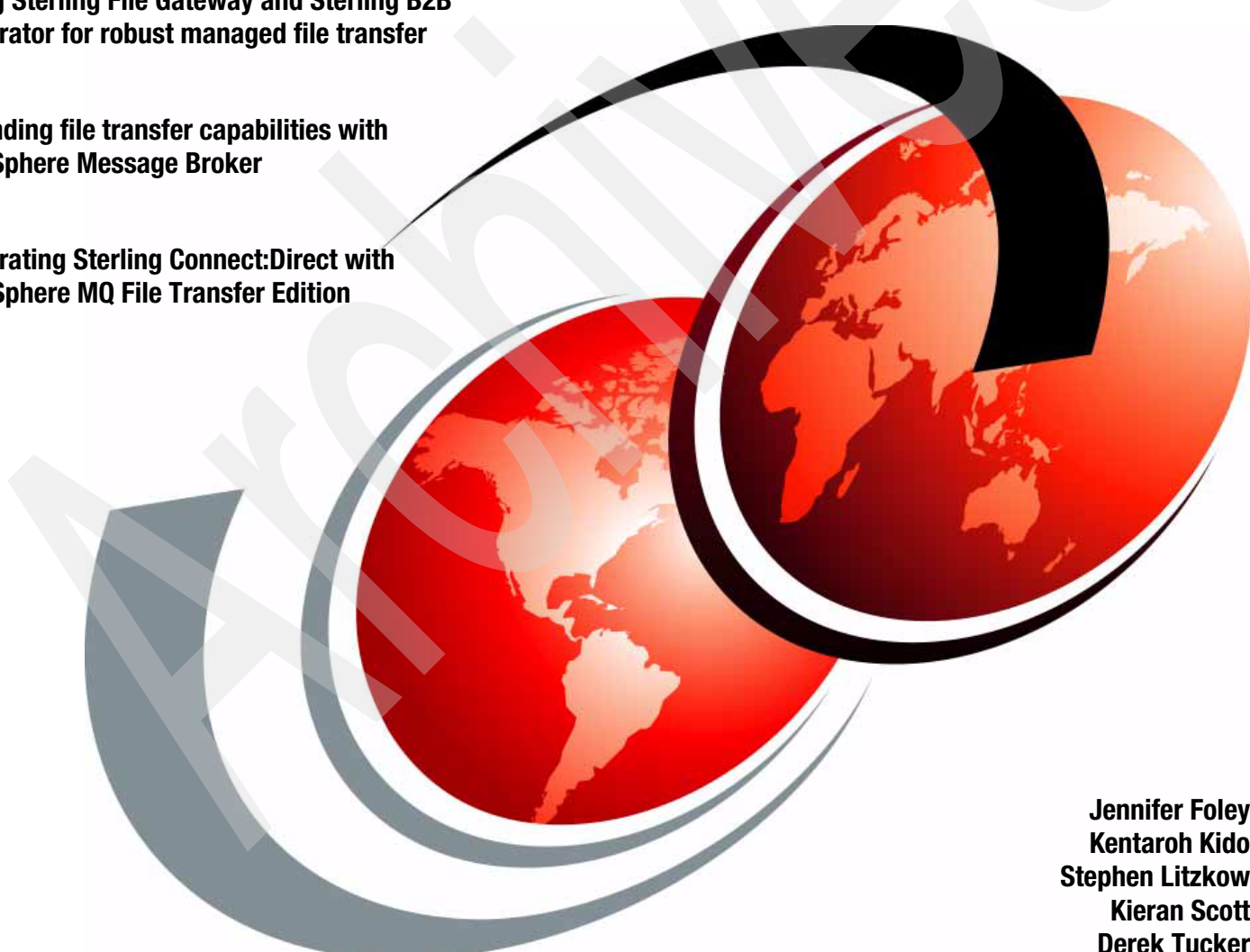


IBM Sterling Managed File Transfer Integration with WebSphere Connectivity for a Multi-Enterprise Solution

Using Sterling File Gateway and Sterling B2B Integrator for robust managed file transfer

Extending file transfer capabilities with WebSphere Message Broker

Integrating Sterling Connect:Direct with WebSphere MQ File Transfer Edition



Jennifer Foley
Kentaroh Kido
Stephen Litzkow
Kieran Scott
Derek Tucker

Redbooks



International Technical Support Organization

**IBM Sterling Managed File Transfer Integration with
WebSphere Connectivity for a Multi-Enterprise
Solution**

March 2011

Archived

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

Archived

First Edition (March 2011)

This book applies to IBM® WebSphere Message Broker V7.0.0.1, IBM® WebSphere MQ V7.0.1, IBM® WebSphere MQ File Transfer Edition V7.0.3, IBM® Sterling File Gateway V2.1, IBM® Sterling B2B Integrator V5.1, and IBM® Sterling Connect:Direct V4.

© Copyright International Business Machines Corporation 2011. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	ix
Trademarks	x
Preface	xi
The team who wrote this book	xi
Now you can become a published author, too!	xiii
Comments welcome	xiii
Stay connected to IBM Redbooks	xiii
Chapter 1. File transfer concepts, technologies, and best practices	1
1.1 Introduction	2
1.1.1 Basic FTP	2
1.1.2 Managed file transfer	2
1.2 Brief history and challenges of file transfer	2
1.3 Overcoming FTP challenges with managed file transfer	5
1.3.1 Improved reliability	5
1.3.2 Improved security	5
1.3.3 Improved auditability and visibility	6
1.3.4 Improved flexibility	6
1.3.5 Cost effectiveness	6
1.4 Managed file transfer best practices	6
1.4.1 Intra-enterprise managed file transfer best practices	7
1.4.2 Multi-enterprise managed file transfer best practices	8
1.5 Comparing intra-enterprise and multi-enterprise file transfers	9
1.5.1 Control	10
1.5.2 Authentication and data validation	10
1.5.3 Network security	11
1.6 Recent additions to the managed file transfer portfolio	11
1.7 Who should read this book	12
Chapter 2. File transfer products overview	13
2.1 Sterling Connect:Direct	14
2.1.1 Sterling Connect:Direct additional features	14
2.1.2 Scalability	16
2.1.3 Capabilities	17
2.1.4 Architectural overview	18
2.1.5 Connect:Direct process language	20
2.2 IBM Sterling B2B Integrator	24
2.2.1 Capabilities	25
2.2.2 Terminology	26
2.3 IBM Sterling File Gateway	27
2.3.1 How Sterling B2B Integrator and Sterling File Gateway work together	27
2.3.2 Capabilities	28
2.3.3 Terminology	29
2.4 IBM Sterling Secure Proxy	31
2.5 WebSphere MQ File Transfer Edition	33
2.5.1 Architecture of WebSphere MQ File Transfer Edition	34
2.5.2 Using Apache Ant	38
2.5.3 Using file transfer pre-processing and post-processing tasks	39

2.5.4 Using WebSphere MQ Advanced Message Security	39
2.6 WebSphere Message Broker	39
2.6.1 Message flows with WebSphere Message Broker	39
2.6.2 Runtime architecture of WebSphere Message Broker	40
2.6.3 Developing message flows with the WebSphere Message Broker Toolkit	40
2.6.4 Deploying message flow applications	41
2.6.5 Administration with WebSphere Message Broker Explorer	41
Chapter 3. Scenario topology overview	43
3.1 An introduction to the scenarios used in this book	44
3.1.1 Internal use of Sterling Connect:Direct and WebSphere MQ File Transfer Edition	44
3.1.2 Using Sterling Connect:Direct for multi-enterprise transfers	44
3.1.3 Multi-Enterprise file transfer using Sterling Connect:Direct, Sterling File Gateway, and WebSphere MQ File Transfer Edition	45
3.1.4 Integrating multi-enterprise transfers with an enterprise service bus	45
3.2 Scenario architecture	46
3.2.1 The external partner: Company A	47
3.2.2 The protected network: Company B	47
3.2.3 DMZ	49
Chapter 4. Managed file transfer within an enterprise	51
4.1 Solution overview	52
4.1.1 Using Sterling Connect:Direct and WebSphere MQ File Transfer Edition	52
4.1.2 Business value	53
4.2 Scenario details	55
4.2.1 Solution components	55
4.2.2 The Sterling Connect:Direct to Sterling Connect:Direct scenario	57
4.2.3 Sterling Connect:Direct push to WebSphere MQ File Transfer Edition	58
4.2.4 Sterling Connect:Direct pulling from WebSphere MQ File Transfer Edition	59
4.2.5 WebSphere MQ File Transfer Edition pushing to Sterling Connect:Direct	60
4.2.6 WebSphere MQ File Transfer Edition pulling from Sterling Connect:Direct	61
4.2.7 Protocols	61
4.2.8 Security	62
4.3 Configuring the solution components	65
4.3.1 Software prerequisites	65
4.3.2 Configuration prerequisites	65
4.3.3 Configuring Sterling Connect:Direct on SysD	66
4.3.4 Configuring Sterling Connect:Direct on SysE	74
4.3.5 Configuring WebSphere MQ File Transfer Edition	75
4.4 Testing the flows	81
4.4.1 Sterling Connect:Direct push to Sterling Connect:Direct	81
4.4.2 Sterling Connect:Direct push file to WebSphere MQ File Transfer Edition	83
4.4.3 Sterling Connect:Direct pull file from WebSphere MQ File Transfer Edition	84
4.4.4 WebSphere MQ File Transfer Edition push to Sterling Connect:Direct	85
4.4.5 WebSphere MQ File Transfer Edition pull to Sterling Connect:Direct	87
4.5 Troubleshooting tips	90
Chapter 5. External transfers using IBM Sterling Connect:Direct and IBM Sterling File Gateway	91
5.1 Solution overview	92
5.1.1 Appropriate use of the scenario	92
5.1.2 Business value	92
5.1.3 Sterling File Gateway features	93
5.2 Scenario details	94

