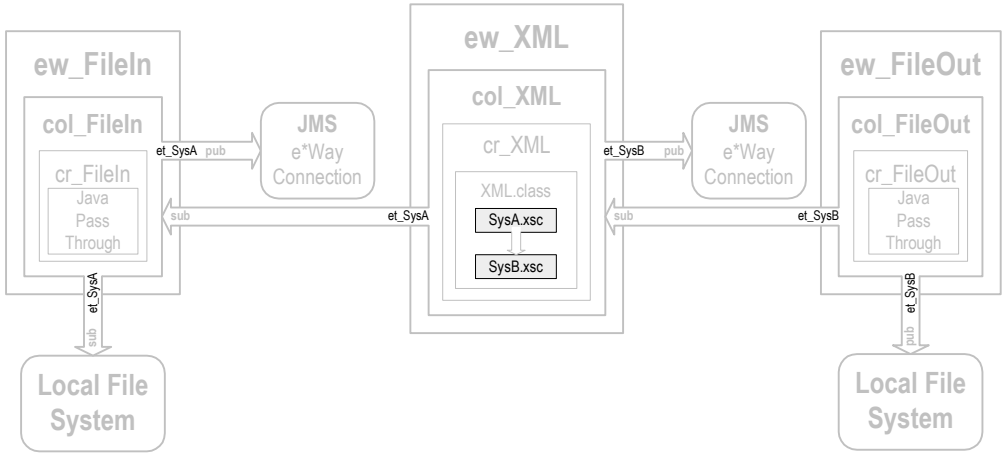
You will perform all configuration steps in the **Components** tab.

5.3 Create the Event Types and Java ETDs

This scenario uses two Event Types. These Event Types both use the same Event Type Definition (ETD); **X12_004010X096_00_hipaaQ3_837_HealCareClai.xsc**. The first Event Type, **et_SysA**, models the ASCII format of the data received from System A. The second Event Type, **et_SysB**, models the XML format required by System B.

Figure 11 shows where these parts fit into the collection of interrelated components that make up the finished Schema.

Figure 11 Event Types and Java ETDs

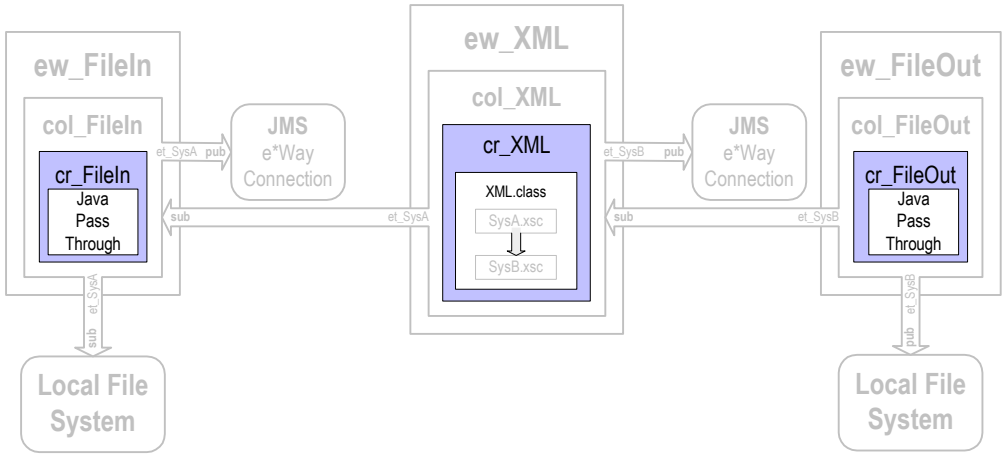


5.4 Create the Collaboration Rules

This scenario uses three Collaboration Rules: two Java Pass Through rules and one Java Collaboration. The Java Pass Through rules, **cr_FileIn** and **cr_FileOut**, are used to route the Events through the e*Gate system and the Java Collaboration Rule **cr_XML** is used to transform the Event from Event Type **et_SysA** to Event Type **et_SysB**.

Figure 12 shows where these parts fit into the collection of interrelated components that make up the finished Schema.

Figure 12 Collaboration Rules



5.4.1 Create the Java Pass Through Collaborations

The Java Pass Through Collaborations are used to bring data into and take data away from the e*Gate system. The following procedure explains how to create the Java Pass Through Collaborations used in this scenario.

Create the cr_FileIn and cr_FileOut Collaboration Rules

1. In the navigator pane of the Enterprise Manager, click the **Collaboration Rules** folder.
2. On the **File** menu, point to **New**, and then click **Collaboration Rules**.
3. In the **New Collaboration Rules Component** dialog box, type **cr_FileIn** for the Collaboration Rule name, and then click **OK**.

   cr_FileIn is added to the list of Collaboration Rules in the Enterprise Manager editor pane.
4. On the list of Collaboration Rules, double-click **cr_FileIn**.
5. In the **Collaboration Rules** section, click **Find** and explore to **collaboration_rules\STCLibrary**, and then double-click **STCJavaPassThrough.class**.

   The path to STCJavaPassThrough.class displays in the **Collaboration Rules** section of the dialog box, and the path to **STCJavaPassThrough.ctl** displays in the **Initialization File** section. The **STCJavaPassThrough.class** file configures the Collaboration Mapping Instances for you. You are not required to make any other changes to **cr_FileIn**.
6. Click **OK** to close the **Collaboration Rules - cr_FileIn Properties** dialog box.
7. Repeat steps 2 through 6 to create the **cr_FileOut** Collaboration Rule. Substitute **cr_FileOut** for the Collaboration Rule name.

5.4.2 Create the Java Collaboration Rule

The procedure for creating a Collaboration Rule that uses the Java Collaboration Service is different from creating other e*Gate Collaboration Rules. Use the following procedure to start the Java Collaboration Editor and create the Java Collaboration Rule used by this scenario.

Create cr_XML and Start the Java Collaboration Editor

1. Use steps 1 through 5 from the procedure described in **"Create the cr_FileIn and cr_FileOut Collaboration Rules" on page 49** to create a new Collaboration Rule named **cr_XML**.
2. Click the **Collaboration Mapping** tab, and then click **Add Instance**.

   An instance row is added to the **Collaboration Mapping** tab.
3. In the **Instance Name** column, type **In** for the instance name.
4. Click **Find**, and explore to **etd\templates\Hipaa_2000**, and then double-click **X12_004010X096_00_hipaaQ3_837_HealCareClai.xsc**.

X12_004010X096_00_hipaaQ3_837_HealCareClai.xsc is added to the **ETD** column of the instance row. You are not required to make any other changes to **In**.

5 Add another ETD instance. Use **Out** for the instance name.

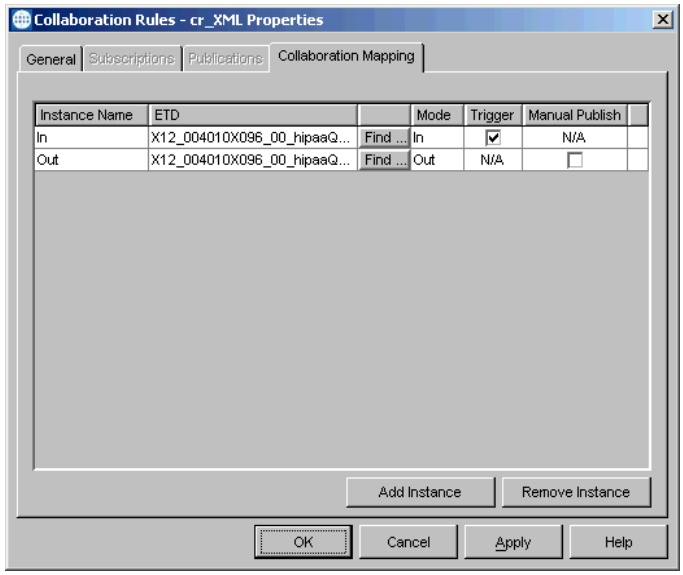
Important: *The Java ETD instance names must be unique per schema.*

6 Find and enter **X12_004010X096_00_hipaaQ3_837_HealCareClai.xsc** as the ETD for **Out**.

7 Click in the **Mode** cell for **Out,** and then click **Out**.

You do not need to make any other changes to **Out**. The completed **Collaboration Mapping** tab looks like the one shown in Figure 13.

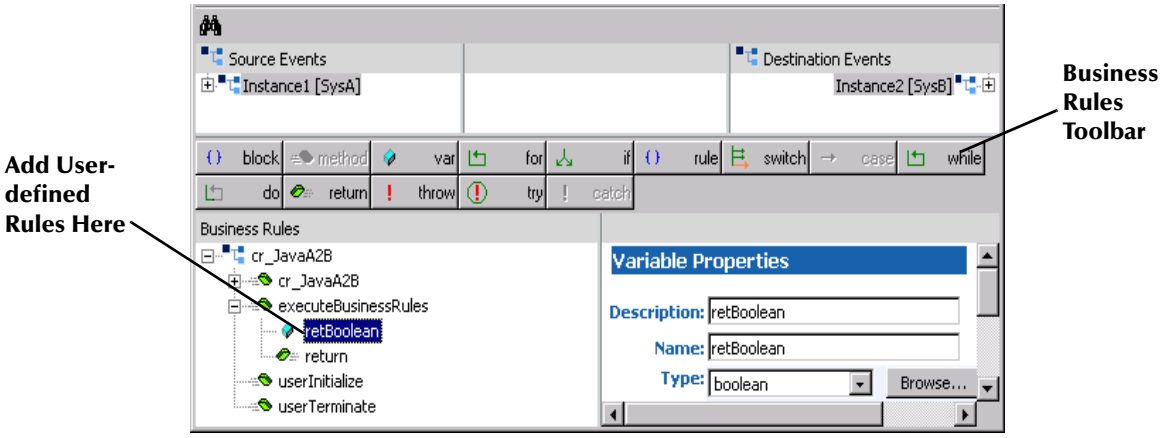
Figure 13 Completed Collaboration Mapping Tab



8 Click the **General** tab, and then in the **Collaboration Rules** area click **New**.

The Java Collaboration Editor opens a new Collaboration Rule with **In (SysA)** as the source Event and **Out (SysB)** as the destination Event as shown in Figure 14.

