Pro EDI in BizTalk Server 2006 R2

Electronic Document Interchange Solutions

Mark Beckner

Apress®

Pro EDI in BizTalk Server 2006 R2: Electronic Document Interchange Solutions

ISBN-13: 978-1-59059-935-8

ISBN-10: 1-59059-935-7

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Lead Editor: Jonathan Hassell
Technical Reviewer: Tony Bernard
Editorial Board: Steve Anglin, Ewan Buckingham, Tony Campbell, Gary Cornell, Jonathan Gennick, Jason Gilmore, Kevin Goff, Jonathan Hassell, Matthew Moodie, Joseph Ottinger, Jeffrey Pepper, Ben Renow-Clarke, Dominic Shakeshaft, Matt Wade, Tom Welsh
Project Manager: Kylie Johnston
Copy Editor: Jennifer Whipple
Associate Production Director: Kari Brooks-Copony
Production Editor: Katie Stence
Compositor: Susan Glinert
Proofreader: Patrick Vincent
Indexer: Carol Burbo
Artist: April Milne
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail `orders-ny@springer-sbm.com`, or visit `http://www.springeronline.com`.

For information on translations, please contact Apress directly at 2855 Telegraph Avenue, Suite 600, Berkeley, CA 94705. Phone 510-549-5930, fax 510-549-5939, e-mail `info@apress.com`, or visit `http://www.apress.com`.

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, neither the author(s) nor Apress shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this work.

The source code for this book is available to readers at `http://www.apress.com`.

This book is dedicated to my mother, who taught me to drink tea,
and my father, who brought a TRS-80 into the house.

Contents at a Glance

Contents

Foreword

At the beginning of the sci-fi movie classic *Star Trek II: The Wrath of Khan*, there is a scene where new Star Fleet trainees blow up the bridge of the starship U.S.S. Enterprise, or so they think (it turns out later to simply be a training simulation). Crew members are blown to bits and thrown helplessly around the bridge, including our favorite Star Fleet officers Spock, Sulu, Uhura, and Dr. McCoy. After the training simulation has ended and Admiral Kirk strides off the bridge, he turns to look at the now resurrected Spock and quips "Aren't you dead?"

The same has been said many times about Electronic Data Interchange (EDI). In early 2000, when XML-based e-commerce exploded onto the scene at the height of the Internet boom, nearly all analysts and vendors alike predicted that the rise of XML B2B technologies would lead to the rapid demise of EDI. Turns out we were only half right. While XML B2B traffic has indeed increased significantly over the past seven years, it hasn't in any way killed off the significant mainstream use of EDI by businesses to automate their business transactions (now accounting for more than one-third of the U.S. gross domestic product). In fact, according to a recent study by Forrester Research ("B2B Integration Trends: Message Formats—Alternatives Grow, but EDI Standards Remain the Leading Option for B2B Messaging"), EDI traffic is still the leading form of B2B e-commerce (representing 85%–90% of total B2B traffic, and growing at a rate of 3%–5% year over year).

It turns out that most businesses, having sunk considerable portions of their IT budgets into EDI over the past 30 years, are fairly reluctant to rip and replace proven, stable supply-chain technologies (even if the newer technology options are superior in many ways). Instead, enterprises are looking for ways to extend the life of the existing EDI infrastructure (often embedded tightly into their mission-critical applications). This might include adding on newer real-time event processing (replacing traditional nightly batching), shifting to transmitting EDI over the Internet (to reduce VAN charges), and applying business process management techniques for automating end-to-end supply-chain collaboration processes (to reduce manual exception handling processes and shrink response times). These IT strategies can all be incrementally deployed around existing EDI applications to enhance business visibility and control, while still offering rapid returns on investment.

With the release of BizTalk Server 2006 R2, Microsoft's fifth-generation integration and business process server, we're hoping to make it even easier and more cost-effective for companies to extend the value of their EDI investments. By integrating comprehensive EDI support (both X12 and EDIFACT standards) as part of our unified platform for building next-generation business process applications, Microsoft is helping customers bridge the worlds of their older and proven EDI applications with the latest cutting-edge innovations such as radio frequency identification (RFID) and service-oriented architecture (SOA).

The result? We think this will allow customers to more easily extend the life of their aging applications and drive greater profitability for their businesses. Or as Spock might say, "Live long and prosper."