5.2.1	Solution components	95
5.2.2	Sterling Connect:Direct to Sterling Connect:Direct using Sterling Secure Proxy	98
5.2.3	External Sterling Connect:Direct push to Sterling File Gateway using Sterling Secure Proxy	99
5.2.4	Internal Sterling Connect:Direct push to Sterling File Gateway using Sterling Secure Proxy	100
5.2.5	Protocols	101
5.2.6	Security	101
5.2.7	Software prerequisites	103
5.2.8	Configuring the solution components	103
5.2.9	Configuration prerequisites	103
5.2.10	Configuring Sterling Connect:Direct on SysA	104
5.2.11	Configuring the Connect:Direct for Linux node SysE_CD	115
5.2.12	Installing and configuring the Connect:Direct server adapter	116
5.2.13	Configuring the proxy	128
5.2.14	Configuring Sterling File Gateway	129
5.3	Testing the flows	144
5.3.1	Sterling Connect:Direct push file to Sterling Connect:Direct	145
5.3.2	External Sterling Connect:Direct push to Sterling File Gateway to internal Sterling Connect:Direct	151
5.3.3	Internal Sterling Connect:Direct push to Sterling File Gateway, to external Sterling Connect:Direct using Sterling Secure Proxy	157
5.3.4	Creating the route in Sterling File Gateway	157
5.4	Troubleshooting	166
Chapter 6. External Transfers with Protocol Switching between IBM Sterling Connect:Direct and WebSphere MQ File Transfer Edition via Sterling File Gateway		
6.1	Solution overview	168
6.1.1	Appropriate use	168
6.1.2	Business value	169
6.2	Scenario details	171
6.2.1	Solution components	172
6.2.2	Inbound: Sterling Connect:Direct to WebSphere MQ File Transfer Edition	176
6.2.3	Outbound: WebSphere MQ File Transfer Edition to Sterling Connect:Direct	177
6.2.4	Protocols	178
6.2.5	Security	178
6.3	Configuring the solution components	181
6.3.1	Software prerequisites	182
6.3.2	Configuration prerequisites	182
6.3.3	Sterling B2B Integrator and Sterling File Gateway customization	182
6.3.4	Configuring Sterling Connect:Direct on SysA	184
6.3.5	Installing the Connect:Direct server adapter on Sterling B2B Integrator	193
6.3.6	Configuring the proxy	193
6.3.7	Configuring Sterling B2B Integrator to use the WebSphere MQ Adapter	193
6.3.8	Configuring FTP Server adapter in Sterling B2B Integrator	194
6.3.9	Configuring WebSphere MQ File Transfer Edition bridge agent	196
6.3.10	Creating custom WebSphere MQ File Transfer Edition protocol	199
6.3.11	Creating a WebSphere MQ reply queue	207
6.3.12	Configuring Sterling File Gateway	208
6.4	Testing the flows	229
6.4.1	Inbound scenario	229
6.4.2	Outbound scenario	238

6.5 Troubleshooting tips	243
Chapter 7. External transfers using IBM WebSphere Message Broker and IBM Sterling File Gateway	245
7.1 Solution overview	246
7.1.1 Appropriate use	246
7.1.2 Business value	246
7.2 Scenario details	248
7.2.1 Solution components	249
7.2.2 Inbound file transfer flow	253
7.2.3 Outbound file transfer flow	254
7.2.4 Protocols	257
7.2.5 Security	257
7.3 Configuring the solution components	260
7.3.1 Software prerequisites	261
7.3.2 Configuration prerequisites	261
7.3.3 Creating the protocol bridge agent	261
7.3.4 Configuring a broker and execution group	266
7.3.5 WebSphere MQ File Transfer Edition with WebSphere Message Broker	271
7.3.6 Creating message flows with WebSphere MQ File Transfer Edition nodes	274
7.3.7 Creating a community in Sterling File Gateway	284
7.3.8 Creating routing channel templates in Sterling File Gateway	288
7.3.9 Enabling WebSphere MQ File Transfer Edition in Sterling File Gateway	288
7.3.10 Modifying FirstCommunity in Sterling File Gateway	288
7.3.11 Creating a listener queue for Sterling B2B Integrator	288
7.3.12 Setting up a trading partner for WebSphere Message Broker	289
7.3.13 Creating a trading partner for myFileGateway	296
7.3.14 Configuring Sterling B2B Integrator for SFTP communication	299
7.3.15 Creating the trading partner for SFTP	310
7.3.16 Creating an inbound routing channel	314
7.3.17 Creating an outbound routing channel	317
7.3.18 Importing key certificate files in Sterling B2B Integrator for HTTPS	319
7.3.19 Configuring Sterling B2B Integrator and myFileGateway for HTTPS	324
7.4 Testing the flows	333
7.4.1 Testing the flow sending an output file over SFTP	334
7.4.2 Testing the scenario downloading a file using myFileGateway	341
7.5 Troubleshooting tips	345
Appendix A. Configuration of WebSphere MQ File Transfer Edition	347
Overview	348
Configuring WebSphere MQ	349
Creating the queue managers	349
Creating the queue manager objects	351
Configuring WebSphere MQ File Transfer Edition	356
Defining the coordination, command, and agent queue manager and WebSphere MQ File Transfer agents	356
Starting the agent	362
Appendix B. Building the WebSphere Message Broker flow	365
Overview of the flow modifications	366
Modifying the sample	367
Creating the ESQL module	374
Other configuration tasks	376
Example files to test the flow	376

Appendix C. Troubleshooting	379
Sterling File Gateway and Sterling B2B Integrator	380
WebSphere MQ File Transfer Edition	384
Sterling Connect:Direct	388
Check ports	395
WebSphere Message Broker tips	396
Transfers do not appear in WebSphere MQ File Transfer Edition Explorer	396
Appendix D. Sample files	401
Sample Ant scripts	402
Push_to_CD.xml script	402
Pull_from_CD.xml script	404
InvokeCD.xml script	406
Sample files used in WebSphere MQ File Transfer Edition within Sterling File Gateway ..	406
SFGFTECreateTransfer.xslt file	406
CustomFileGatewayDeliveryFTE.bmpl file	408
Customer_overrides.properties file	419
AFTEExtensionsCustomer.properties source file	421
AFTEExtensionsCustomer.xml source file	422
Appendix E. Additional material	425
Locating the web material	425
Using the web material	425
Downloading and extracting the web material	426
Related publications	427
IBM Redbooks	427
Online resources	427

Archived

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.


COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the web at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

CICS®	IMS™	RETAIN®
DataPower®	MQSeries®	System z®
DB2®	OS/400®	Tivoli®
eServer™	Redbooks®	WebSphere®
i5/OS®	Redpaper™	z/OS®
IBM®	Redbooks (logo)  ®	z/VSE™

The following terms are trademarks of other companies:

Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Preface

This IBM® Redbooks® publication describes how with the acquisition of Sterling Commerce, an IBM company, IBM has enhanced its managed file transfer portfolio consisting of MQ File Transfer Edition and IBM Sterling Business Integration Suite. The Sterling Business Integration Suite consists of IBM Sterling File Gateway and IBM Sterling Connect:Direct. Sterling Commerce, an IBM company, transforms and optimizes your business collaboration network by improving business agility, efficiency, and performance. This is done with a comprehensive, yet modular, suite of solutions that solve integration challenges both inside and outside an enterprise. Sterling Commerce, an IBM company, has been recognized as the market leader in providing B2B integration and managed file transfer solutions.

This book is intended for those organizations that find themselves wanting to trade data in a secure, reliable, and auditable way across protocols both intra-enterprise and multi-enterprise. Architects, system administrators, system programmers, and developers looking to build a managed file transfer solution should read this book. With a recent acquisition, many organizations can find themselves with various options to move files across multi-enterprises. This book shows how to combine IBM Sterling Connect:Direct, IBM Sterling File Gateway, and MQ File Transfer Edition to provide a seamless transport.

The team who wrote this book

This book was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Jennifer Foley is a WebSphere® on System z® IT Specialist based in Dallas, TX. She works with customers to grow and modernize their WebSphere portfolio on System z. Her current product expertise is in the WebSphere Connectivity and Application Infrastructure portfolios. She has developed customer demos available through the IBM DEMOcentral organization and written technical documents for internal use within IBM. She has a Bachelor of Science degree from the University of Oklahoma in computer science with a minor in mathematics.

Kentaroh Kido is an IT Specialist with IBM Systems Engineering Co., Ltd. in Japan. He has three years of experience in designing and implementing messaging infrastructure and has been in technical support for messaging products. He specializes in WebSphere MQ, WebSphere MQ File Transfer Edition, and WebSphere Message Broker on open platform. He holds a master's degree in computer science from Tsukuba University, Japan.

Stephen Litzkow is a Software Engineer with the IBM Software Group in the United States working primarily with IBM Sterling Connect:Direct. He has 15 years of experience in application development and system administration. He also teaches IBM classes about IBM Sterling Connect:Direct and cryptography.