Figure 14 XML Before Adding User-Defined Code



Create cr_XML.class in the Java Collaboration Rules Editor

1 On the **View** menu, click **Display Code**.

The **Business Rules** pane now displays the Java code in addition to the code labels.

2 In the **Source Events** and **Destination Events** panes, expand **In** and **Out** to display the leaf nodes of the ETDs.

3 In the **Business Rules** pane, under the **executeBusinessRules** method, click the **retBoolean** variable.

All the user-defined rules you add for this scenario are added within the **executeBusinessRules** method, and placed between the **retBoolean** variable and the **return** rule (see Figure 14).

4 With the **retBoolean** variable selected, drag the **ISA_InteContHead** segment node from the **Source Events** pane to the **ISA_InteContHead** segment node in the **Destination Events** pane.

5 With the Source **X12_004010X096_00_hipaaQ3_837_HealCareClaiInner** segment selected, click the **For** button on the **Business Rules** toolbar.

A **For** rule is added to the **executeBusinessRules** method.

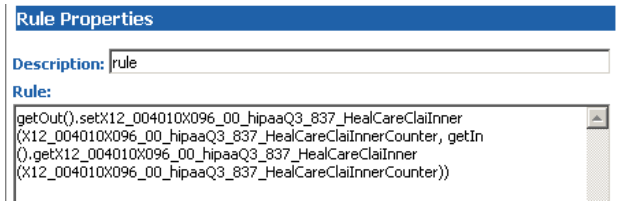
6 With the **For** rule selected in the **Business Rules** pane, do the following:

A Drag the **X12_004010X096_00_hipaaQ3_837_HealCareClaiInner** segment in the **Source Events** pane to the **X12_004010X096_00_hipaaQ3_837_HealCareClaiInner** segment in the **Destination Events** pane.

B Ensure that the repetition instance for both source and destination is **X12_004010X096_00_hipaaQ3_837_HealCareClaiInnerCounter**.

C The **Sibling or Child** dialog appears. Click **Child**.

Figure 15 Completed Rule



7 With the **For** rule selected, drag the **IEA_InteContTrai** segment node from the **Source Events** pane to the **IEA_InteContTrai** segment node in the **Destination Events** pane.

8 With the rule created in step 7 selected, click the **Rule** button on the **Business Rules** toolbar.

A **Rule** is added to the **executeBusinessRules** method.

9 Drag the **setXMLOutput** rule from the **Destination Events** pane to the Rules box.

The **Parameter for method: setXMLOutput()** dialog appears.

10 Enter **true**, and click **OK**.

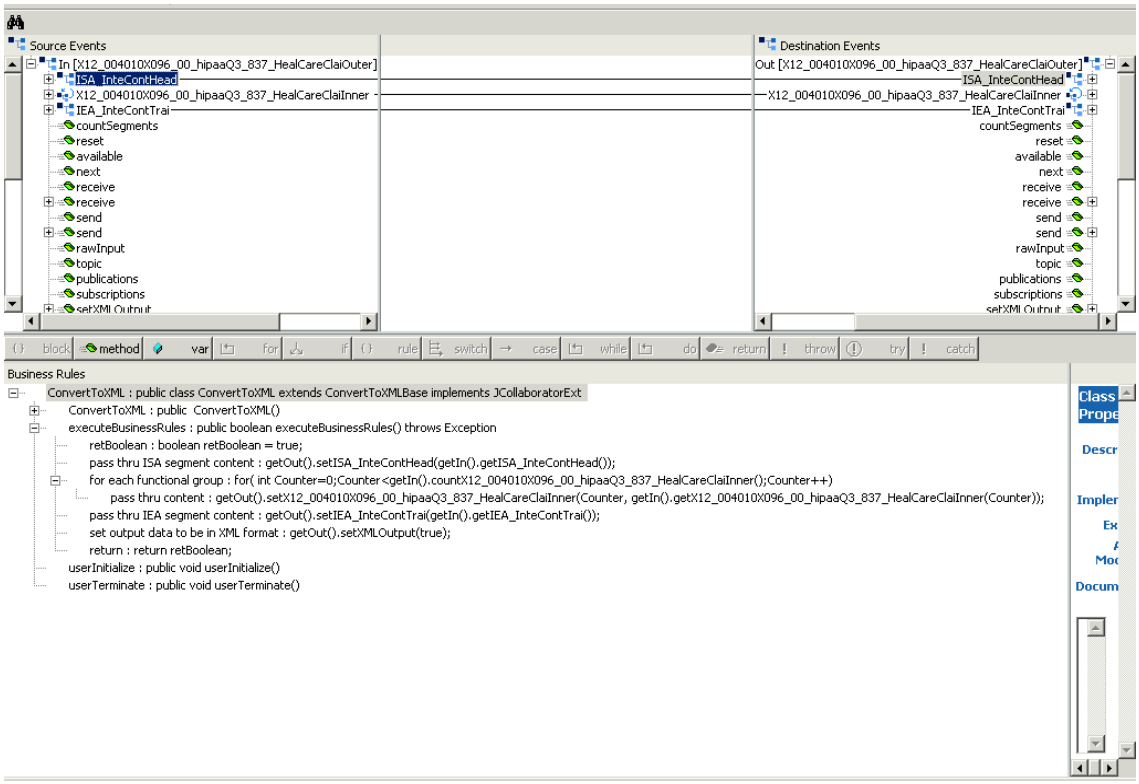
11 On the **File** menu, click **Save**.

12 In the **Save** dialog box, navigate to the **collaboration_rules** folder, and then save the **cr_XML.xpr** file.

13 On the **File** menu, click **Compile**.

The Java source code is compiled. When the compiler is finished "Compile Completed" is displayed in the Compile/Debug pane. The Compile/Debug pane also displays any errors generated by the compilation process. Clear any errors before you continue.

Figure 16 ConvertToXML After Adding User-Defined Code



14 On the **File** menu, click **Exit**.

You may be prompted to save changes. The JCE closes and in the **Collaboration Rules - cr_XML Properties** dialog box, **colaboration_rules\cr_XML.class** is entered in the **Collaboration Rules** box and **colaboration_rules\cr_XML.ctl** is entered in the **Initialization file** box.

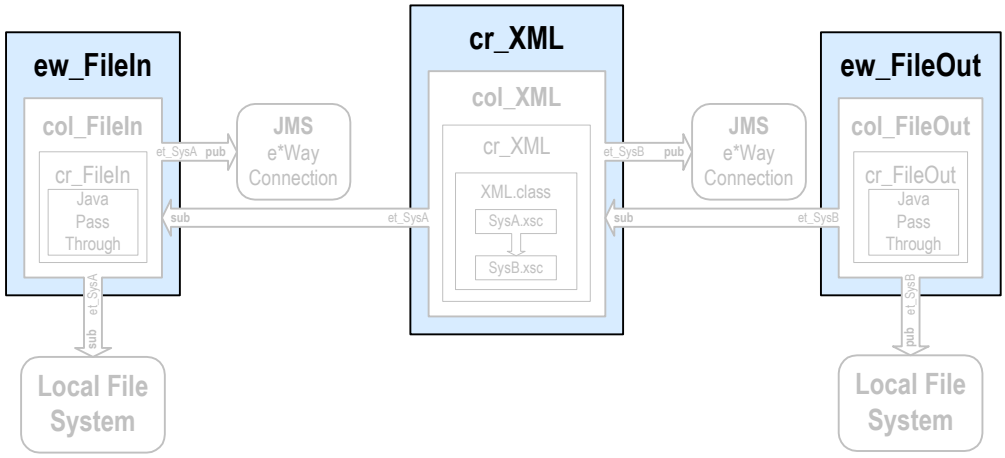
15 Click **OK** to close the **Collaboration Rules - cr_XML Properties** dialog box.

5.5 Add the e*Ways and e*Way Connection

After you have created your ETDs and Collaborations, you are ready to add and configure the e*Gate components that use these parts.

Figure 17 highlights the components added in this step.

Figure 17 e*Ways and JMS e*Way Connection



5.5.1 Add and Configure the e*Ways

Add and Configure the ew_FileIn file e*Way

1. In the e*Gate Enterprise Manager, in the navigation pane, click the Control Broker (***hostname*_cb**).
2. On the **File** menu, point to **New**, point to **Module**, and then click **e*Way**.
3. In the **New e*Way Component** dialog box, type **ew_FileIn** for the e*Way name, and then click **OK**.

   The **ew_FileIn** e*Way is added to the Schema.
4. Right-click **ew_FileIn**, and then click **Properties**.
5. In the **e*Way - ew_FileIn Properties** dialog box, in the **Executable file** area, click **Find**.
6. In the **File Selection** dialog box, browse for and double-click the file **stcewfile.exe**.