Burley Kawasaki
Director of Product Management, Connected Systems Division
Microsoft Corporation

About the Author

MARK BECKNER is a technical consultant specializing in business development and enterprise application integration. He runs his own consulting firm, Inotek Consulting Group, LLC, delivering innovative solutions to large corporations and small businesses.

His projects have included engagements with numerous clients throughout the United States and range in nature from mobile application development to complete integration solutions. He is the coauthor of *BizTalk 2006 Recipes: A Problem-Solution Approach* (Apress, 2006), has spoken at a number of venues, including Microsoft's BizTalk Server 2004 in a Service-Oriented World conference, and is certified in Microsoft technologies, including MCSD.NET.

Mark resides with his wife, Sara, and their dog Rosco, the Adventure Mutt, in western Colorado. He can be contacted at `mbeckner@inotekgroup.com` for any questions regarding the concepts in this book, or for advice on integration projects, BizTalk 2006 implementations, or other consulting endeavors.

About the Technical Reviewer

TONY BERNARD currently serves as a senior program manager in the Connected Systems Division of Microsoft's Server and Tools Business. In this role he is responsible for driving the business-to-business (B2B) strategy and feature set for BizTalk Server, Microsoft's enterprise integration platform. Prior to joining Microsoft, Tony served as the vice president of technology for TrueCommerce. In this role he was responsible for all software product development, information systems management, and technical operations at the company.

Prior to joining TrueCommerce, Tony served as senior director of systems architecture for FreeMarkets Inc. In this role, he was responsible for the overall application, data, and technical architecture of FreeMarkets' customer-facing software products. He was an early member of the FreeMarkets team and was with the company for more than six and a half years, helping to guide it through three consecutive years of 300% head count and revenue growth. During that time he held several development, architecture, research and development, and technical management roles. He is also listed as a co-inventor on several patents in the FreeMarkets intellectual property portfolio and prior to his departure was a member of the FreeMarkets/Ariba merger integration team.

Before joining FreeMarkets, Tony was employed as a senior manager at Ernst and Young LLP, where as a member of the information technology consulting practice, he assisted Fortune 500 companies with all aspects of designing, developing, and deploying information technology solutions to meet their business needs. Before that, Tony held several other technology-related positions, including systems coordinator for the Securities Lending group of Mellon Bank; project manager for CIScorp, a systems integration firm; and senior consultant for Accenture.

Tony holds a BS in applied mathematics with a concentration in operations research and management science from Carnegie Mellon University. He has also guest lectured at Duquesne University, the Katz Graduate School of Business at the University of Pittsburgh, and the Heinz School at Carnegie Mellon University.

Acknowledgments

I would like to thank the following individuals for their contributions to my professional life:

- Sara Ann, who allowed me to walk the road that led me to where I am today
- Stephen Roger, who kept me on full-time projects after I went independent
- Brandon Gross, who brought me on as a coauthor on the previous book
- Scott Jones and Brennan O'Reilly, who taught me how to code and how to eat
- Dan Allworth, for placing me on the first BizTalk 2006 R2 EDI project
- Long Duong, the only person I know who actually read my previous work

A great deal of gratitude is owed to the ever widening network of friends and colleagues who are bound for incredible successes, including the guys at Attunix Corporation; Altriva, LLC; Guardian ProStar Inc.; and EMC's Microsoft Practice.

Introduction

The EDI capabilities of BizTalk Server 2006 R2 are a new implementation of an old technology. EDI originated in the 1970s to facilitate the delivery of data between businesses. Rather than diminishing in scope, however, EDI continues to play a major role in commerce today. Because of this, an extensive engine has been added to BizTalk Server to provide for robust handling of EDI transmissions. This book will immerse the reader in working with the new EDI components and demonstrate how to successfully build and deploy a fully functional solution.

Who This Book Is For

The EDI implementation for R2 is unlike any previous incarnation of EDI in BizTalk. Because of this, there is material covered in this book that is equally useful to seasoned BizTalk architects and new developers alike. The intention of the material is to provide all of the detail necessary for any level of developer to envision, build, test, deploy, and support an EDI solution. There are several topics, such as exception handling, that are common to all BizTalk implementations and will prove useful in architecting any solution. The majority of the book, however, from the EDI schemas and party configurations to the new EDI reporting tools, is completely new material not covered in other mediums and will prove to be useful to anyone engaged in a BizTalk 2006 R2 EDI project.