Kieran Scott is a Senior IT Specialist for the Federated Integration Test team based in Hursley, UK. In this role, Kieran works as a developer, tester, architect, and author. Kieran specializes in the area of scenario integration testing, integrating many different IBM products into cross-platform scenarios and documenting best practices. His expertise in this area led to him becoming involved in early investigation work integrating Sterling Managed File Transfer solutions with the IBM portfolio, including WebSphere MQ File Transfer Edition and

WebSphere Message Broker. The scenarios described in this book build on the early prototypes that he developed during that time.

Derek Tucker is a Senior Software Engineer with the IBM Software Group in the United States and has been a member of the Sterling File Gateway development team since 2008. He has 15 years of experience developing and integrating enterprise applications across a variety of industries. He holds a master's degree in computer science from the University of Colorado, Boulder.



Figure 1 (Left to right) Margaret Ticknor, Kieran Scott, Jennifer Foley, Kentaroh Kido, Steve Litzkow

Thanks to the following people for their contributions to this project:

Margaret Ticknor (Project Leader), Carla Sadtler, Tamikia Barrow
International Technical Support Organization, Raleigh Center

Andy Gibbs
IBM Hursley

Beverly Hrablook, Dee Milam, Carol Otto, Brian Persichitte, Laura Poepelman, Sandy Schriever, Gary White, Robert Zebian
IBM US

Dina Maiorana
Click4Care US

Mick Lickman
IBM UK

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at: ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!


We want our books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- ▶ Use the online **Contact us** review Redbooks form found at:
ibm.com/redbooks
- ▶ Send your comments in an email to:
redbooks@us.ibm.com
- ▶ Mail your comments to:
IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- ▶ Find us on Facebook:
<http://www.facebook.com/IBMRedbooks>
- ▶ Follow us on Twitter:
<http://twitter.com/ibmredbooks>
- ▶ Look for us on LinkedIn:
<http://www.linkedin.com/groups?home=&gid=2130806>
- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:
<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>
- ▶ Stay current on recent Redbooks publications with RSS Feeds:
<http://www.redbooks.ibm.com/rss.html>

Archived



File transfer concepts, technologies, and best practices

This chapter presents an overview of how businesses are using file transfer protocols to move files within the enterprise (internally) and between enterprises (externally). It also provides a brief historical view of file transfer technologies. This chapter compares intra-enterprise file transfer to multi-enterprise file transfers by examining the similarities and differences that make them separate technologies.

We also discuss the challenges surrounding the multi-enterprise use of the File Transfer Protocol (FTP), how managed file transfer can overcome these issues, and what best practices businesses to consider when looking to move files within and out of an enterprise.

File transfer definitions: For the purpose of this book, *intra-enterprise* describes file transfers that take place inside an organization. *Multi-enterprise* is defined as file transfers that take place across different organizations, often crossing through a demilitarized zone.

1.1 Introduction

For many organizations, the exchange of files between business systems remains a common and important integration methodology. Files are the simplest unit of data to exchange and often represent the lowest common denominator for an enterprise infrastructure.

Although the exchange of files is conceptually simple, doing so in the enterprise is a challenge to manage and audit. This difficulty is brought into clear focus when an organization needs to perform file transfer with another business organization, perhaps using a different physical network, with different security requirements, and perhaps a different governance or regulatory framework.

Despite an abundance of technologies for communicating across systems, including web services, Web 2.0, and enterprise messaging, file transfer remains a common method of integrating business systems.

1.1.1 Basic FTP

File transfer has a long history. There are many existing tools that support it in some form. The simplest and best known technology for file transfer is the File Transfer Protocol (FTP), which was first made available in UNIX® systems in the 1970s. Today, the broad availability of FTP on almost all platforms makes it an easy choice when the need to exchange files arises. However, performing mission-critical file transfers using FTP does have issues with limited reliability, recoverability, security, and auditability.

1.1.2 Managed file transfer

Managed file transfer addresses the need that organizations have to configure, track, and audit file transfer activity consistently. Typically, an organization using managed file transfer has the following needs:

- ▶ **Auditability:** File transfer activity must be logged so that administrators can determine where each file is sent and when the transfer occurred. The transfer log needs to be centrally accessible.
- ▶ **Security:** File transfer requests require acceptance from authorized people or application systems.
- ▶ **Recoverability and reliability:** Network or other errors that interrupt a transfer must not cause the transfer to be abandoned or partial files to be received.
- ▶ **Platform connectivity:** File transfers must span multiple platforms.

1.2 Brief history and challenges of file transfer

The most commonly known network protocols are Transmission Control Protocol and the Internet Protocol (TCP/IP), which were the first two networking protocols defined to the Internet Protocol Suite. The TCP/IP model comprises four layers:

- ▶ Application
- ▶ Transport
- ▶ Internet
- ▶ Link

Historically, as the distributed computing model grew, the enterprise use of TCP/IP grew to support the local area networks and the first ventures into internet computing.

FTP was first introduced in 1972. Since then, FTP has been helping companies move volumes of single and batched files between distributed servers. As other communication protocols have been introduced, many protocols for moving files have emerged. Other application layer technologies, such as Hypertext Transfer Protocol (HTTP), HTTP over Secure Sockets Layer (SSL) (HTTPS), Simple Asynchronous File Transfer (SAFT), Secure Copy (SCP), Secure File Transfer Protocol (SFTP), and File Transfer Protocol over SSL (FTPS), have been adopted to meet business demands that FTP alone cannot.

Enterprises today depend on a mix of technologies to move files across their internal systems. These technologies include home-grown solutions, often built around FTP and vendor solutions. The vendor solutions are typically managed file transfer solutions that provide enterprises with features to secure, configure, track, and audit file transfer activity consistently.

Many customers move to a managed file transfer solution to satisfy regulatory-mandated compliance requirements. Compliance mandates, such as Health Insurance Portability and Accountability Act (HIPAA), Payment Card Industry Data Security Standard (PCI DSS), Voluntary Product Accessibility Template (VPAT), and Sarbanes-Oxley Act (SOX), require that the entire transaction flow be secured, auditable, documented, and accountable. As a result, companies must address the inherent weaknesses that exist with basic File Transfer Protocols.

Moving files in and out of protected networks also creates many security risks that cannot be addressed with basic file transfer. Entities typically choose to exchange data with external partners through one of the application layer protocols (for example, SFTP and HTTPS), proprietary protocols, e-mail, or through business-to-business technology and standards. It is common for organizations to use a hybrid of technologies to move data in and out of their protected networks.

The most secured and reliable way to move files between organizations is through business-to-business messaging standards like EDIINT AS1, AS2, and AS3, or ebMS, which are specifically designed for securely exchanging data over a public network. Business-to-business applications seek to improve organizational partnerships and transform the partnerships into inter-organizational relationships by acknowledging that the trading entities are known to one another and that all users are registered. The data exchange allows organizations to directly exchange information in a secure, standard method.

Business-to-business messaging protocols are standards that utilize application layer protocols and provide mechanisms for securing the data through encryption, signatures, and non-repudiation of sender and acknowledgment. The protocols are typically a wrapper or envelope that encompasses the business-to-business document or payload.

Business-to-business document standards like EDIX12, EDIFACT, HIPAA, HL7, and ebXML are designed to be cross-industry standards that provide a single architecture that utilizes a common uniform data format for electronic communications.

Typically, business-to-business solutions encompass traditional file transfer protocols, but also include the ability to move files according to the published business-to-business messaging and document standards. Additionally, vendor business-to-business offerings generally include partner profile management and transaction-viewing capabilities.

As the dependency on FTP and similar technologies and the need to externalize file transfers has grown, the limitations of the FTP and other application layer protocols has become a challenge for many companies. While companies have grown their file transfer infrastructure

using these application layer protocols, the lack of security and management capabilities have sent them looking for better solutions for security and management of their file transfers.

Challenges surrounding FTP in a multi-enterprise file transfer

File transfer is the simplest form of exchanging data between business entities and requires a common integration. The lowest common denominator to move a file is typically FTP.

FTP is a standard network protocol used over a TCP/IP network that allows businesses to move files between disparate systems regardless of operating systems. This flexibility is largely due to the client-server architecture of FTP. The inclusion of FTP into almost all operating systems has enabled its widespread use. This pervasive use of the technology has presented businesses with many challenges regarding the use of FTP. These challenges include but are not limited to the reliability, security, auditing/visibility, flexibility, and operational costs surrounding the use of FTP.

Limited reliability

Lacking checkpoint restart capability, file delivery is unreliable when a network interruption occurs. This situation often results in corrupt or partial files at the destination. Additionally, even with a successful transfer, the data can still be unusable at the destination due to a lack of character-set conversion. Ultimately, this can be costly for businesses that rely on FTP to move mission-critical files. Many companies find themselves with staff dedicated to cleaning up incomplete or failed transfers.

Limited security

FTP often requires a user name and password to send a file. These user names and passwords are sent across with the file as plain text. Additionally, many implementations of FTP do not offer privacy, authentication, or encryption. With checksum not available for all implementations of FTP, it can be almost impossible for companies to know or be able to show that the data that was supposed to be sent was actually sent without being modified.

Many implementations of FTP also lack a non-repudiation capability that allows for businesses to receive digital acknowledgement with trading partners that they successfully received the transfer. (*Non-repudiation* is the concept that an organization cannot refute the validity of a statement or contract). Non-repudiation typically must be in place to meet government standards regarding transfers in a court of law.

Limited auditability and visibility

With no centralized monitoring or management, FTP transfers can be nearly impossible to track from start to finish as the file moves across the enterprise. Logging capabilities are often limited and might only record transfers between systems that are directly connected. The limited logging requires companies to check logs at the source and destination servers for every file transfer. This can make it difficult to track a file across machines. Often, there is no history allowing for a complete picture of where the file has traveled.