   The **bin\stcewfile.exe** file is added as the executable file, causing the component to become a file e*Way.
7. In the **Configuration file** area, click **New**.

   The e*Way Configuration File Editor opens with a default file e*Way configuration file ready for editing.
8. In the **Goto Section** list, click **Poller (inbound) settings**.
9. In the **PollDirectory** box, type **C:\eGate\Client\Data\HIPAA** and then press **ENTER**.

   C:\eGate\Client\Data\HIPAA is added as the directory to be polled to the **PollDirectory** list. No other changes are necessary to the **ew_FileIn** e*Way's configuration file.
10. On the **File** menu, click **Save**.

11 In the **Save As** dialog box, click **Save** to accept the default filename (**ew_FileIn.cfg**) and save the file.

12 On the **File** menu, click **Close** to quit the e*Way Configuration File Editor.

The **configs\stcewfile\ew_FileIn.cfg** file is added to the **Configuration file** area in the **e*Way - ew_FileIn Properties** dialog box.

13 Click the **Start Up** tab, and then select the **Start automatically** check box.

14 Click **OK** to close the **e*Way - ew_FileIn Properties** dialog box.

Add and Configure the ew_FileOut file e*Way

Adding the **ew_FileOut** e*Way follows the same general procedure as that outlined for adding the **ew_FileIn** e*Way above.

1 Use steps 1 through 7 from **"Add and Configure the ew_FileIn file e*Way" on page 54** to add another file e*Way named **ew_FileOut** and open its configuration file for editing.

2 In the e*Way Configuration File Editor, in **General Settings**, click **NO** for **AllowIncoming**, and **YES** for **AllowOutgoing**.

3 In the **Goto Section** list, click **Outbound (send) settings**.

4 Add **C:\eGate\Client\Data\HIPAA** as the default **OutputDirectory**.

5 Add **HIPAAoutput%d.dat** as the default **OutputFileName**.

No other changes are necessary to the **ew_FileOut** e*Way's configuration file.

6 On the **File** menu, click **Save**.

7 In the **Save As** dialog box, click **Save** to accept the default file name (**ew_FileOut.cfg**) and save the file.

8 On the **File** menu, click **Close** to quit the e*Way Configuration File Editor.

9 Click the **Start Up** tab, and then select the **Start automatically** check box.

10 Click **OK** to close the **e*Way - ew_FileOut Properties** dialog box.

5.5.2 Add the Multi-Mode e*Way

1 In the e*Gate Enterprise Manager, in the navigation pane, click the Control Broker (***hostname_cb***).

2 On the **File** menu, point to **New**, point to **Module**, and then click **e*Way**.

3 In the **New e*Way Component** dialog box, type **ew_XML** for the e*Way name, and then click **OK**.

The **ew_XML** e*Way is added to the Schema.

4 Right-click the **ew_XML** e*Way in the editor pane, and then click **Properties**.

5 In the **Configuration file** area, click **New**.

The e*Way Configuration File Editor opens with a default Multi-Mode e*Way configuration file.

6 Scroll to the bottom of the **JVM Settings** parameters and click **Remote debugging port number**.

7 In the **Remote debugging port number** box, type **8000**, and then press **ENTER**.

8000 is listed as the **Remote debugging port number**. No other changes are necessary to the **ew_XML** e*Way's configuration file.

Important: ***In-schema debugging*** *must be enabled on the Participating Host for this to work. See the e*Gate Integrator Installation Guide for more information.*

8 On the **File** menu, click **Save**.

9 In the **Save As** dialog box, click **Save** to accept the default filename (**ew_XML.cfg**) and save the file.

10 On the **File** menu, click **Close** to quit the e*Way Configuration File Editor.

11 Click the **Start Up** tab, and then select the **Start automatically** check box.

12 Click **OK** to close the **e*Way - ew_XML Properties** dialog box.

5.5.3 Configure the IQ Manager

1 In the e*Gate Enterprise Manager, in the navigation pane, double-click the IQ manager (***hostname***_**iqmgr**).

2 In the **Configuration File** area, click **New**.

3 From the **File** menu click **Save** and then click **Save** again to accept the default name.

4 Click the **Start Up** tab, select the **Start automatically** check box, and then click **OK**.

5.5.4 Add the JMS e*Way Connection

1 In the navigator pane of the Enterprise Manager, click the **e*Way Connections** folder.

2 In the editor pane, right-click, point to **New**, and then click **e*Way Connection**.

3 In the **New e*Way Connection Component** dialog box, type **JMS** for the e*Way Connection name, and then click **OK**.

JMS is added to the list of e*Way Connections.

4 In the editor pane, double-click **JMS**.

The **e*Way Connection - JMS Properties** dialog box displays.

5 From the **e*Way Connection Type** drop-down list, select **SeeBeyond JMS**.

6 In the **Configuration File** area, click **New**.

7 From the **JMS IQ Manager** drop-down list, select your IQ Manager.

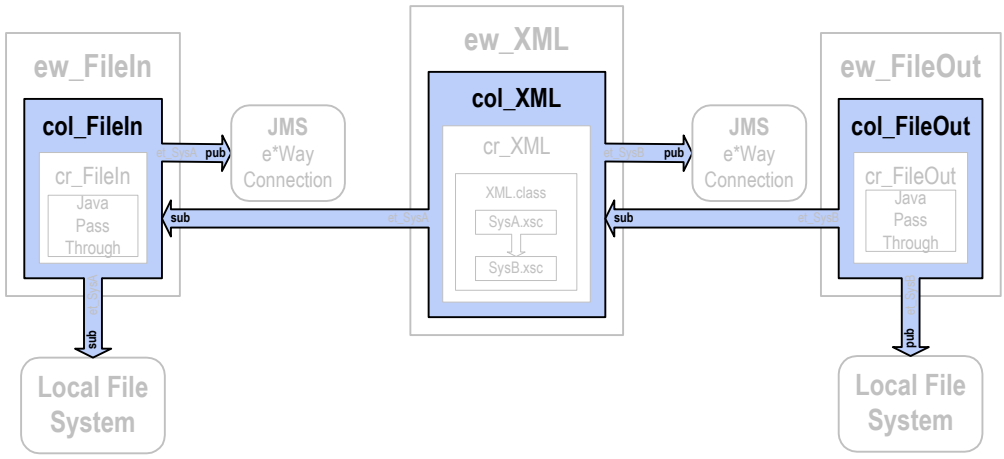
8 Click **OK** to save your configuration file.

9 Click **OK** to close the **e*Way Connection - JMS Properties** dialog box.

5.6 Add the Collaborations that Route the Data

The Collaborations are used by the e*Ways to route the data through the e*Gate system. Typically, the Collaborations are configured in upstream to downstream order. Figure 18 shows the relationships of the Collaborations to the remainder of the parts that make up the complete Schema.

Figure 18 Collaborations Showing Publish and Subscribe Relationships



5.6.1 Add and Configure col_FileIn

The **col_FileIn** Collaboration brings the data into the e*Gate system. Use the following procedure to add and configure **col_FileIn**.

1. In the e*Gate Enterprise Manager, in the navigation pane, click the **ew_FileIn** e*Way.
2. On the **File** menu, point to **New**, and then click **Collaboration**.
3. In the **New Collaboration Component** dialog box, type **col_FileIn** for the Collaboration name, and then click **OK**.
4. In the editor pane, double-click **col_FileIn**.

   The **Collaboration - col_FileIn Properties** dialog box displays.
5. In the **Collaboration Rules** list, click **cr_FileIn**.
6. In the **Subscriptions** area, click **Add**.

   A row is added to the **Subscriptions** box.
7. In the **Instance Name** column, select **JavaPassThroughIn**. In the **Event Type** column, click **et_SysA** on the list, and then in the Source column, select **<EXTERNAL>** from the list.
8. In the **Publications** area, click **Add**.

   A row is added to the **Publications** box.

9 In the **Instance Name** column, select **JavaPassThroughOut**. In the **Event Type** column, click **et_SysA** on the list, and then in the **Destination** column, select **JMS** from the list.

10 Click **OK** to close the **Collaboration - col_FileIn Properties** dialog box.

5.6.2 Add and Configure col_XML

The **col_XML** Collaboration changes the data from the **et_SysA** Event Type to the **et_SysB** Event Type. Use the following procedure to add and configure **col_XML**.

1 Use steps 1 through 4 from **"Add and Configure col_FileIn" on page 57** to add a Collaboration to the **ew_XML** e*Way named **col_XML** and open its properties dialog box.

2 In the **Collaboration Rules** list, click **cr_XML**.

3 In the **Subscriptions** area, click **Add**.

A row is added to the **Subscriptions** box.

4 Double-click in the **Instance Name** column and click **In** on the list.

5 Double-click in the **Event Type** column and click **et_SysA** on the list.

6 Double-click in the **Source** column and click **JMS** on the list.

7 In the **Publications** area, click **Add**.

A row is added to the **Publications** area.

8 Double-click in the **Instance Name** column, and then click **Out** on the list.

9 Double-click in the **Event Type** column, and then click **et_SysB** on the list.

10 Double-click in the **Destination** column, and then click **JMS** on the list.

11 Click **OK** to close the **Collaboration - col_XML Properties** dialog box.

5.6.3 Add and Configure col_FileOut

The **col_FileOut** Collaboration sends the transformed data out of the e*Gate system. Use the following procedure to add and configure **col_FileOut**.

1 Use steps 1 through 4 from **"Add and Configure col_FileIn" on page 57** to add a Collaboration to the **ew_FileOut** e*Way named **col_FileOut** and open its properties dialog box.