How This Book Is Structured

There are a number of exercises throughout the course of this book illustrating how to work with different components. All chapters with exercises that require code begin with an exercise outlining how to set up the sample code that accompanies that chapter (available for download via the Apress website). All exercises can be worked through from scratch, or the code can simply be deployed and referenced. The following chapters provide end-to-end coverage of building, deploying, and supporting an EDI BizTalk Server 2006 R2 solution:

> **Chapter 1, "EDI Schemas"**: At the foundation of all BizTalk implementations are the schemas. BizTalk Server 2006 R2 ships with thousands of predefined EDI schemas, each representing a different EDI document type. These serve as a starting point and are to be refined—based on the trading partner requirements—to define the final structure of the EDI document. This chapter outlines how to work with these schemas and how to validate their accuracy.
>
> **Chapter 2, "Trading Partner Configuration"**: Using BizTalk parties to implement the details for a trading partner includes setting the header and footer information for an EDI document and specifying details around the delivery and acknowledgement of documents. Party configuration and EDI schemas are both used to define the outcome to the final EDI text document.
>
> **Chapter 3, "Retrieving and Mapping Data"**: Determining the source of data for an EDI document and how that source data is manipulated to create the appropriate data is an essential step in the formation of EDI documents. This chapter outlines how to use the BizTalk SQL Receive Adapter to retrieve data from a source database, and how to use the BizTalk map components to do standard EDI document mapping.

Chapter 4, "EDI and Orchestrations": Many times, if more than a single step is needed for the creation of an EDI document, orchestrations will play an important role in the creation and delivery of EDI documents. One of the most important features covered in this chapter is the concept of a common exception handling infrastructure.

Chapter 5, "Transporting Documents": There are a variety of BizTalk components that may be used to send and receive data. Both AS2 and VAN infrastructures are used in the delivery of EDI documents. This chapter describes how to configure parties to deliver using AS2, how to use the FTP adapters for sending and receiving data, and how to work with a third-party AS2 adapter.

Chapter 6, "Trading Partner Testing": Once a solution has been developed, it must be tested. This generally requires extensive interaction with trading partners to ensure that the data being delivered is valid and complete. This chapter outlines the most common approaches to testing, how to determine where documents are in the overall creation and delivery process, how to work with administrative BizTalk tools, and how to use the new EDI reporting.

Chapter 7, "Deployment and Production Support": This chapter walks through the different deployment options available for BizTalk R2 EDI solutions. It also demonstrates how to keep a solution supported and optimized in a production environment.

Prerequisites

This book demonstrates the functionality of BizTalk Server 2006 R2 EDI; therefore, it is essential that this version of BizTalk is available, and that the EDI components are installed. EDI can be added to a BizTalk installation through the use of the configuration.exe file located in the root Microsoft BizTalk Server 2006 folder. When this utility is run, the final option in the list is BizTalk EDI/AS2 Runtime. The features associated with this option must be successfully installed.

Downloading the Code

The source code for this book is available to readers at `http://www.apress.com` in the Source Code/Downloads section. The exercises in this book assume that this code has been extracted to the C:\ drive. The sample code includes all of the code used in the exercises throughout the book. However, it is possible to work through all of the solutions in this book without the use of this supporting code.

Contacting the Author

Mark Beckner can be contacted via email at `mbeckner@inotekgroup.com`, or via standard mail at the following address:

Inotek Consulting Group, LLC
P.O. Box 4890
Grand Junction, CO 81502

Visit Inotek Consulting online at `http://www.inotekgroup.com`. All inquiries, comments, and feedback are appreciated.

CHAPTER 1

EDI Schemas

This chapter provides an overview of working with EDI documents and BizTalk schemas. It introduces where the information for EDI schemas comes from, how to modify the schema based on a trading partner's requirements, how to validate the schema and document content, and how to deploy the schema once it has been completed. This chapter should be read along with the content of Chapter 2 on trading partner configuration, as it is essential to understand both concepts to be able to define and complete a full EDI schema and create a full EDI instance, with all header, context, detail, and footer (summary) information.