Limited flexibility

Single-threaded FTP can only send or receive a single file at a time. To initiate a transfer, FTP also requires both the sending and the destination machines to be available simultaneously. FTP only allows for point-to-point transfers. This can mean that systems not directly connected together might have to be routed administratively through other boxes. FTP also lacks the ability for companies to prioritize the transfers.

These flexibility limitations often make it difficult or impossible for companies to automate their file transfer processes. When automation is possible, any scripts used during the transfer process generally reside on the machine on which they are used. This can require changes to be made on various servers that require platform-specific skills to make the modifications.

Often, these limitations paired together dictate what and how companies can move data across and outside their enterprises.

High maintenance costs

Seemingly free, FTP can be costly for companies as they try to develop around the inefficiencies. Requiring companies to dedicate resources to this task, the efforts to build, maintain, and support a custom solution built around FTP can quickly add up, causing companies to spend unexpected funding on this free File Transfer Protocol.

1.3 Overcoming FTP challenges with managed file transfer

Even with the availability of technologies such as web services and Web 2.0, file transfer still remains one of the most common ways for enterprises to exchange data. As more data is exchanged intra-enterprise and multi-enterprise, the need for more efficient means of moving files has become prevalent. This has brought about the concept of managed file transfer. Typically, managed file transfer refers to software solutions that allow for secure transfer of data from one location to another. A managed file transfer system introduces control, management, and auditability to address problems that arise when file transfers are used to integrate or connect business systems in the organization. Generally, managed file transfer solutions have features such as reporting the completion status of file transfers, auditing, global visibility, automation, and non-repudiation. These features are specifically designed to overcome the common challenges surrounding the enterprise use of FTP.

The following sections discuss how managed file transfer overcomes the current challenges of reliability, security, availability, flexibility, and costs when using FTP.

1.3.1 Improved reliability

Managed file transfer software tries to ensure that file contents only appear at the destination completely intact. The techniques for assured delivery vary among managed file transfer solutions, but most include the ability to resume or restart a file transfer that is interrupted because of network or system availability. Many solutions also include the ability to perform common code character set conversions based on selections specified when initiating a file transfer that enables the software to detect operating systems.

1.3.2 Improved security

Allowing for encryption of data, managed file transfer software keeps user IDs and passwords secured while in flight. Many offerings of managed file transfer software also include a checksum feature that allows a business to guarantee that the data being transferred is in its original form and has not been corrupted or tampered with.

Additionally, managed file transfer software allows companies to utilize non-repudiation that allows trading partners to be notified when the transfer has been received successfully, subsequently reducing the risk for conflicts or litigation. For file transfer, namely multi-enterprise file transfers, an organization cannot dispute that they received a file whenever a message disposition notification is given by specific protocols saying that the transfer is complete. This message disposition notification is a digital signature that the receiver did receive the transfer.

1.3.3 Improved auditability and visibility

Complete and detailed audit logs of the entire journey that a file takes are one of the main features typically offered in managed file transfer software. The logs typically show where the file originated, where the file went, who sent it, who initiated the transfer, who received the file, when the transfer was initiated, and when it was completed. Additionally, many of the software offerings for managed file transfer include the capability to control and monitor file transfers from a central location. The centralized control allows for administrators to easily keep an eye on where things are flowing throughout the enterprise.

1.3.4 Improved flexibility

Managed file transfer software strives to allow for time-independent transfers. Several of the most robust managed file transfer offerings include features that allow companies to initiate file transfers independent of source and target systems being available. Many of these managed file transfer offerings allow companies to send files to systems not directly connected, without requiring additional scripting or manual work. Furthermore, many managed file transfer solutions allow for files of any size to be sent across their technology. This approach allows the company to function based on business demands instead of technology dependencies.

Many solutions for managed file transfer include the ability to reconfigure and deploy a file transfer instantaneously from anywhere in the infrastructure. Moreover, many offerings of managed file transfer allow for multi-threaded transfers that enable businesses to send and receive multiple files at the same time.

The centralized control and monitoring available in many of the managed file transfer offerings also allows for automation to be set up in a central location. Many offerings expand the abilities for automation by integrating or building on scripting languages. Many of the offerings for managed file transfer include automation features for scheduling, triggering, and event-driven transfers.

1.3.5 Cost effectiveness

Companies looking to reduce costs can capitalize on the industry knowledge and experience of managed file transfer vendors by utilizing a managed file transfer solution. By utilizing offerings from managed file transfer solutions, companies can eliminate the costs associated with continually updating, maintaining, and improving their home-grown file transfer solution. Additionally, by utilizing the managed file transfer implementations, they will be able to take advantage of the continual improvements to features as business demands change and advance. Ultimately, this can allow for a higher utilization of the technology, while giving companies significant cost-savings.

1.4 Managed file transfer best practices

In this section we review best practices for file transfers that span both intra-enterprise and multi-enterprise networks.

1.4.1 Intra-enterprise managed file transfer best practices

Typically, companies are unaware of the various file transfers taking place in their protected network. It is common to walk into a business and find a variation of the scenario shown in Figure 1-1.

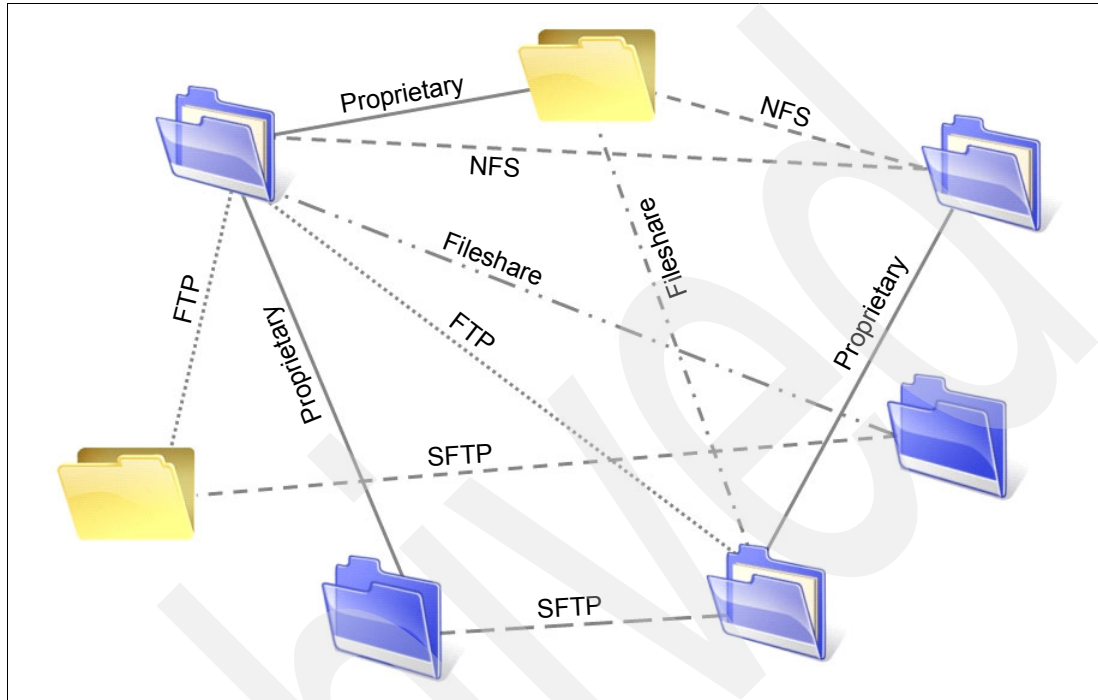


Figure 1-1 Typical file transfer topology found within companies

The combinations of many protocols and technologies leave your technicians scrambling to maintain disparate systems. Another challenge facing organizations, keeping employees' skills for the various protocols up to date, can be challenging too. Additionally, the more technologies that you add to an organization, the harder it is to control who is using the technologies and to monitor the activity.

Intra-enterprise managed file transfers re-enforce a company's service-oriented architecture (SOA) strategy. The ideal managed file transfer architecture contains automation, centralized and event-based logging, centralized monitoring, centralized setup and management, and a documented and standardized transport. The infrastructure contains:

- ▶ The ability to secure files in transit
- ▶ Flexibility in file transfers
- ▶ Granular user control over file transfers
- ▶ Monitoring of a file's journey
- ▶ Auditing of transfers
- ▶ Visibility of transfers
- ▶ Checkpoint restart of transfers

The ideal intra-enterprise managed file transfer topology utilizes a single reliable transport (Figure 1-2). The topology cuts down on the cost and time for information technology (IT) development and maintenance by eliminating the need to write code. This allows for the configuration and extension of the managed file transfer software to consolidate IT administration and operational efforts. The ideal managed file transfer topology preserves the integrity of data to support a company's compliance requirements by being secure, reliable, and resilient and allowing auditing. The topology for moving files easily integrates with a SOA enterprise service bus to transform, parse, and route data.