2 In the **Collaboration Rules** list, click **cr_FileOut**.

3 In the **Subscriptions** area, click **Add**.

4 A row is added to the **Subscriptions** box.

5 In the **Instance Name** column, select **JavaPassThroughIn**. In the **Event Type** column, click **et_SysB** on the list, and then in the **Source** column, select **JMS** from the list.

6 In the **Publications** area, click **Add**.

A row is added to the **Publications** area.

7 In the **Instance Name** column, select **JavaPassThroughOut**. In the **Event Type** column, click **et_SysB** on the list, and in the **Destination** column, select **<EXTERNAL>** from the list.

8 Click **OK** to close the **Collaboration - col_FileOut Properties** dialog box.

5.7 Test the Scenario

The following road map steps are covered in this section:

Step 7: Review the complete Schema.

Step 8: Test the Schema.

Step 9: Start the Schema.

Step 10: Troubleshoot any problems.

5.7.1 Review the Complete Schema

Table 28 lists all the components for the Schema. Check all the settings. Substitute the name of the machine running the schema for hostname where applicable.

Table 28 HIPAA Components

Component	Logical Name	Settings
Schema	HIPAA	
Control Broker	*hostname*_cb	
IQ Manager	*hostname*_iqmgr	Service = SeeBeyond JMS Config file = *hostname*_iqmgr.cfg Start Up = Auto
Event Type	et_SysA	SysA.xsc
	et_SysB	SysB.xsc
Java ETD	SysA.xsc	Package Name = SysApackage
	SysB.xsc	Package Name = SysBpackage
Collaboration Rule	cr_FileIn	Service = Java JavaPassThroughIn GenericInEvent.ssc Trigger JavaPassThroughOut GenericOutEvent.ssc
	cr_XML	Service = Java In SysA.xsc In Trigger Out SysB.xsc Out
	cr_FileOut	Service = Java JavaPassThroughIn GenericInEvent.ssc Trigger JavaPassThroughOut GenericOutEvent.ssc

Table 28 HIPAA Components

Component	Logical Name	Settings
Java Collaboration Rule	cr_XML.class	Source = In Destination = Out
Inbound e*Way	ew_FileIn	Executable = stcewfile.exe Config file = ew_FileIn.cfg Start Up = Auto Collaboration = col_FileIn
Outbound e*Way	ew_FileOut	Executable = stcewfile.exe Config file = ew_FileOut.cfg Start Up = Auto Collaboration = col_FileOut
Multi-Mode e*Way	ew_XML	Executable = stceway.exe Config file = ew_XML.cfg Start Up = Auto Collaboration = col_XML
JMS e*Way Connection	JMS	Service = SeeBeyond JMS Config file = jms.cfg
Collaboration	col_FileIn	Collab Rule = cr_FileIn Subscription = et_SysA from <EXTERNAL> Publication = et_SysA to JMS
	col_XML	Collab Rule = cr_XML Subscription = et_SysA from JMS Publication = et_SysB to JMS
	col_FileOut	Collab Rule = cr_FileOut Subscription = et_SysB from JMS Publication = et_SysB to <EXTERNAL>

5.7.2 Test the Schema

Test the scenario by sending data into the system and verifying the output.

Start the Schema

1. Use the following command to start the Control Broker from a command line.

   ```
   stccb.exe -rh hostname -rs HIPAA -ln hostname_cb -un username -up password
   ```

2. Start the e*Gate Monitor.
3. Verify that all the components in the schema are running.

Testing in Windows 2000

1. Once all the scenario components have been started successfully, use Windows Explorer to navigate to **c:\eGate\client\data\HIPAA**.
2. Change the file extension on the input file **HIPAAinput.txt** to **.fin**.

3 Click **Yes** to confirm this choice.

4 Verify that the extension changes to **.~in** indicating that the **ew_FileIn** e*Way has retrieved the file.

5 Almost immediately, the output file, **HIPAAoutput#.dat**, should appear in the directory, indicating a successful conclusion to the test.

shows a section of the original HIPAA X12 data, and shows a section of the HIPAA XML data.

Figure 19 Original data

```
ISA*00*          *00*          *01*6264712000     *01*6264716000     *010126*1709*U*0
00003231 8*0*T*:~GS*HC*901234572000*908887732000*010126*1709*32318*T*004010X096~ST*837
32318~BHT*0019*00*Hipaa_012601_W02*20010126*1615*CH~REF*87*3920394930203~NM1*41*1
*JOHNSON*BARBARA*T***46*9012345918341~PER*IC*ARTHUR JONES*ED*(614)555-1212*ED*(614)55
1212*EM*(614)555-1212~NM1*40*2*SMITH*****46*111222333~HL*1**20*1~PRV*BI*ZZ*
123456789009876543217689584 73~CUR*85*USA~NM1*85*2*JONES*****24*43202~N3*PO BOX 123*15
WEST 57TH STREET~N4*CINCINNATI*OH*43017*US~REF*0B*500~REF*06*3920394930203~PER*IC*MAG
```

Figure 20 Converted data

```
- <envelope format="X12">
  - <segment code="ISA" name="Interchange Control Header">
    - <element code="I01" name="Authorization Information Qualifier">
        <value>00</value>
      </element>
    - <element code="I02" name="Authorization Information">
        <value />
      </element>
    - <element code="I03" name="Security Information Qualifier">
        <value>00</value>
      </element>
    - <element code="I04" name="Security Information">
        <value />
      </element>
    - <element code="I05" name="Interchange ID Qualifier">
        <value>01</value>
      </element>
    - <element code="I06" name="Interchange Sender ID">
        <value>6264712000</value>
      </element>
    - <element code="I05" name="Interchange ID Qualifier">
        <value>01</value>
      </element>
    - <element code="I07" name="Interchange Receiver ID">
        <value>6264716000</value>
      </element>
```

Appendix A

ASC X12 Overview

This appendix provides an overview of the X12 standard, including:

- An overview of ASC X12, including the structure of an X12 envelope, data elements, and syntax.
- An explanation of how to use the generic message structures provided as an add-on to e*Gate to help you quickly create the structures you need for various X12 transactions.

For specific information on HIPAA, refer to **Chapter 2**, **"HIPAA Overview" on page 9**.

A.1 Introduction to X12

The following sections provide an introduction to X12.

A.1.1 What Is ASC X12?

ASC X12 is an EDI (electronic data interchange) standard, developed for the electronic exchange of machine-readable information between businesses.

The Accredited Standards Committee (ASC) X12 was chartered by the American National Standards Institute (ANSI) in 1979 to develop uniform standards for interindustry electronic interchange of business transactions—electronic data interchange (EDI). The result was the X12 standard.

The ASC X12 body develops, maintains, interprets, and promotes the proper use of the ASC X12 standard. Data Interchange Standards Association (DISA) publishes the ASC X12 standard and the UN/EDIFACT standard. The ASC X12 body comes together three times a year to develop and maintain EDI standards. Its main objective is to develop standards to facilitate electronic interchange relating to business transactions such as order placement and processing, shipping and receiving information, invoicing, and payment information.

The ASC X12 EDI standard is used for EDI within the United States. UN/EDIFACT is broadly used in Europe and other parts of the world.

X12 was originally intended to handle large batches of transactions. However, it has been extended to encompass real-time processing (transactions sent individually as they are ready to send, rather than held for batching) for some healthcare transactions to accommodate the healthcare industry.

A.1.2 What Is a Message Structure?

The term *message structure* (also called a transaction set structure) refers to the way in which data elements are organized and related to each other for a particular EDI transaction.

In e*Gate, a message structure is called an Event Type Definition (ETD). Each message structure (ETD) consists of the following:

- Physical hierarchy

  The predefined way in which envelopes, segments, and data elements are organized to describe a particular X12 EDI transaction.

- Delimiters

  The specific predefined characters that are used to mark the beginning and end of envelopes, segments, and data elements.

- Properties

  The characteristics of a data element, such as the length of each element, default values, and indicators that specify attributes of a data element—for example, whether it is required, optional, or repeating.

The transaction set structure of a claim that is sent from a payer to a provider defines the header, trailer, segments, and data elements required by claim transactions. Installation of X12 templates for a specific version includes transaction set structures for each of the transactions available in that version.

e*Xchange Partner Manager uses e*Gate Event Type Definitions, which are based on the X12 message structures, to verify that the data in the messages coming in or going out is in the correct format. There is a message structure for each X12 transaction.

The list of transactions provided is different for each version of X12, and for each customized implementation. This book addresses the transactions covered by the May 1999 and May 2000 implementations of the HIPAA standard.

A.2 Components of an X12 Envelope

X12 messages are all ASCII text, with the exception of the BIN segment which is binary.

Each X12 message is made up of a combination of the following elements:

- Data elements
- Segments
- Loops

Elements are separated by delimiters.

More information on each of these is provided below.

A.2.1 Data Elements

The data element is the smallest named unit of information in the ASC X12 standard. Data elements can be broken down into two types. The distinction between the two is strictly a matter of how they are used. The two types are:

- Simple

  If a data element occurs in a segment outside the defined boundaries of a composite data structure it is called a simple data element.

- Composite

  If a data element occurs as an ordinally positioned member of a composite data structure it is called a composite data element.

Each data element has a unique reference number; it also has a name, description, data type, and minimum and maximum length.