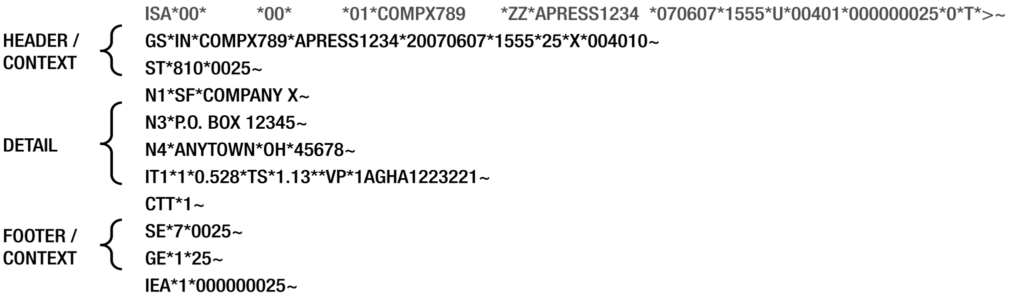
Schemas are considered the foundation of a BizTalk implementation—all other components are dependent on them being in place. The primary steps in defining and working with EDI schemas are the following:

- **Using an EDI implementation guide**: The EDI implementation guide is the starting point for all EDI solutions. It defines all of the documents that a trading partner uses, how those documents are formed (e.g., what segments and data elements are available), and what data is expected to be present when the document is delivered.
- **Determining the right BizTalk schema**: Based on the implementation guide, the appropriate BizTalk schema can be determined. There will likely be multiple schemas in any EDI solution, some specific to trading partners, others shared by multiple trading partners.
- **Modifying the BizTalk schema based on the trading partner**: Once the correct schema has been determined and added to a BizTalk project, the next step is to modify the schema to fit the requirements of the individual trading partner. Often this involves removing unneeded nodes (segments), modifying field lengths, and determining whether data elements are mandatory or optional.
- **Promoting fields**: Field promotion becomes a factor in implementations that will be using send port filtering or orchestration logic based on the content of an EDI document.
- **Validating and generating EDI instances**: During schema development, existing EDI instances will need to be validated and new file instances may need to be created. Validation generally involves taking a known instance of an EDI document that works for a given trading partner and validating it against the BizTalk schema. Based on the results, the schema may need to be further revised.

Schema Overview

All documents processed in BizTalk Server, whether they are EDI or otherwise, adhere to a schema. There are numerous EDI documents in existence, ranging from invoices and bills of lading to functional and technical acknowledgements, and all of these documents have their own schema. On top of that, each EDI trading partner may have its own variation on an individual EDI document: Company A may expect slightly different information in a slightly different format for its invoice than Company X does. Each of these unique documents is defined by a different BizTalk schema (also known as an *XSD*).

Figure 1-1 shows an example of an EDI document. The document has been split into the different parts that will be referred to throughout the course of this book.



```
ISA*00*          *00*          *01*COMPX789       *ZZ*APRESS1234     *070607*1555*U*00401*000000025*0*T*>~
GS*IN*COMPX789*APRESS1234*20070607*1555*25*X*004010~
ST*810*0025~
N1*SF*COMPANY X~
N3*P.O. BOX 12345~
N4*ANYTOWN*OH*45678~
IT1*1*0.528*TS*1.13**VP*1AGHA1223221~
CTT*1~
SE*7*0025~
GE*1*25~
IEA*1*000000025~
```

HEADER / CONTEXT
DETAIL
FOOTER / CONTEXT

Figure 1-1. *Diagram of an EDI document representing an X12 810 invoice*

Because of the nature of EDI implementations, document structure, and trading partner requirements, it is common practice to have one BizTalk schema for every EDI document type per trading partner. On occasion, however, a schema may be shared between multiple trading partners; in this case, defining a “common schema project” can be helpful. Figure 1-2 illustrates a sample BizTalk project schema structure for two trading partners that both have unique invoice schemas and a shared bill of lading schema. One of the trading partners also has its own unshared purchase order schema.

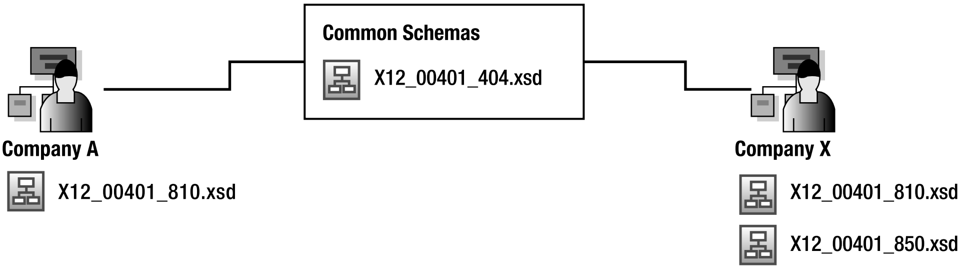


Figure 1-2. *Schema project structure*

Exercise 1-1 walks through setting up the sample code provided for this chapter. This chapter uses the files located in C:\Apress.Integration\Chapter 1.