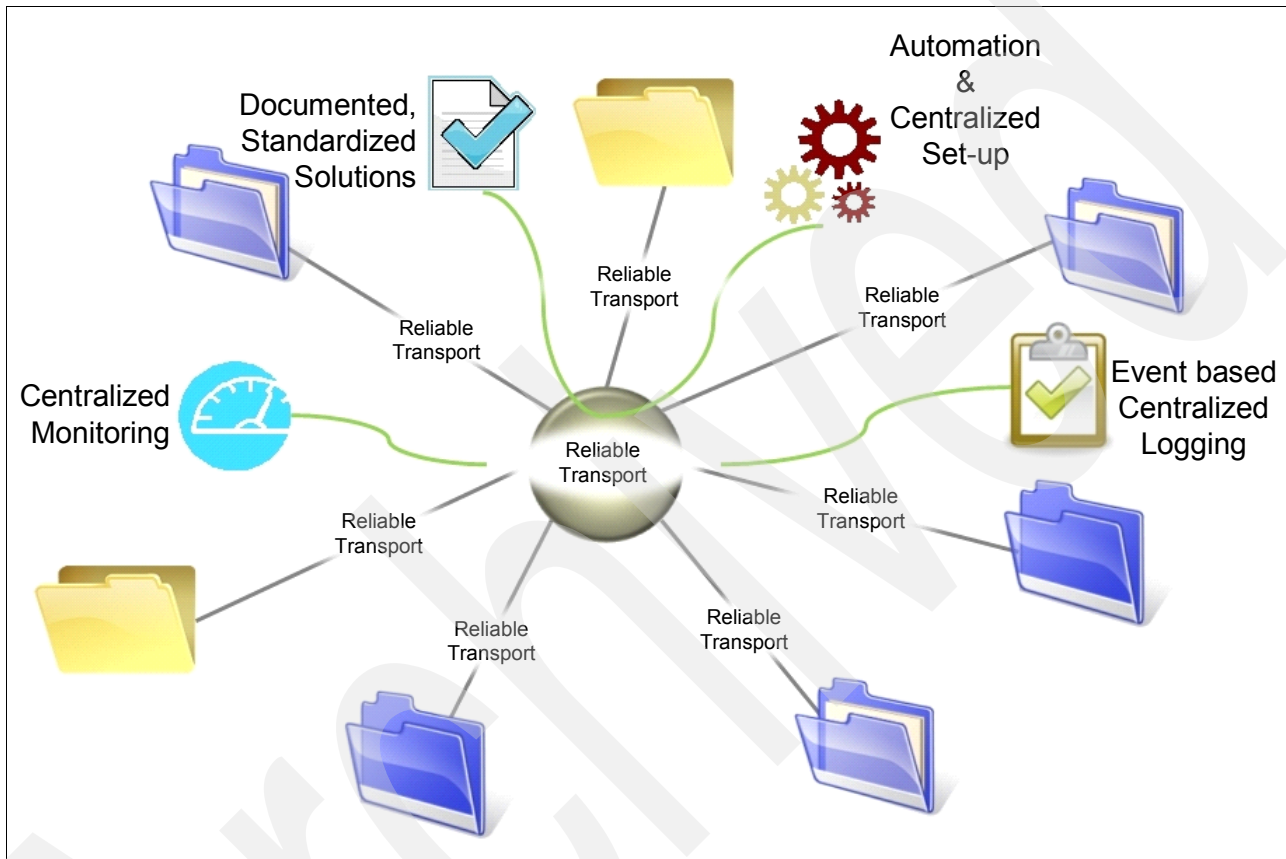


Figure 1-2 Ideal managed file transfer topology

The key features for intra-enterprise managed file transfer best practices are:

- ▶ Common reliable transport protocol
- ▶ Centralized monitoring
- ▶ Event-based centralized audit logging
- ▶ Automation
- ▶ Centralized setup and configuration
- ▶ Documented standardized solutions
- ▶ Check-point recovery
- ▶ Centralized management

1.4.2 Multi-enterprise managed file transfer best practices

Enterprises working with various trading partners typically are sending data across multiple firewalls and are working with multiple document, messaging, and transport standards. The various protocols and document standards require proper management and support. As more trading partners are integrated into the managed file transfer topology, it becomes

necessary to manage trading partner IDs, authenticate users, and authenticate the data flowing into and out of the company.

The ideal multi-enterprise managed file transfer topology places a gateway between a company's intra-enterprise managed file transfer topology and its trading partner. Typically, this gateway resides in the demilitarized zone (DMZ). The gateway should be capable of handling a wide range of protocols to meet current and future business requirements. Additionally, the gateway should be designed to handle large amounts of file transfers, meet or exceed performance expectations, and meet security certification standards.

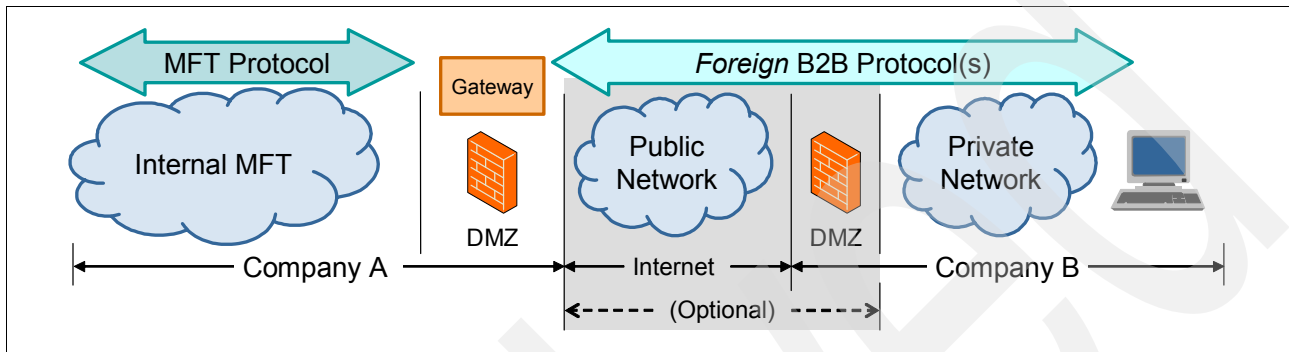


Figure 1-3 Ideal multi-enterprise managed file transfer topology

The gateway should also support standards-based business-to-business protocols and be designed specifically for DMZ deployments. B2B gateways allow for ease of managing and connecting to trading partners using industry standards and provides trading partner management for business-to-business governance, utilizes business-to-business protocol policy enforcement, implements access control, executes message filtering, and performs data security. A trait also common to most B2B gateway products is a user interface for B2B configuration and transaction viewing. The user interface of the B2B gateway is used to correlate documents and acknowledgements while displaying all associated events.

The key features of multi-enterprise managed file transfer best practices are:

- ▶ Trading partner management
- ▶ Hardened security for DMZ deployment
- ▶ Business-to-business governance and security
- ▶ Broad range of business-to-business and transport protocol support
- ▶ User interface for configuration and transaction viewing
- ▶ Interface for trading partner transaction viewing
- ▶ Assured delivery with automatic resend

1.5 Comparing intra-enterprise and multi-enterprise file transfers

Any time that a file moves there are certain basic concerns regarding the movement between the source and the destination. The severity and detail of these concerns vary depending on whether the file in transit is subject to meeting industry compliance standards, moving over a public network (internet), or staying inside the protected network. The level of risk associated with moving a document outside the enterprise is the key differentiator between intra-enterprise and multi-enterprise file transfer and drives the types of technologies and products used to mitigate that risk.

1.5.1 Control

The amount of control that one enterprise has over file transfers varies depending on whether the files are transferred within the enterprise or outside the enterprise to external partners.

Intra-enterprise file transfers

File transfers moving only inside an organization can be controlled easier than a multi-enterprise file transfer. While moving a file inside an enterprise, entities can control the coordination of the source and destination targets, view logging data available at the source or the destination, and store information pertaining to the transfer that is required for auditing. These activities can be performed with a complete picture of the components involved in the transfer.

The level and type of information included in logging varies depending on the protocol chosen for the file transfer. Additionally, with access to both source and destination targets, organizations can see a complete picture of the file transfer and monitor the source or destination targets as needed to alert them of issues relating to the file transfer.

Multi-enterprise file transfers

Moving files outside of an organization's firewalls forces the organization to relinquish a certain amount of control over the transmission. With the inability to control what is going on at the external end of the transfer, coordinating the submission of the transfer can be challenging. Many of the protocols require the sending and receiving platforms to be available at the time of the transfer initiation. This can require an organization to resubmit a transfer request multiple times before the transfer completes. Additionally, many of the protocols can report a successful transmission even when the file did not actually transmit for various reasons. Without the ability to see the transmission completely, organizations might have no idea whether a file was successfully received if they use traditional application layer file transfer protocols. Standards such as AS2 help resolve this issue through a Message Disposition Notification (MDN) that provides a message-level acknowledgment that the transfer was a success.

Monitoring and complete visibility can be a challenge with multi-enterprise transfers. Most of the typical protocols do not provide any visibility or monitoring into the transmission. Certain protocols can log at the source, destination, or both, but without access to both logs, entities are left with an incomplete picture.

1.5.2 Authentication and data validation

When receiving a data transmission, whether intra-enterprise or multi-enterprise, there are questions regarding the transmission. Namely, who sent it and whether this is the data that should have been sent. The requirements for verifying the sender and the validity of the data vary depending on whether the transfer occurred intra-enterprise or multi-enterprise.

Intra-enterprise file transfers

In intra-enterprise file transfers, organizations typically do little user authentication or data validation. Typically, the sender's user ID is authenticated against the destination operating system to ensure that the sender has access to the system and permission to access or write to the desired destination directory. This process does not usually include any data validation to ensure that the data is safe because the electronic transmission is coming from a trusted source inside the protected network.

Multi-enterprise file transfers

User ID authentication and data validation are a primary concern for multi-enterprise file transfers. However, the authentication concerns begin before a user ID or the data comes into view. First, a multi-enterprise file transfer should verify that the Internet Protocol (IP) address or range of addresses being used in the transmission is allowed access to the systems inside the DMZ. This is typically done at the external firewall utilizing inbound firewall rules. Once the IP address or range of addresses is determined to be valid, the user or partner ID involved in the transmission should be authenticated before files can be exchanged. Once the user or partner IDs are validated, also validate the data to ensure that the file type and format is allowed by the receiving system.