A.2.2 Segments

A segment is a logical grouping of data elements. In X12, the same segment can be used for different purposes. This means that a field's meaning can change based on the segment. For example:

- The NM1 segment is for *any* name (patient, provider, organization, doctor)
- The DTP segment is for *any* date (date of birth, discharge date, coverage period)

For more information on the X12 enveloping segments, refer to **"Structure of an X12 Envelope" on page 65**.

A.2.3 Loops

Loops are sets of repeating ordered segments. In X12 you can locate elements by specifying:

- The transaction set (for example, 270)
- The loop (for example, "loop 1000" or "info. receiver loop")
- The occurrence of the loop
- The segment (for example, BGN)
- The field number (for example, 01)
- The occurrence of the segment (if it is a repeating segment)

A.2.4 Delimiters

In an X12 message, the various delimiters act as syntax, dividing up the different elements of a message. The delimiters used in the message are defined in the interchange control header, the outermost layer enveloping the message. For this reason, there is flexibility in the delimiters that are used.

No suggested delimiters are recommended as part of the X12 standards, but the industry-specific implementation guides do have recommended delimiters.

The default delimiters used by the SeeBeyond HIPAA ETD Library are the same as those recommended by the industry-specific implementation guides. These delimiters are shown in Table 29.

Table 29 Default Delimiters in X12 ETD Library

Type of Delimiter	Default Value
Segment terminator	~ (tilde)
Data element separator	* (asterisk)
Subelement (component) separator	: (colon)

Within e*Xchange Partner Manager, delimiters are specified at the outer envelope level. The delimiters you define are applied to all transaction types.

If you do not specify delimiters, e*Xchange expects the default delimiters as shown in Table 29.

Note: *It is important to note that errors could result if the transmitted data itself includes any of the characters that have been defined as delimiters. Specifically, the existence of asterisks within transmitted application data is a known issue in X12, and can cause problems with translation.*

A.3 Structure of an X12 Envelope

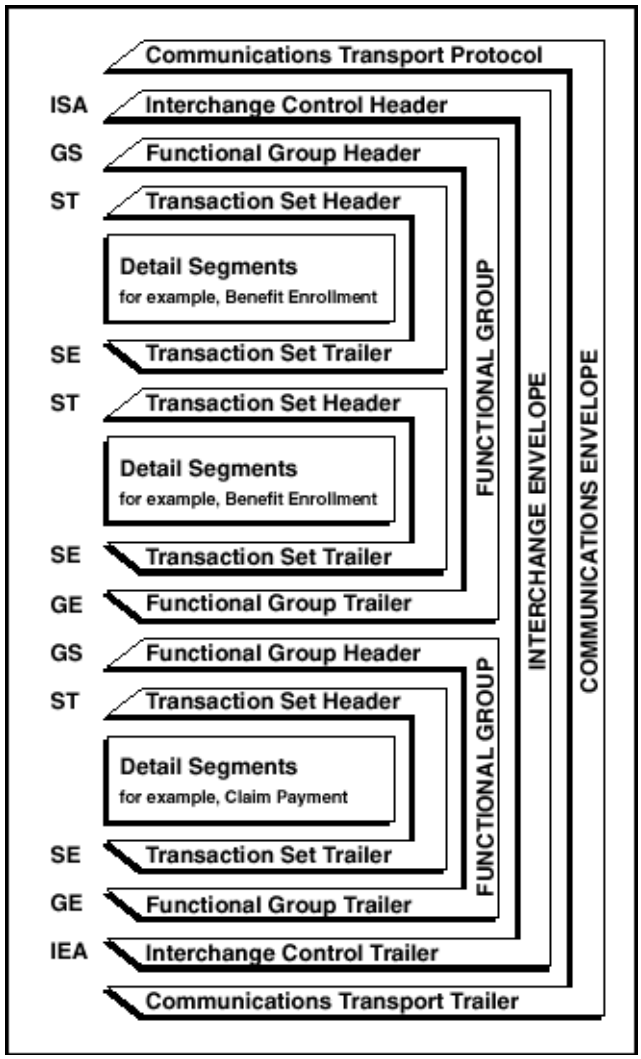
The rules applying to the structure of an X12 envelope are very strict, to ensure the integrity of the data and the efficiency of the information exchange.

The actual X12 message structure has three main levels. From the highest to the lowest they are:

- Interchange Envelope
- Functional Group
- Transaction Set

A schematic of X12 envelopes is shown in Figure 21. Each of these levels is explained in more detail in the following sections.

Figure 21 X12 Envelope Schematic



Note: *The above schematic is from Appendix B of an ASC X12 Implementation Guide.*

Figure 22 shows the standard segment table for an X12 997 (Functional Acknowledgment) as it appears in the X12 standard and in most industry-specific implementation guides.

Figure 22 X12 997 Segment Table

Table 1 - Header

POS. #	SEG. ID	NAME	REQ. DES.	MAX USE	LOOP REPEAT
010	ST	Transaction Set Header	M	1	
020	AK1	Functional Group Response Header	M	1	
		LOOP ID - AK2			999999
030	AK2	Transaction Set Response Header	O	1	
		LOOP ID - AK2/AK3			999999
040	AK3	Data Segment Note	O	1	
050	AK4	Data Element Note	O	99	
060	AK5	Transaction Set Response Trailer	M	1	
070	AK9	Functional Group Response Trailer	M	1	
080	SE	Transaction Set Trailer	M	1	

Figure 23 shows the same transaction as viewed in the Monk ETD Editor in e*Gate.

Figure 23 X12 997 Viewed in Monk ETD Editor

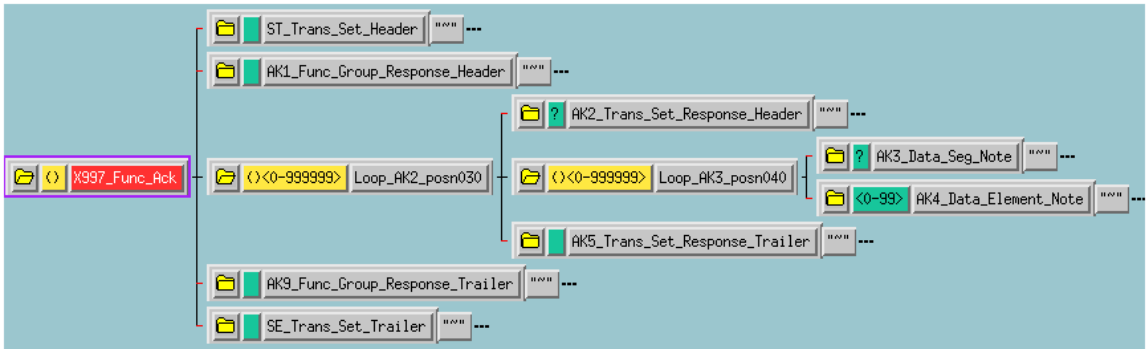


Figure 24 shows the same transaction as viewed in the Java ETD Editor.

Figure 24 X12 997 Viewed in Java ETD Editor

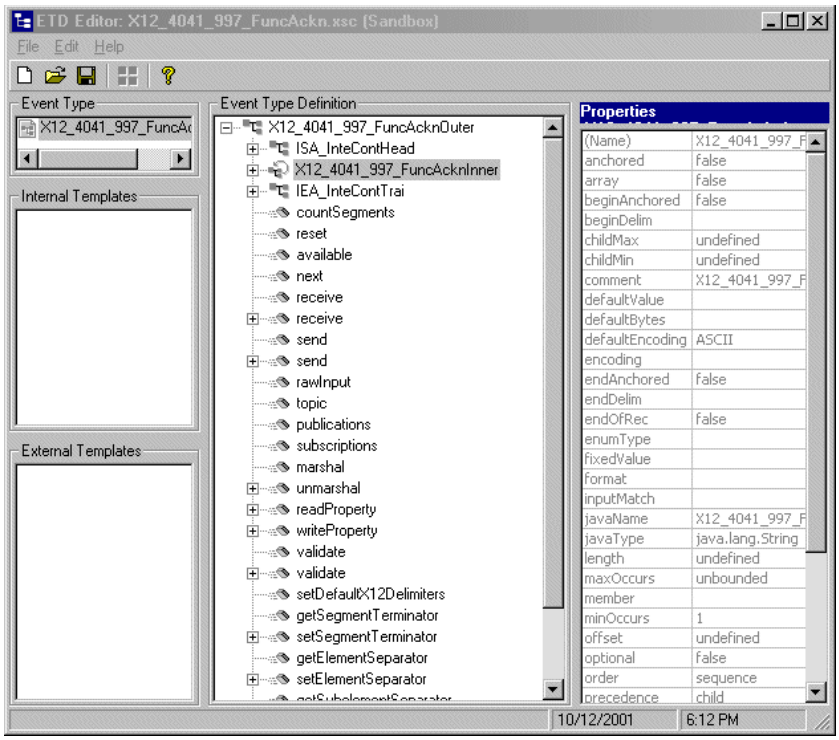


Figure 25 shows an example of a positive 997 acknowledgment, as viewed in the Message Tracking window in the e*Xchange Partner Manager.