Exercise 1-1. Preparing the Solution Files

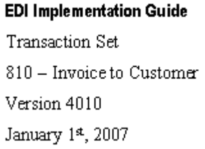
Use the steps in this exercise to prepare the environment on the local machine without having to work through each of the exercises individually:

1. Make sure that the Chapter1 sample files have been extracted and placed on the C:\ drive.
2. The components can be deployed in a single package by importing the Chapter1.MSI file available in C:\Apress.Integration\Chapter 1\BizTalk Application. Open BizTalk Administration Console and right-click the Applications folder. Select Import ➤ MSI File. This will create the EDI.Demonstration.Chapter1 BizTalk application and all of the send and receive ports used in this chapter. It will also import the schemas. Alternatively, there are two solutions that can be deployed manually (rather than using the MSI):
   a. Open the solution contained in C:\Apress.Integration\Chapter 1 called Apress.Integration.EDI810.CompanyX.sln. This solution contains two items: a modified version of the X12 810 schema, tailored to Company X, and a property schema containing information about promoted fields. Deploy this by right-clicking the solution in Visual Studio and selecting Deploy.
   b. Open the solution called Apress.Integration.EDIFACT.ServiceExtension.sln. This solution contains the X12 service schema with modifications. This can also be deployed by right-clicking the solution in Visual Studio and selecting Deploy.

EDI Implementation Guide and Document Layout

The single most important component in determining information about the trading partner is the EDI implementation guide. An implementation guide is available from virtually any company using EDI. This guide outlines all of the documents being exchanged and all of the segments and fields that are used within each document. Implementation guides are generally comprised of the following sections:

- **Transaction set specification**: The transaction set specification is very similar to a title page—it simply defines what document is being covered and gives some high-level information about that document. Figure 1-3 shows a sample transaction set specification page.
- **Introduction**: The introduction is an overview of all the segments that are used within the given transaction set. The segments are defined and it is noted whether they are mandatory or optional and how many times they are expected to appear in a document; any notes associated with the segment are outlined. The introduction is comprised of three parts: header, detail, and footer.
- **Segment definitions**: After the introduction, each segment is presented, including all of the data elements (or *fields*) that are available on the segment, the length, and whether they are mandatory or optional. Any notes associated with each data element are defined in the segment definitions.