1.5.3 Network security

As files travel across the network, security measures should be put in place to protect the data as it is transmitted.

Intra-enterprise file transfers

Typically, encryption inside a protected network is not an issue for organizations. The exception to this rule occurs when the data in question is subject to compliance standards, such as Payment Card Industry (PCI). These standards can often require data to be encrypted when on a file system and in flight. When this is necessary, entities look to encrypting the data or encrypting the file system and the transmission channel. Additionally, there are few, if any, concerns about ports being used for the electronic communication.

Multi-enterprise file transfers

Every open port through a firewall is another possible entry into the organization's demilitarized zone (DMZ), protected network, or both. Multi-enterprise file transfers might require one or more ports to be opened through the external firewall depending on the protocol in use. The inner firewall needs to be locked down to allow ports only open from the gateway sitting in the DMZ. As more protocols are added to the organization, more ports need to be opened through the firewalls. Proper network and data security must be used together to ensure that the data and the intra-enterprise protected network are protected.

Once the file leaves an organization's secured zone, if the data contains sensitive information, such as user credentials or account numbers, it immediately becomes a security risk. The risk typically requires the data to be encrypted while it is in the DMZ and while in flight into and out of the DMZ and to and from an external partner.

1.6 Recent additions to the managed file transfer portfolio

With the acquisition of Sterling Commerce, an IBM Company, IBM enhanced its managed file transfer portfolio consisting of WebSphere MQ File Transfer Edition with the Sterling Business Integration Suite. The Sterling Business Integration Suite consists of IBM Sterling File Gateway and IBM Sterling Connect:Direct. Sterling Commerce transforms and optimizes your business collaboration network by improving business agility, efficiency, and performance. This is done with a comprehensive, yet modular, suite of solutions that solve integration challenges both inside and outside an enterprise. Sterling Commerce has been recognized as the market leader in providing B2B integration and managed file transfer solutions.

These managed file transfer components from Sterling Commerce partnered with WebSphere MQ File Transfer Edition deliver proven value by protecting privacy and integrity of data in transit with governance, eliminate operations call center traffic regarding file transfer

exceptions, show a faster time to revenue, and bring a six-sigma level performance to key business processes. IBM Sterling File Gateway acts as the hub for managed file transfer by providing a broad file transfer protocol support, governance, management, and visibility. IBM Sterling Connect:Direct provides partner-to-partner file transfer optimized for high-volume, assured data delivery of files. MQ File Transfer Edition provides enterprise class integrity, performance, and auditability for managed file transfer. The integration and combination of these products allows for organizations to switch between protocols internally, allowing for diversity across business needs. It also positions the organization to easily move files outside their secured intra-enterprise network through an edge server to the external trading partner regardless of what protocol the external trading partner is using.

1.7 Who should read this book

The operations of organizations vary and often result in the various departments within an organization acquiring their own solutions. This can mean that multiple departments within the organization are using different products to accomplish a task such as file transfer. This book is intended for those organizations that find themselves wanting to trade data in a secure, reliable, and auditable way across protocols both intra-enterprise and multi-enterprise.

Architects, system administrators, system programmers, and developers looking to build a managed file transfer solution should read this book. With a recent acquisition, many organizations can find themselves with various options to move files across multi-enterprises. This book shows how to combine Sterling Connect:Direct, Sterling File Gateway, and MQ File Transfer Edition to provide a seamless transport.



File transfer products overview

In this chapter, we introduce the IBM products that we use in this book for multi-enterprise file transfers. You can use these products to move files between the two IBM proprietary protocols and common application layer protocols, both internally and externally.

We discuss the following IBM products in this chapter:

- ▶ IBM Sterling Connect:Direct
- ▶ IBM Sterling B2B Integrator
- ▶ IBM Sterling File Gateway
- ▶ IBM Sterling Secure Proxy
- ▶ IBM WebSphere MQ File Transfer Edition
- ▶ IBM WebSphere Message Broker

2.1 Sterling Connect:Direct

Sterling Connect:Direct is a point-to-point (peer-to-peer), file-based integration solution that is designed for around-the-clock, unattended operation. It provides assured delivery and high-volume, secure data exchange within and between enterprises. It is designed to move files that contain any type of data (for example text, EDI, binary, digital content, or image) across multiple platforms, disparate file systems, and disparate media, while maintaining high performance levels and throughput. Many industries throughout the world use Sterling Connect:Direct to move large volumes of data and to connect to remote offices. Unlike FTP implementations, Sterling Connect:Direct eliminates the need for manual intervention in data delivery, improving personnel productivity and the reliability of business processes.

Sterling Connect:Direct provides the following benefits:

- ▶ **Predictability**

Files can be sent using assured delivery through automated scheduling, checkpoint restart, and automatic recovery or retry. If a file transfer is interrupted, Sterling Connect:Direct attempts to resume the transfer at a predefined interval for a configured duration of time. The activity and statistics that are associated with the file transfer are logged to provide an audit trail that accounts for all actions taken during a file transmission.

- ▶ **Security**

The Sterling Connect:Direct proprietary protocol and user authentication through user proxies allows customer information to remain private during the file transfer. Featuring security options to control data access, network access, or access to system resources, Sterling Connect:Direct can interface with operating system and vendor-supplied access control and security software. The optional implementation of IBM Sterling Connect:Direct Secure Plus gives organizations the ability to use a comprehensive cryptographic solution for strong mutual authentication using X.509 certificates, SSL and TLS data encryption, and data integrity checking.

- ▶ **Performance**

Sterling Connect:Direct can handle demanding file transfer workloads, including high volumes of small files and transmission of large, terabyte size files. Additionally, Sterling Connect:Direct provides optional data compression that is configured for maximum compression or compression based on the optimal use of system resources.

2.1.1 Sterling Connect:Direct additional features

You can add the additional options that we describe in this section to Sterling Connect:Direct to extend its functionality.

IBM Sterling Connect:Direct Secure Plus

The Sterling Connect:Direct Secure Plus option is available to provide a full security solution. This option is a separate, additional licensing option of Sterling Connect:Direct. Sterling Connect:Direct Secure Plus enables organizations to use security protocols to secure data during electronic transmission. The protocols that are available for use include Transport Layer Security (TLS), Secure Sockets Layer (SSL), and Station-to-Station (STS).

The SSL and TLS protocols provide the following levels of security:

▶ **Layer 1: Server authentication**

Server authentication is activated when a trading partner connects to a Connect:Direct server. After the initial handshake, the Connect:Direct server sends its digital certificate to the trading partner. The trading partner checks expiry and for a trusted certificate authority.

▶ **Layer 2: Client authentication**

For client authentication, the trading partner must send its own certificate. When client authentication is enabled, the trading partner's certificate is requested after the server authentication is complete. The Connect:Direct server verifies that the client certificate is signed by a trusted source before establishing the connection.

▶ **Layer 3: Common certificate verification**

The Sterling Connect:Direct Secure Plus server searches the certificate file that is received during client authentication for a matching certificate common name. If a certificate common name is not found, communication fails.

The following encryption algorithms are included with Sterling Connect:Direct Secure Plus Option:

▶ **Symmetric:**

- AES
- DES
- 3DES
- RC4

▶ **Asymmetric: RSA**

- ▶ **FIPS:** Uses Crypto-C, which is Sterling Commerce's FIPS 140-2 validated security module on the UNIX and Windows® platforms and that uses the IBM eServer™ cryptographic coprocessor on the mainframe.

The following FIPS-validated algorithm implementations are supported by the Sterling Connect:Direct Secure Plus Option:

- DES, FIPS 46-3, NIST Certificate #160
- 3DES, FIPS 46-3, NIST Certificate #100
- SHA-1, FIPS 180-1, NIST Certificate #89
- AES, FIPS 197, NIST Certificate #5
- DSA, FIPS 186-2, NIST Certificate #70

FIPS compliance can be achieved with Sterling Connect:Direct only by installing Sterling Connect:Direct Secure Plus Option and enabling FIPS mode on the supported platforms.

File Agent

Sterling Connect:Direct contains a component called File Agent that provides unattended file management through monitoring and detection capabilities that can enhance automation in Connect:Direct processes.

You can configure Sterling Connect:Direct File Agent to operate in the following ways:

- ▶ Watch for any file to appear in a watched directory. When the added file is detected, submit a default Connect:Direct process.
- ▶ Use a watched file event rule or system event rule that is enabled for configuration to override a default Connect:Direct process. When the criteria for a rule is met, the File Agent submits the Connect:Direct process that is associated with that rule.
- ▶ Create File Agent rules based on the following properties:
 - A full or partial name of the file is detected in a watched directory. The watched directory can be a local directory on the Connect:Direct server or a network drive.
 - The size of the file is detected in a watched directory.
 - A system event title or contents exist.

File Agent is distributed with Sterling Connect:Direct for UNIX, Windows, and z/OS®. It can also be downloaded from the Sterling Commerce Customer Center portal.

Sterling File Accelerator

Sterling File Accelerator is a user-defined type (UDT), UDP-based Data Transfer, solution that provides faster file transfers for high-volume files than TCP over high-speed networks with high latency.

Microsoft Windows Software Development Kit

The Microsoft® Windows Software Development Kit (SDK) is used to integrate Sterling Connect:Direct operations into custom-built applications. The SDK uses a 32-bit interface for C and C++ and an OLE automation server for Visual Basic applications. The SDK also provides ActiveX controls for Submit Process and Select Statistics commands. The tools that are available in the SDK include C API functions, C++ Class interface, ActiveX control interface, direct automation servers, and user exits.