Note: *The message shown in Figure 25 was part of a batch and therefore includes only the ST/SE (transaction set) envelope layer.*

Figure 25 Positive 997 (Functional Acknowledgment) Viewed in Message Tracking

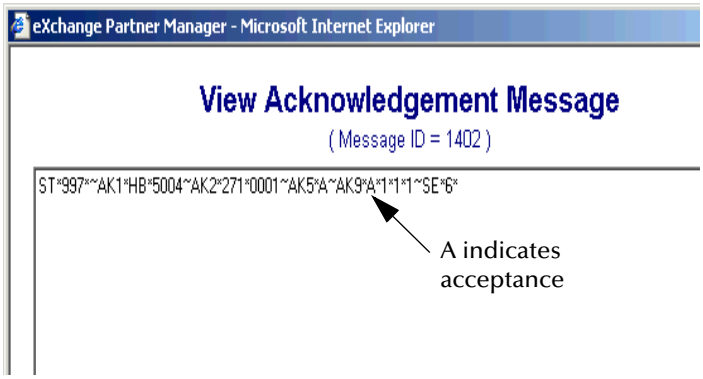
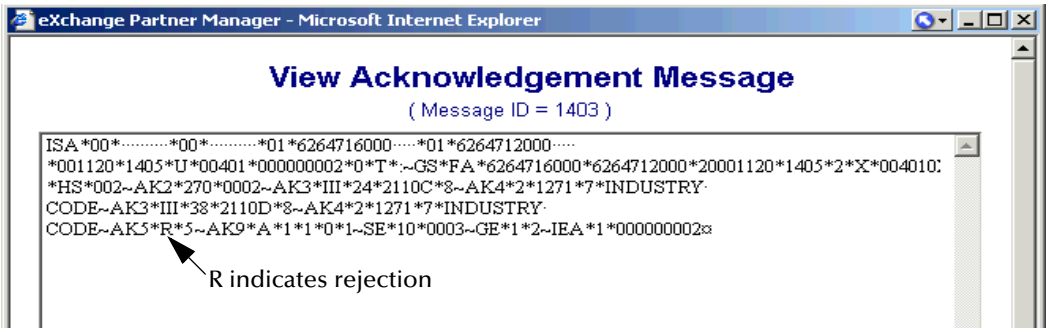


Figure 26 shows an example of a negative 997 acknowledgment, as viewed in the Message Tracking window in the e*Xchange Partner Manager.

Note: *The message shown in Figure 26 is an interactive message and therefore includes all enveloping layers.*

Figure 26 Negative 997 (Functional Acknowledgment) Viewed in Message Tracking



A.3.1 Transaction Set (ST/SE)

Each transaction set (also called a transaction) contains three things:

- A transaction set header
- A transaction set footer
- A single message, enveloped within the header and footer

The transaction has a three-digit code, a text title, and a two-letter code; for example, **997, Functional Acknowledgment (FA)**.

The transaction is comprised of logically related pieces of information, grouped into units called segments. For example, one segment used in the transaction set might convey the address: city, state, ZIP code, and other geographical information. A transaction set can contain multiple segments. For example, the address segment could be used repeatedly to convey multiple sets of address information.

The X12 standard defines the sequence of segments in the transaction set and also the sequence of elements within each segment. The relationship between segments and elements could be compared to the relationship between records and fields in a database environment.

Figure 27 Example of a Transaction Set Header (ST)

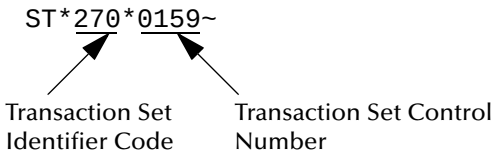
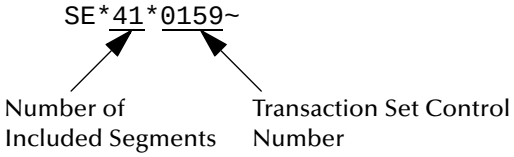


Figure 28 Example of a Transaction Set Trailer (SE)



A.3.2 Functional Group (GS/GE)

A functional group is comprised of one or more transaction sets, all of the same type, that can be batched together in one transmission. The functional group is defined by the header and trailer; the Functional Group Header (GS) appears at the beginning, and the Functional Group Trailer (GE) appears at the end. Many transaction sets can be included in the functional group, but all transactions must be of the same type.

Within the functional group, each transaction set is assigned a functional identifier code, which is the first data element of the header segment. The transaction sets that comprise a specific functional group are identified by this functional ID code.

The functional group header (GS) segment contains the following information:

- Functional ID code (the two-letter transaction code; for example, PO for an 850 Purchase Order, HS for a 270 Eligibility, Coverage or Benefit Inquiry) to indicate the type of transaction in the functional group
- Identification of sender and receiver
- Control information (the functional group control numbers in the header and trailer segments must be identical)
- Date and time

The functional group trailer (GE) segment contains the following information:

- Number of transaction sets included
- Group control number (originated and maintained by the sender)

Figure 29 Example of a Functional Group Header (GS)

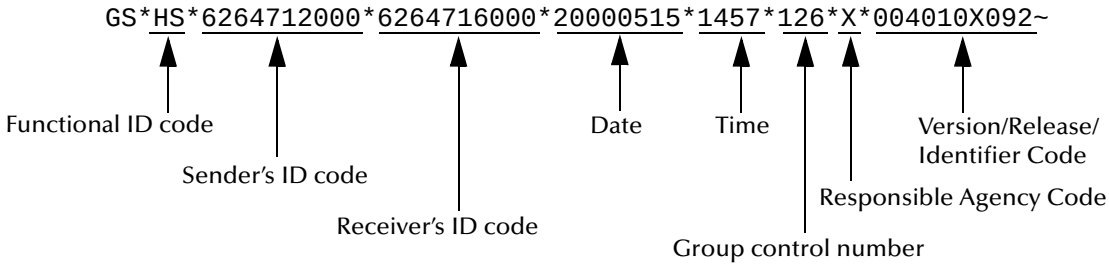
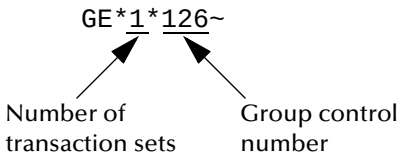


Figure 30 Example of a Functional Group Trailer (GE)



A.3.3 Interchange Envelope (ISA/IEA)

The interchange envelope is the wrapper for all the data to be sent in one batch. It can contain multiple functional groups. This means that transactions of different types can be included in the interchange envelope, with each type of transaction stored in a separate functional group.

The interchange envelope is defined by the header and trailer; the Interchange Control Header (ISA) appears at the beginning, and the Interchange Control Trailer (IEA) appears at the end.

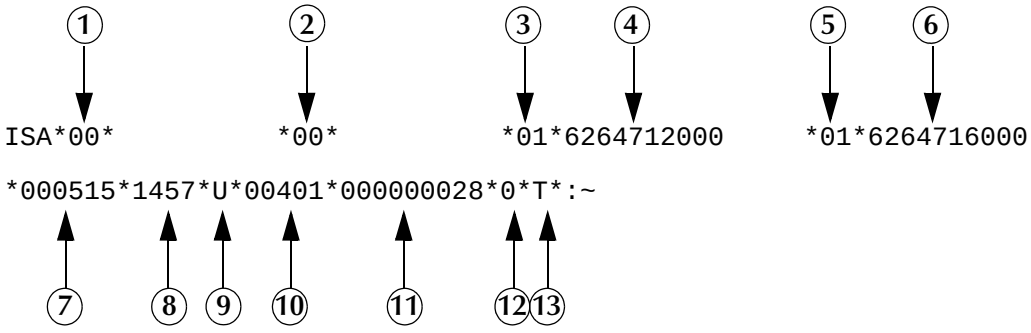
As well as enveloping one or more functional groups, the interchange header and trailer segments include the following information:

- Data element separators and data segment terminator
- Identification of sender and receiver
- Control information (used to verify that the message was correctly received)
- Authorization and security information, if applicable

The sequence of information that is transmitted is as follows:

- Interchange header
- Optional interchange-related control segments
- Actual message information, grouped by transaction type into functional groups
- Interchange trailer

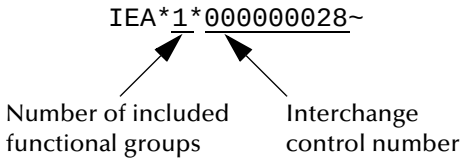
Figure 31 Example of an Interchange Header (ISA)



Interchange Header Segments from Figure 31:

1 Authorization Information Qualifier
2 Security Information Qualifier
3 Interchange ID Qualifier
4 Interchange Sender ID
5 Interchange ID Qualifier
6 Interchange Receiver ID
7 Date
8 Time
9 Repetition Separator
10 Interchange Control Version Number
11 Interchange Control Number
12 Acknowledgment Requested
13 Usage Indicator

Figure 32 Example of an Interchange Trailer (IEA)



A.3.4 Control Numbers

The X12 standard includes a control number for each enveloping layer:

- ISA13—Interchange Control Number
- GS06—Functional Group Control Number
- ST02—Transaction Set Control Number

The control numbers act as identifiers, useful in message identification and tracking. The e*Xchange Partner Manager includes a flag for each control number, so you can choose not to assign control numbers to outgoing messages and not to store control numbers on incoming messages.

ISA13 (Interchange Control Number)

The ISA13 is assigned by the message sender. It must be unique for each interchange. This is the primary means used by e*Xchange Partner Manager to identify an individual interchange.

GS06 (Functional Group Control Number)

The GS06 is assigned by the sender. It must be unique within the Functional Group assigned by the originator for a transaction set.

Note: *The Functional Group control number GS06 in the header must be identical to the same data element in the associated Functional Group trailer, GE02.*

ST02 (Transaction Set Control Number)

The ST02 is assigned by the sender, and is stored in the transaction set header. It must be unique within the Functional Group.