EDI Implementation Guide

Transaction Set

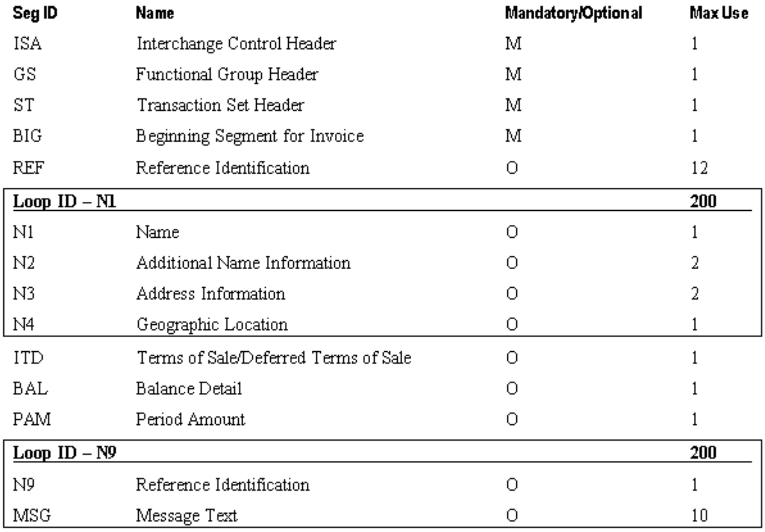
810 – Invoice to Customer

Version 4010

January 1st, 2007

Figure 1-3. *The transaction set specification page in the implementation guide*

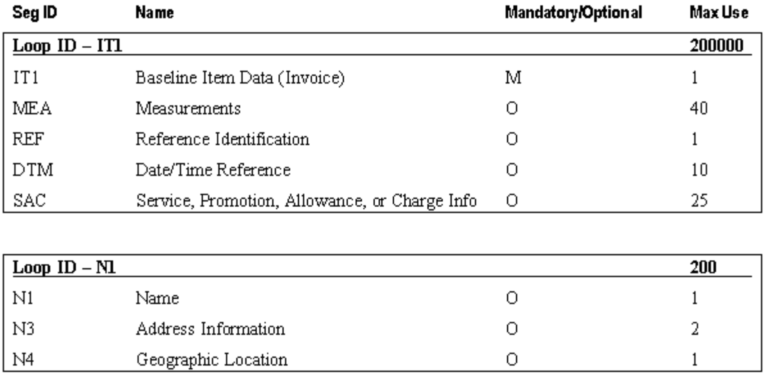
The following figures are representations of what should be found in the introduction of an EDI implementation guide. Figure 1-4 shows the header segments, whether the segment is mandatory or optional, and how many times the segments will appear in the document. Note that there are two looping segments, N1 and N9, both appearing up to 200 times in a given EDI instance.



Seg ID	Name	Mandatory/Optional	Max Use
ISA	Interchange Control Header	M	1
GS	Functional Group Header	M	1
ST	Transaction Set Header	M	1
BIG	Beginning Segment for Invoice	M	1
REF	Reference Identification	O	12
Loop ID – N1			**200**
N1	Name	O	1
N2	Additional Name Information	O	2
N3	Address Information	O	2
N4	Geographic Location	O	1
ITD	Terms of Sale/Deferred Terms of Sale	O	1
BAL	Balance Detail	O	1
PAM	Period Amount	O	1
Loop ID – N9			**200**
N9	Reference Identification	O	1
MSG	Message Text	O	10

Figure 1-4. *Header segment in the introduction*

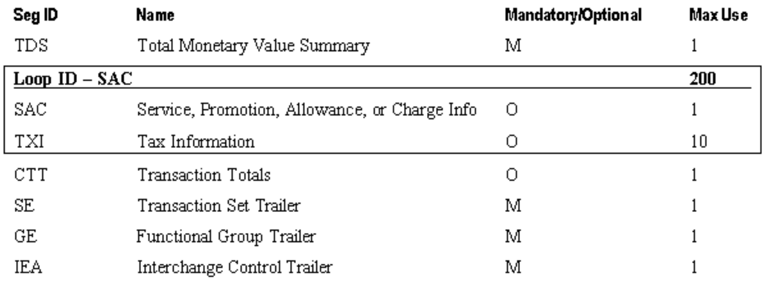
Figure 1-5 shows all of the detail segments and whether they are optional and what their usage is. The IT1 loop can appear one to many times, whereas the N1 loop can appear zero to many times.



Seg ID	Name	Mandatory/Optional	Max Use
Loop ID – IT1			**200000**
IT1	Baseline Item Data (Invoice)	M	1
MEA	Measurements	O	40
REF	Reference Identification	O	1
DTM	Date/Time Reference	O	10
SAC	Service, Promotion, Allowance, or Charge Info	O	25
Loop ID – N1			**200**
N1	Name	O	1
N3	Address Information	O	2
N4	Geographic Location	O	1

Figure 1-5. *Detail segment*

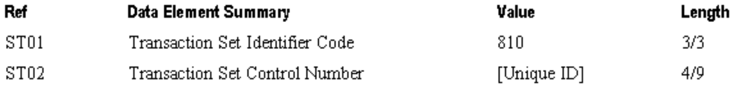
The final section of the implementation guide is the definition of the footer (or summary) segments. These segments are shown in Figure 1-6.



Seg ID	Name	Mandatory/Optional	Max Use
TDS	Total Monetary Value Summary	M	1
Loop ID – SAC			**200**
SAC	Service, Promotion, Allowance, or Charge Info	O	1
TXI	Tax Information	O	10
CTT	Transaction Totals	O	1
SE	Transaction Set Trailer	M	1
GE	Functional Group Trailer	M	1
IEA	Interchange Control Trailer	M	1

Figure 1-6. *Footer (summary) segment*

Figure 1-7 shows a sample version of a segment definition from an EDI implementation guide. The introduction states that the ST header (transaction set header) segment will be in the 810 EDI document, that it is a mandatory segment, and that it will appear one time. The segment definition for the ST segment shows the specific implementation information for that segment. For example, the first field, Transaction Set Identifier Code, has a constant value of 810 and must be exactly three characters long. The second field, ST02, is a number that will be automatically set by BizTalk, ensuring that it is a unique value.