Simple Network Management Protocol Agent

The Sterling Connect:Direct Simple Network Management Protocol (SNMP) Agent is a proxy agent that enables a Connect:Direct server to provide information to SNMP network management stations. This agent allows the SNMP network management stations to have access to the following information:

- ▶ General condition of the Connect:Direct server
- ▶ Alerts for events requiring further investigation
 - Possible security violations
 - Failing processes
 - Session failure

Clustering solutions

Sterling Commerce provides support for clustered environments, such as IBM Sysplex, Symantec Veritas, Sun Solaris Cluster, and Microsoft Cluster Server.

2.1.2 Scalability

The Sterling Connect:Direct event-based architecture enables the movement of high volumes of files and large file sizes as a result of no product-defined limits on file sizes. The event-based architecture scales to ensure that organizations can handle peak demand and can keep pace as business volumes grow, whether they are operating mainframes or distributed or clustered servers.

2.1.3 Capabilities

Sterling Connect:Direct supports around-the-clock, unattended operations using the following built-in features:

- ▶ Automation and management
 - Schedules jobs on a one-time, recurring, or continuous basis
 - Assigns and manage file transfer workloads to internal queues
 - Uses event-driven alert notification
 - Integrates with back-end systems using process language builds scripts
 - Gains programmatic access to transfers from other applications through API and SDK
- ▶ Assured file delivery
 - Checkpoint restart
 - Automatic recovery from network interruptions
 - Automated alert notifications for success or failure
- ▶ Security and compliance
 - Standard Sterling Connect:Direct
 - Interfaces with operating system security for user authentication
 - Provides a complete audit trail of data movement through extensive statistics logs
 - Sterling Connect:Direct Secure Plus
 - User authentication
 - X.509 certificates for authentication
 - Data encryption (SSL/TLS)
 - Certificate and Certificate Revocation List (CRL) checking
 - FIPS 140-2 and Common Criteria certification
 - Sterling Secure Proxy for the Sterling Connect:Direct protocol
 - DMZ-based authentication, session break, and SSL termination
 - No file stored in the DMZ
 - No inbound ports opened in the firewall
 - Validation of the Sterling Connect:Direct protocol
- ▶ Multiple platform support
 - Operating systems support
 - z/OS and z/VSE™
 - OpenVMS
 - i5/OS® (OS/400®)
 - UNIX and Linux®
 - Windows
 - HP NonStop
 - Sterling Connect:Direct Select (Java™ version that can run on multiple platforms)
 - Network protocols support
 - TCP/IP
 - SNA
 - UDT (UNIX 4.0, z/OS 4.8, Windows 4.5)

Sterling Connect:Direct ensures data delivery to the correct destination within the correct time window, allowing the receiving application to process and act upon it consistently.

2.1.4 Architectural overview

Sterling Connect:Direct includes the following components that define the local and remote nodes, the users who can access the nodes, and the functions that users can perform:

- ▶ *Local nodes* are defined during installation. The definition specifies information such as the operating system, default user ID, TCP/IP address, and port number that is associated with the local node. You can modify these settings after installation.
- ▶ *Local user authorities* restrict the ability of each user to perform certain tasks. Sterling Connect:Direct has two types of users:
 - Administrators
 - General users

Each type of user has a set of default privileges. These default privileges can act as templates to assign to other user authorities and to restrict user access.

- ▶ *Remote user proxies* contain remote user information for operations that are initiated from a remote Connect:Direct node. The definition identifies a proxy relationship between a user ID at a remote node and a local user ID. This mapping of user IDs allows for remote nodes to submit work without explicitly defining user IDs and passwords in the processes and eliminates the need to share passwords with trading partners.
- ▶ *Client interfaces* allow communication with the Connect:Direct server. The client interfaces offered include a web browser interface, a GUI, a command-line interface (CLI), and panels.

Sterling Connect:Direct relationships

Sterling Connect:Direct uses a peer-to-peer relationship and a client/server relationship. A *client* is used to communicate with the server regarding the transfer work to be performed. A client is one of the user interfaces provided with Sterling Connect:Direct that sends processes or commands to initiate the transfer. The *server* is the system where Sterling Connect:Direct resides to send a file or receive a file. Each data transfer involves a local and a remote Connect:Direct server (also referred to as a *node*). The servers work together to perform work in a peer-to-peer relationship.

Figure 2-1 shows the relationships between Connect:Direct nodes during peer-to-peer sessions.

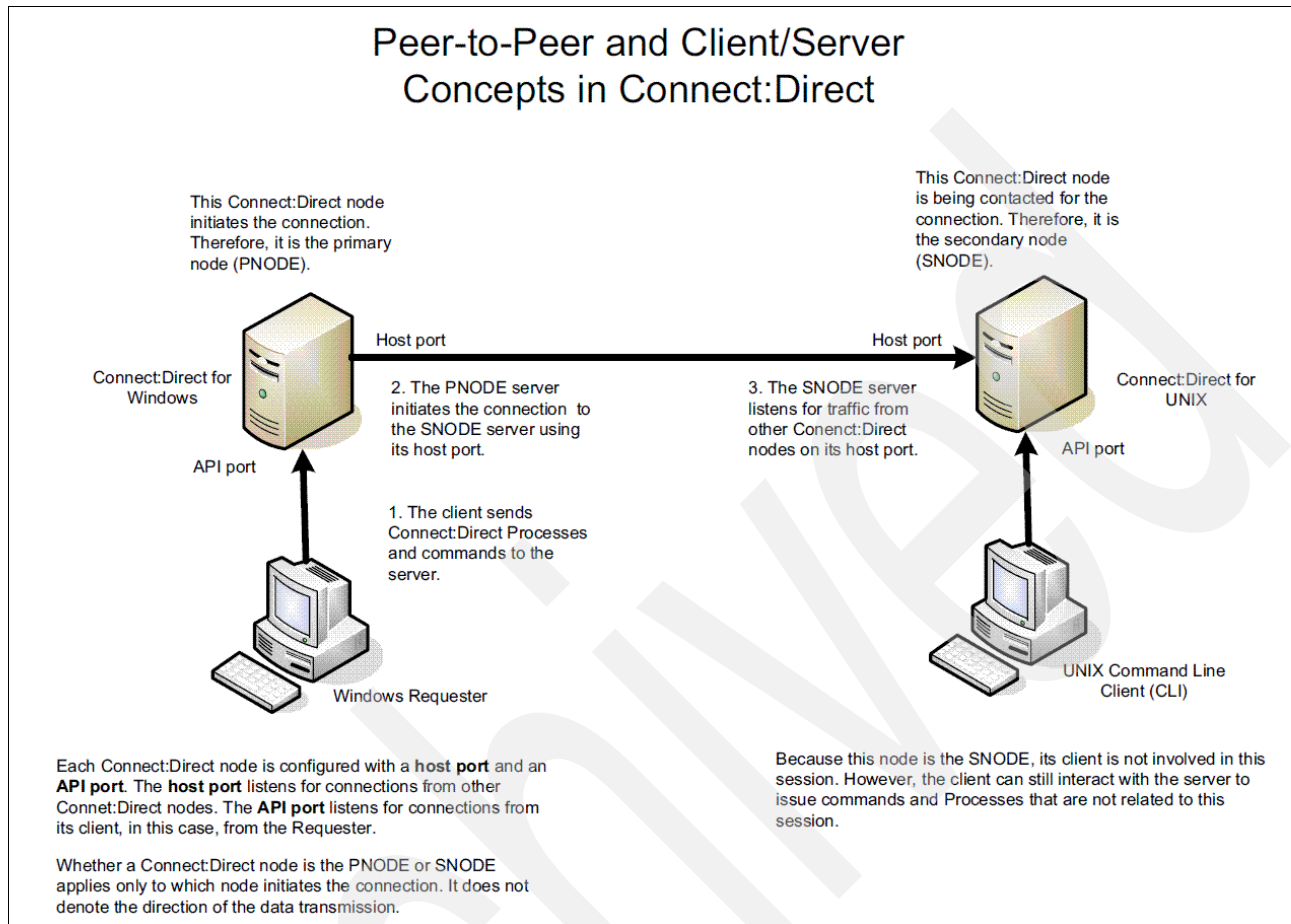


Figure 2-1 Sterling Connect:Direct relationships overview

The server that initiates the connection is the *primary* node (PNODE). The server that receives the connection for transfers is the *secondary* node (SNODE). A Sterling Connect:Direct server can manage multiple concurrent connections with other servers and act as both a PNODE and a SNODE.

Netmap

The Network Map (netmap) is a file that is created during the Sterling Connect:Direct installation that identifies the remote nodes with which each local node can communicate and the communication information that is needed to establish a connection. A remote node entry must be created in the netmap for which each remote node that the local node communicates. Each netmap entry contains the following information:

- ▶ Remote node name
- ▶ Operating system
- ▶ Session characteristics for a protocol
- ▶ Transfer and protocol information for the available communications paths

User interfaces

The following interfaces connect to Connect:Direct servers as clients to initiate work:

- ▶ Sterling Connect:Direct Browser user interface

The browser allows users to create, submit, and monitor Connect:Direct processes from a web browser. Additionally, system administration tasks, such as viewing and changing the netmap or initialization parameters, are performed through the interface. The ability to perform administration tasks depends on the platform on which the user is logged in and the security level for that user.