Note: *The control number in ST02 must be identical with the SE02 element in the transaction set footer, and must be unique within a Functional Group (GS-GE). Once you have defined a value for ST02, e*Xchange Partner Manager uses the same value for SE02.*

A.4 Acknowledgment Types

X12 includes two types of acknowledgment, the TA1 Interchange Acknowledgment and the 997 Functional Acknowledgment.

A.4.1 TA1, Interchange Acknowledgment

The TA1 acknowledgment verifies the interchange envelopes only. The TA1 is a single segment and is unique in the sense that this single segment is transmitted without the GS/GE envelope structures. A TA1 acknowledgment can be included in an interchange with other functional groups and transactions.

A.4.2 997, Functional Acknowledgment

The 997 includes much more information than the TA1. The 997 was designed to allow trading partners to establish a comprehensive control function as part of the business exchange process.

There is a one-to-one correspondence between a 997 and a functional group. Segments within the 997 identify whether the functional group was accepted or rejected. Data elements that are incorrect can also be identified.

Many EDI implementations have incorporated the acknowledgment process into all of their electronic communications. Typically, the 997 is used as a functional acknowledgment to a functional group that was transmitted previously.

The 997 is the acknowledgment transaction recommended by ASC X12.

The acknowledgment of the receipt of a payment order is an important issue. Most corporate originators want to receive at least a Functional Acknowledgment (997) from the beneficiary of the payment. The 997 is created using the data about the identity and address of the originator found in the ISA and/or GS segments.

Some users argue that the 997 should be used only as a point-to-point acknowledgment and that another transaction set, such as the Application Advice (824) should be used as the end-to-end acknowledgment.

A.4.3 Application Acknowledgments

Application acknowledgments are responses sent from the destination system back to the originating system, acknowledging that the transaction has been successfully or unsuccessfully completed. The application advice (824) is a generic application acknowledgment that can be used in response to any X12 transaction. However, it has to be set up as a response transaction; only TA1 and 997 transactions are sent out automatically.

Other types of responses from the destination system to the originating system, which may also be considered application acknowledgments, are responses to query transactions—for example, the Eligibility Response (271) which is a response to the Eligibility Inquiry (270).Other types of responses from the destination system to the originating system, which may also be considered application acknowledgments, are

responses to query transactions—for example, the Eligibility Response (271) which is a response to the Eligibility Inquiry (270).

A.5 Key Parts of EDI Processing Logic

The five key parts of EDI processing logic are listed in Table 30. The table describes each term, and lists its language analogy along with its associated e*Gate Collaboration scripts.

Table 30 Key Parts of EDI Processing

Term	Description	Language Analogy	e*Gate Collaboration Scripts
structures	format, segments, loops	syntax	ETD files or structures
validations	data contents "edit" rules	semantics	validation scripts
translations (also called mapping)	reformatting or conversion	translation	translation scripts
enveloping	header and trailer segments	envelopes	part of translation
acks	acknowledgments	return receipt	e*Way scripts

e*Gate uses the structures, validations, translations, enveloping, and acknowledgments listed below to support HIPAA.

A.5.1 Structures

The Event Type Definition library for HIPAA includes pre-built ETDs for all supported HIPAA versions.

A.5.2 Validations, Translations, Enveloping, Acknowledgments

e*Gate does not include any pre-built validations, transformations, enveloping, or acknowledgments. These scripts can be built in either the Monk or Java versions of the Collaboration Rules Editor graphical user interface (GUI). These GUIs provide a user-friendly drag-and-drop front end for creating Monk or Java scripts.

However, installation of the e*Xchange Partner Manager includes a set of custom Monk validations for HIPAA transactions.

Note: *In e*Gate, translations are called Collaborations.*

A.5.3 X12 Acknowledgments in e*Xchange Partner Manager

All X12 acknowledgments are automatically handled in e*Xchange Partner Manager. This allows you to configure the transaction set, if any, that is expected as the acknowledgment. e*Xchange Partner Manager can automatically create any type of X12 acknowledgment, including TA1, 997, 824, and transaction-specific acknowledgments.

For more information on X12 acknowledgment types, refer to **"Acknowledgment Types" on page 73**.

A.5.4 Trading Partner Agreements

There are three levels of information that guide the final format of a specific transaction. These three levels are:

- The ASC X12 standard

  ASC X12 publishes a standard structure for each X12 transaction.

- Industry-specific Implementation Guides

  Specific industries publish Implementation Guides customized for that industry. Normally, these are provided as recommendations only. However, in certain cases, it is extremely important to follow these guidelines. Specifically, since HIPAA regulations are law, it is important to follow the guidelines for these transactions closely.

- Trading Partner Agreements

  It is normal for trading partners to have individual agreements that supplement the standard guides. The specific processing of the transactions in each trading partner's individual system might vary between sites. Because of this, additional documentation that provides information about the differences is helpful to the site's trading partners and simplifies implementation. For example, while a certain code might be valid in an implementation guide, a specific trading partner might not use that code in transactions. It would be important to include that information in a trading partner agreement.

A.6 Additional Information

For more information on X12, visit the following Web sites:

- For X12 standard:

  http://www.disa.org

- For Implementation Guides: Washington Publishing Company at

  http://www.wpc-edi.com

Note: *This information is correct at the time of going to press; however, SeeBeyond has no control over these sites. If you find the links are no longer correct, use a search engine to search for* ***X12****.*

Appendix B

HIPAA Files

This appendix provides information on the HIPAA files provided with e*Gate and e*Xchange. For more information on the SeeBeyond HIPAA solution, refer to **Chapter 3**, **"The SeeBeyond Solution" on page 19**.

B.1 e*Xchange

The following files are provided with e*Xchange.

May 1999 Files

The May 1999 HIPAA transactions included with e*Xchange version 4.5 are listed in Table 31.

Table 31 HIPAA Transactions (May 1999) Provided with e*Xchange

File Name	Transaction
X12_270EligibCoverageBenefitInquiry_004010X092_hipaa	270 (Eligibility Coverage or Benefit Inquiry)
X12_271EligibCoverageBenefitInfo_004010X092_hipaa	271 (Eligibility Coverage or Benefit Information)
X12_276HealthCareClaimStatusRequest_004010X093_hipaa	276 (Health Care Claim Status Request)
X12_277HealthCareClaimStatusNotification_004010X093_hipaa	277 (Health Care Claim Status Notification)
X12_278HealthCareServicesReviewInfo_004010X094_hipaa_a1	278 (Health Care Services Review Information: Request for Review)
X12_278HealthCareServicesReviewInfo_004010X094_hipaa_a3	278 (Health Care Services Review Information: Response to Request)
X12_820PaymentOrderRemittanceAdvice_004010X061_hipaa	820 (Payment Order Remittance Advice)
X12_834BenefitEnrollmentandMaint_004010X095_hipaa	834 (Benefit Enrollment and Maintenance)
X12_835HealthCareClaimPaymentAdvice_004010X091_hipaa	835 (Health Care Claim Payment Advice)

Table 31 HIPAA Transactions (May 1999) Provided with e*Xchange (Continued)

File Name	Transaction
X12_837HealthCareClaim_004010X098_hipaa_q1	837 (Health Care Claim: Professional)
X12_837HealthCareClaim_004010X097_hipaa_q2	837 (Health Care Claim: Dental)
X12_837HealthCareClaim_004010X096_hipaa_q3	837 (Health Care Claim: Institutional)

May 2000 Files

The May 2000 HIPAA transactions included with e*Xchange version 4.5 are listed in Table 32.

Table 32 HIPAA Transactions (May 2000) Provided with e*Xchange

File Name	Transaction
X12_270EligibCoverageBenefitInquiry_004010X092_00_hipaa	270 (Eligibility Coverage or Benefit Inquiry)
X12_271EligibCoverageBenefitInfo_004010X092_00_hipaa	271 (Eligibility Coverage or Benefit Information)
X12_276HealthCareClaimStatusRequest_004010X093_00_hipaa	276 (Health Care Claim Status Request)
X12_277HealthCareClaimStatusNoti_004010X093_00_hipaa	277 (Health Care Claim Status Notification)
X12_278HealthCareServicesReviewInfo_004010X094_00_hipaa_a1	278 (Health Care Services Review Information: Request for Review)
X12_278HealthCareServicesReviewInfo_004010X094_00_hipaa_a3	278 (Health Care Services Review Information: Response to Request)
X12_820PaymentOrderRemittanceAdvice_004010X061_00_hipaa	820 (Payment Order Remittance Advice)
X12_834BenefitEnrollmentandMaintenance_004010X095_00_hipaa	834 (Benefit Enrollment and Maintenance)
X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa	835 (Health Care Claim Payment Advice)
X12_837HealthCareClaim_004010X096_00_hipaa_q3	837 (Health Care Claim: Professional)
X12_837HealthCareClaim_004010X097_00_hipaa_q2	837 (Health Care Claim: Dental)
X12_837HealthCareClaim_004010X098_00_hipaa_q1	837 (Health Care Claim: Institutional)

B.1.1 HIPAA e*Xchange Files for e*Gate

The following files are provided with e*Xchange, but as they include the GS/GE and ISA/IEA enveloping, these are suitable for use outside e*Xchange when a complete Event structure is required. For example, when using e*Gate to translate from X12 to a business application's proprietary data format.

These files are stored in the same location as the standard e*Xchange format HIPAA files. The file names have "_xlate" (for May 1999 files) or "_xlat" (for May 2000 files) appended to the file name to indicate that these are the translation files and include the interchange control and functional group header and footer.

May 1999 Files

The May 1999 HIPAA transactions included with e*Xchange version 4.5 are listed in Table 31.