Ref	Data Element Summary	Value	Length
ST01	Transaction Set Identifier Code	810	3/3
ST02	Transaction Set Control Number	[Unique ID]	4/9

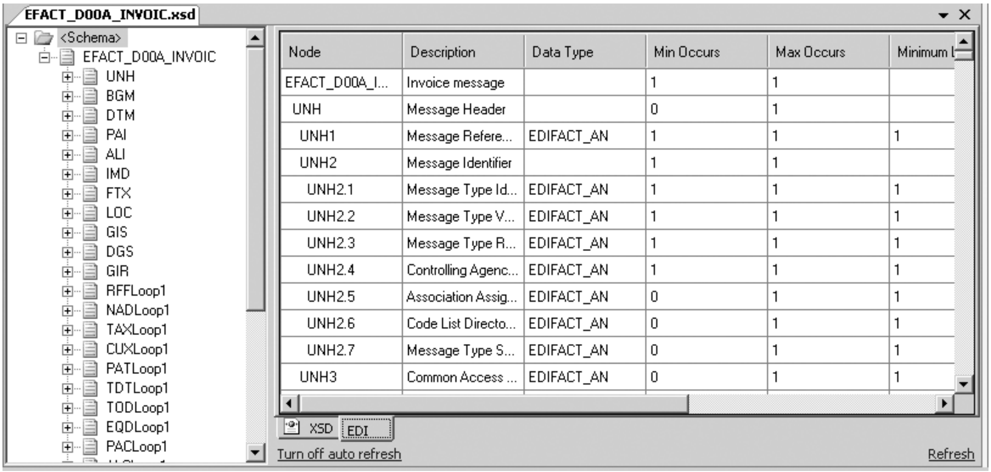
Figure 1-7. *ST segment*

Note The ST segment is different from the majority of segments in that it must be part of the EDI X12 schemas, but at the same time it has some values that are set by the EDI send pipeline. Most segments are either completely set by the pipeline (such as the ISA segment via trading partner configuration) or are completely handled in mapping.

Schema Development and Deployment

BizTalk Server 2006 R2 is packaged with thousands of EDI schemas that can be used as starting points for a BizTalk EDI implementation (accessing these schemas is shown in Exercise 1-2, later in this section). These schemas are designed to contain a superset of all the nodes that may be required by any trading partner for the specified document type. There will frequently be fields that need to be modified (i.e., changing a “required” field to “optional,” changing the maximum length of a field, or changing the type of data that can be stored in a field), and there will frequently be the need to eliminate a number of the nodes that do not pertain to the trading partner in question.

EDI schemas have an additional tab available to them in the Visual Studio Editor, shown in Figure 1-8. This tab gives quick access to some of the most prominent properties that are set for EDI data elements. Aside from this tab, working with EDI schemas is no different than working with any other type of BizTalk schema.

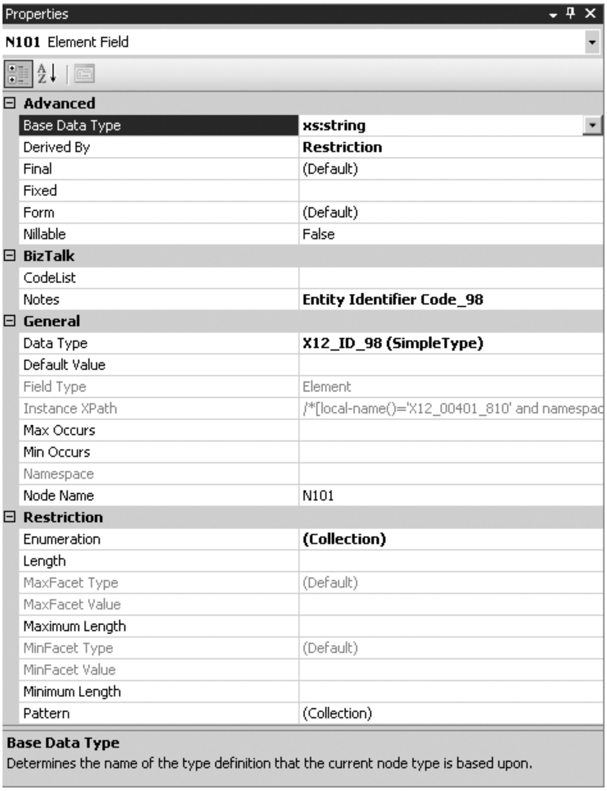


Node	Description	Data Type	Min Occurs	Max Occurs	Minimum L
EFACT_D00A_I...	Invoice message		1	1	
UNH	Message Header		0	1	
UNH1	Message Refere...	EDIFACT_AN	1	1	1
UNH2	Message Identifier		1	1	
UNH2.1	Message Type Id...	EDIFACT_AN	1	1	1
UNH2.2	Message Type V...	EDIFACT_AN	1	1	1
UNH2.3	Message Type R...	EDIFACT_AN	1	1	1
UNH2.4	Controlling Agenc...	EDIFACT_AN	1	1	1
UNH2.5	Association Assig...	EDIFACT_AN	0	1	1
UNH2.6	Code List Directo...	EDIFACT_AN	0	1	1
UNH2.7	Message Type S...	EDIFACT_AN	0	1	1
UNH3	Common Access ...	EDIFACT_AN	0	1	1

Figure 1-8. *The BizTalk schema EDI tab in Visual Studio*

Figure 1-9 shows all of the properties that can be set in an EDI schema. The most important properties to pay attention to are as follows:

- **Base Data Type and Data Type**: These properties indicate what type of data is expected to be present in the field. There are standard types, such as string and int, but there are also a large number of EDI-specific fields, such as X12_ID. Each of the EDI-specific fields has different properties associated with it (such as enumerations) and will automatically set these values when selected. Generally, the value of the element is set appropriately by default. If a trading partner needs a different type of data in this field, it is easiest to simply set this to xs:string, which allows anything to be present. Of course, by doing this, any EDI validation will be removed from this field.
- **Notes**: This is a description property that states what the field represents. It can be helpful during development and testing.
- **Max and Min Occurs**: These represent how many times the node or element can occur in the document. Occasionally it may be necessary to change these values, especially when a node is marked as “mandatory,” but the trading partner does not expect it to be present on every document.
- **Length, Maximum Length, and Minimum Length**: These three fields dictate how long the data in the field can be. It is frequently necessary to modify these values for a given element, since different trading partners expect different lengths of data.
- **Enumeration**: This field contains an array of values that can be entered and compared against a schema. If the value in the field does not match a value in the enumeration, it will not be valid. See Exercise 1-3 (in the “X12 and EDIFACT Schemas” section, later in this chapter) on changing these values.



Properties

N101 Element Field

⊟ **Advanced**	
Base Data Type	**xs:string**
Derived By	**Restriction**
Final	(Default)
Fixed	
Form	(Default)
Nillable	False
⊟ **BizTalk**	
CodeList	
Notes	**Entity Identifier Code_98**
⊟ **General**	
Data Type	**X12_ID_98 (SimpleType)**
Default Value	
Field Type	Element
Instance XPath	/*[local-name()='X12_00401_810' and namespac
Max Occurs	
Min Occurs	
Namespace	
Node Name	N101
⊟ **Restriction**	
Enumeration	**(Collection)**
Length	
MaxFacet Type	(Default)
MaxFacet Value	
Maximum Length	
MinFacet Type	(Default)
MinFacet Value	
Minimum Length	
Pattern	(Collection)

Base Data Type
Determines the name of the type definition that the current node type is based upon.

Figure 1-9. *EDI schema element properties*

Exercise 1-2 demonstrates how to access the default EDI schemas that ship with BizTalk Server 2006 R2.

Exercise 1-2. Accessing Default BizTalk EDI Schemas

There are a large number of EDI schemas that ship with BizTalk Server 2006 R2 and are made available when the EDI components are installed. They can be accessed as follows:

1. Browse to the directory c:\Program Files\Microsoft BizTalk Server 2006\XSD_Schema\EDI.
2. Before schemas are extracted, there is a single file: MicrosoftEdiXSDTemplates.exe. This is a compressed file that contains thousands of EDI schemas. Run this file to extract all of the schemas to the local path. Figure 1-10 shows the location of this file in Windows Explorer.