Table 33 HIPAA Transactions (May 1999) Provided with e*Xchange

File Name	Transaction
X12_270EligibCoverageBenefitInquiry_004010X092_hipaa_xlate	270 (Eligibility Coverage or Benefit Inquiry)
X12_271EligibCoverageBenefitInfo_004010X092_hipaa_xlate	271 (Eligibility Coverage or Benefit Information)
X12_276HealthCareClaimStatusRequest_004010X093_hipaa_xlate	276 (Health Care Claim Status Request)
X12_277HealthCareClaimStatusNotification_004010X093_hipaa_xlate	277 (Health Care Claim Status Notification)
X12_278HealthCareServicesReviewInfo_004010X094_hipaa_a1_xlate	278 (Health Care Services Review Information: Request for Review)
X12_278HealthCareServicesReviewInfo_004010X094_hipaa_a3_xlate	278 (Health Care Services Review Information: Response to Request)
X12_820PaymentOrderRemittanceAdvice_004010X061_hipaa_xlate	820 (Payment Order Remittance Advice)
X12_834BenefitEnrollmentandMaint_004010X095_hipaa_xlate	834 (Benefit Enrollment and Maintenance)
X12_835HealthCareClaimPaymentAdvice_004010X091_hipaa_xlate	835 (Health Care Claim Payment Advice)
X12_837HealthCareClaim_004010X098_hipaa_q1_xlate	837 (Health Care Claim: Professional)
X12_837HealthCareClaim_004010X097_hipaa_q2_xlate	837 (Health Care Claim: Dental)
X12_837HealthCareClaim_004010X096_hipaae_q3_xlate	837 (Health Care Claim: Institutional)

May 2000 Files

The May 2000 HIPAA transactions included with e*Xchange version 4.5 are listed in Table 32.

Table 34 HIPAA Transactions (May 2000) Provided with e*Xchange

File Name	Transaction
X12_270EligibCoverageBenefitInquiry_004010X092_00_hipaa_xlat	270 (Eligibility Coverage or Benefit Inquiry)
X12_271EligibCoverageBenefitInfo_004010X092_00_hipaa_xlat	271 (Eligibility Coverage or Benefit Information)
X12_276HealthCareClaimStatusRequest_004010X093_00_hipaa_xlat	276 (Health Care Claim Status Request)
X12_277HealthCareClaimStatusNotification_004010X093_00_hipaa_xlat	277 (Health Care Claim Status Notification)
X12_278HealthCareServicesReviewInfo_004010X094_00_hipaa_a1_xlat	278 (Health Care Services Review Information: Request for Review)
X12_278HealthCareServicesReviewInfo_004010X094_00_hipaa_a3_xlat	278 (Health Care Services Review Information: Response to Request)
X12_820PaymentOrderRemittanceAdvice_004010X061_00_hipaa_xlat	820 (Payment Order Remittance Advice)
X12_834BenefitEnrollmentandMaint_004010X095_00_hipaa_xlat	834 (Benefit Enrollment and Maintenance)
X12_835HealthCareClaimPaymentAdvice_004010X091_00_hipaa_xlat	835 (Health Care Claim Payment Advice)
X12_837HealthCareClaim_004010X096_00_hipaa_q3_xlat	837 (Health Care Claim: Professional)
X12_837HealthCareClaim_004010X097_00_hipaa_q2_xlat	837 (Health Care Claim: Dental)
X12_837HealthCareClaim_004010X098_00_hipaa_q1_xlat	837 (Health Care Claim: Institutional)

5.7.3 e*Gate Files for HIPAA Transactions

The X12 portion of the HIPAA ETD Library provides Java Event Type Definitions (**.xsc** and **.jar** files) for all nine standard X12 transactions that have been adopted by HIPAA, as listed in Table 35 and Table 36. These ETDs are stored in the following locations:

<eGate>\server\registry\repository\default\etd\templates\Hipaa_1999

<eGate>\server\registry\repository\default\etd\templates\Hipaa_2000

These transactions are based on the October 1997 X12 standard; that is, Version 4, Release 1, Sub-release 0 (004010) (version 4010).

Table 35 HIPAA 1999 X12 Transactions

File	Transaction Name
X12_004010X092_99_hipaa270_EligCoveOrBeneInqu	270 (Eligibility Coverage or Benefit Inquiry)
X12_004010X092_99_hipaa271_EligCoveOrBeneInfo	271 (Eligibility Coverage or Benefit Information)
X12_004010X093_99_hipaa276_HealCareClaiStatRequ	276 (Health Care Claim Status Request)
X12_004010X093_99_hipaa277_HealCareClaiStatNoti	277 (Health Care Claim Status Notification)
X12_004010X094_99_hipaaA1_278_HealCareServReviInfo X12_004010X094_99_hipaaA3_278_HealCareServReviInfo	278 (Two versions: Health Care Services Review Information and Request for Review/Response to Request)
X12_004010X061_99_hipaa820_PaymOrdeAdvi	820 (Payment Order Remittance Advice)
X12_004010X095_99_hipaa834_BeneEnroAndMain	834 (Benefit Enrollment and Maintenance)
X12_004010X091_99_hipaa835_HealCareClaiPaym	835 (Health Care Claim Payment Advice)
X12_004010X098_99_hipaaQ1_837_HealCareClai X12_004010X097_99_hipaaQ2_837_HealCareClai X12_004010X096_99_hipaaQ3_837_HealCareClai	Health Care Claim (three versions: Professional, Dental, and Institutional)

Table 36 HIPAA 2000 X12 Transactions

File	Transaction Name
X12_004010X092_00_hipaa270_EligCoveOrBeneInqu	270 (Eligibility Coverage or Benefit Inquiry)
X12_004010X092_00_hipaa271_EligCoveOrBeneInfo	271 (Eligibility Coverage or Benefit Information)
X12_004010X093_00_hipaa276_HealCareClaiStatRequ	276 (Health Care Claim Status Request)
X12_004010X093_00_hipaa277_HealCareClaiStatNoti	277 (Health Care Claim Status Notification)
X12_004010X094_00_hipaaA1_278_HealCareServReviInfo X12_004010X094_00_hipaaA3_278_HealCareServReviInfo	278 (Two versions: Health Care Services Review Information and Request for Review/Response to Request)
X12_004010X061_00_hipaa820_PaymOrdeAdvi	820 (Payment Order Remittance Advice)
X12_004010X095_00_hipaa834_BeneEnroAndMain	834 (Benefit Enrollment and Maintenance)
X12_004010X091_00_hipaa835_HealCareClaiPaym	835 (Health Care Claim Payment Advice)
X12_004010X098_00_hipaaQ1_837_HealCareClai X12_004010X097_00_hipaaQ2_837_HealCareClai X12_004010X096_00_hipaaQ3_837_HealCareClai	Health Care Claim (three versions: Professional, Dental, and Institutional)

The NCPDP portion of the HIPAA ETD Library provides request and response transactions for all the HIPAA-approved NCPDP transaction codes, as listed in Table 37. These ETDs are stored in:

<eGate>\server\registry\repository\default\etd\templates\NCPDP

Table 37 NCPDP Transaction Codes

Code	Transaction Name
NCPDP_T51_E1_REQ_EligVeriRequ NCPDP_T51_E1_RESP_EligResp	E1 (Eligibility Verification)
NCPDP_T51_B1_REQ_BillRequ NCPDP_T51_B1_RESP_BillResp	B1 (Billing)
NCPDP_T51_B2_REQ_ReveRequ NCPDP_T51_B2_RESP_ReveResp	B2 (Reversal)
NCPDP_T51_B3_REQ_RebiRequ NCPDP_T51_B3_RESP_RebiResp	B3 (Rebill)
NCPDP_T51_P1_REQ_PrioAuthRequAndBillRequ NCPDP_T51_P1_RESP_PrioAuthRequAndBillResp	P1 (Prior Authorization Request and Billing)
NCPDP_T51_P2_REQ_PrioAuthReveRequ NCPDP_T51_P2_RESP_PrioAuthReveResp	P2 (Prior Authorization Reversal)
NCPDP_T51_P3_REQ_PrioAuthInquRequ NCPDP_T51_P3_RESP_PrioAuthInquResp	P3 (Prior Authorization Inquiry)
NCPDP_T51_P4_REQ_PrioAuthRequOnlyRequ NCPDP_T51_P4_RESP_PrioAuthRequOnlyResp	P4 (Prior Authorization Request Only)
NCPDP_T51_N1_REQ_InfoRepoRequ NCPDP_T51_N1_RESP_InfoRepoResp	N1 (Information Reporting)
NCPDP_T51_N2_REQ_InfoRepoReveRequ NCPDP_T51_N2_RESP_InfoRepoReveResp	N2 (Information Reporting Reversal)
NCPDP_T51_N3_REQ_InfoRepoRebiRequ NCPDP_T51_N3_RESP_InfoRepoRebiResp	N3 (Information Reporting Rebill)
NCPDP_T51_C1_REQ_ContSubsRepoRequ NCPDP_T51_C1_RESP_ContSubsRepoResp	C1 (Controlled Substance Reporting)
NCPDP_T51_C2_REQ_ContSubsRepoReveRequ NCPDP_T51_C2_RESP_ContSubsRepoReveResp	C2 (Controlled Substance Reporting Reversal)
NCPDP_T51_C3_REQ_ContSubsRepoRebiRequ NCPDP_T51_C3_RESP_ContSubsRepoRebiResp	C3 (Controlled Substance Reporting Rebill)

Index

O

P

R

S

T

V

W

X