- ▶ Connect:Direct requester

The requester is a GUI that allows users to connect to servers to perform tasks such as:

- Initiate file transfers.
- Run remote programs or batch jobs.
- Create, submit, or monitor Connect:Direct processes.
- Manage administration tasks.

The requester is available for use with Sterling Connect:Direct on Windows, UNIX, and OpenVMS servers.

- ▶ CLI

The CLI allows users to issue commands to the servers and monitor processes. The CLI is available for use with Connect:Direct servers on Windows, UNIX, HP NonStop, and OpenVMS.

- ▶ Panels

The panels are available only for z/OS systems. The panels are used through the Interactive System Productivity Facility (ISPF) Interactive User Interface for administration.

2.1.5 Connect:Direct process language

The Connect:Direct process language gives instructions to the Connect:Direct servers, telling them the work to perform in an organization. The process contains special statements and parameters that perform data movement and pre-transfer or post-transfer activities, such as:

- ▶ Move files between Connect:Direct nodes.
- ▶ Run jobs, programs, and commands on the Sterling Connect:Direct system.
- ▶ Start other processes.
- ▶ Handle processing errors.

The processes can link to network or application activities to create a continuous cycle of processing. For example, a network message can trigger a file transfer that is used by another application. As the process executes and completes, audit information is available for analysis and for use in future processing.

Processes contain parameters to control the attributes. The parameters are specified within the actual process or when the process is submitted. Any parameters specified at submission override parameters that are coded in the process. The following parameters are available to control attributes:

- ▶ Scheduling information
 - Set a process to run automatically at a specific day and time or at a given interval without operator intervention.
- ▶ Integration with existing security systems
 - Specify user IDs and passwords in a process to allow it to work within the organization's existing network security system.
- ▶ Data transmission integrity
 - Specify checkpoint and restart intervals within a file transmission to allow restarts to begin automatically from the most recent checkpoint.
- ▶ Compression
 - Use data compression for copy operations for shorter transmission times.
- ▶ User notification
 - Notify users automatically of successful or unsuccessful transfers.

Using the process language

The Connect:Direct process uses its own scripting language to define the work that is performed in a process. A process statement must be the first statement in a process. Any statements following the process statement can occur in any order. Each statement in the process uses parameters to control activities, such as execution start time, user notification, security, or accounting data. You can use the following statements in a process:

- ▶ PROCESS
- ▶ COPY
- ▶ RUN JOB
- ▶ RUN TASK
- ▶ SUBMIT
- ▶ SYMBOL
- ▶ Conditional (IF, EIF, ELSE, EXIT, GOTO)
- ▶ Pend

You can create a process using one or more of the following tools:

- ▶ Process Builder feature in the Sterling Connect:Direct browser
- ▶ Connect:Direct requester for Microsoft Windows
- ▶ Text file submitted to a Connect:Direct server through a batch utility, command line, or an application written using the Sterling Connect:Direct API
- ▶ z/OS system panels

The Process Builder is a GUI that enables you to build, modify, and save processes. It handles the Connect:Direct process syntax rules automatically and eliminates the typographical mistakes that you might encounter when creating a process with a text editor. The process can undergo syntax validation and allows you to submit completed processes from the Process Builder.

After the process is created, it is submitted to a server for execution. Figure 2-2 shows how a process executes.

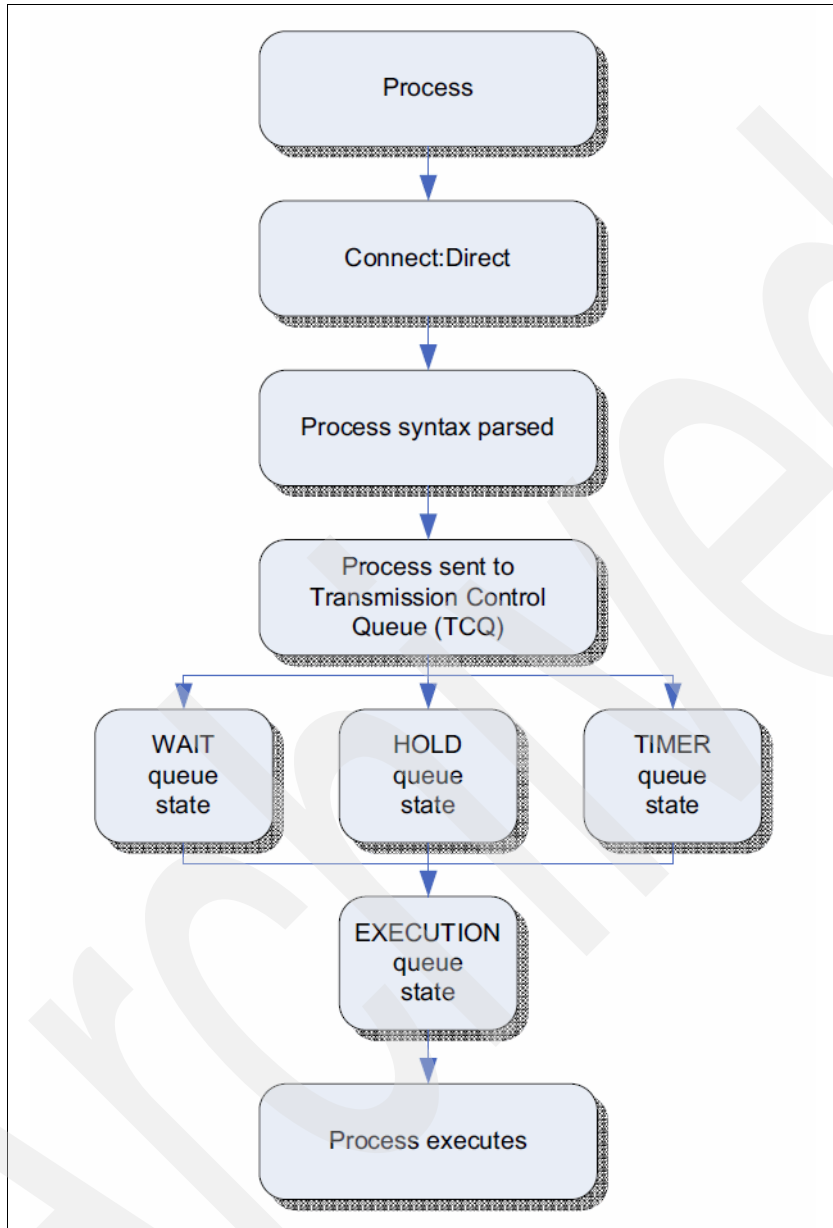


Figure 2-2 Process execution steps

Table 2-1 describes the execution steps shown in Figure 2-2.

Table 2-1 Process execution step explanation

Step	Description
Process submitted	A user submits a process from a Connect:Direct process library or from a Sterling Connect:Direct browser.
Process syntax parsed	The parser within Sterling Connect:Direct verifies the process syntax.

Step	Description
Process sent to Transmission Control Queue (TCQ)	<p>When the process passes syntax checking, it is placed on a work queue, according the parameters that are defined in the process, such as priority, class, or start time. The work queues are jointly referred to as the <i>TCQ</i>.</p> <p>A process is in one of the following states in a TCQ:</p> <ul style="list-style-type: none"> ▶ EXECUTION: The process is executing. ▶ WAIT: The process is waiting until a connection with its SNODE is established or available. The process might also be waiting for its turn to execute on an existing session. ▶ HOLD: The process might have been submitted with a HOLD or RETAIN@ parameter. The process remains in a HOLD step until it is released by an operator or until an SNODE connects with a request for held work. The HOLD queue is also used for processes that stop executing when an error occurs. ▶ TIMER: A process submitted with a STARTT parameter designates the time or date, or both, when the process executes. Processes that initially failed due to an inability to connect to an SNODE or because of a file allocation failure can also display in this queue while waiting for a retry interval to expire. Processes such as these retry automatically. <p>A queue process can be queried and manipulated through Sterling Connect:Direct commands such as:</p> <ul style="list-style-type: none"> ▶ SELECT ▶ CHANGE ▶ DELETE ▶ FLUSH ▶ SUSPEND PROCESS <p>A message indicating that the process submitted successfully is created when the process is placed in the TCQ. At that point, the process statements have passed syntax validation, but the process cannot be selected for execution.</p>
Process executes	The process is selected for execution based on process parameters and the availability of the SNODE.

Manage the process

Sterling Connect:Direct provides the following tools to allow you to manage the processes that are used:

- ▶ Process Monitor

Allows you to view processes in the TCQ, release held processes, change the status of a process, and delete a process.
- ▶ Process Notification Utility

Allows users to change the notification method that is defined at the Sterling Connect:Direct installation to notify users of the process execution.
- ▶ Message Lookup

Allows users to look up the meaning of an error message. This is used to view more information regarding the error message.

- ▶ **SNMP**
Allows users using SNMP to capture messages. Users identify which messages to include and determine whether messages are trapped or logged in to the event log.
- ▶ **Cyclic redundancy check (CRC)**
The CRC checking determines whether data received by Sterling Connect:Direct over the network has been altered during transmission. To allow for data integrity during the transmission, CRC is generated for the entire buffer, including the header. CRC calculates a short, fixed-length binary sequence for each block of data and for sending or storing them together. When a block is read or received, the calculation is repeated. If the new CRC does not match the earlier calculation, the process execution is stopped and restarted from the last checkpoint record. CRC is performed only for TCP/IP processes and cannot be used with the Sterling Connect:Direct Secure Plus Option, which uses its own data integrity checking natively.
- ▶ **CLI**
The CLI provides commands to access queues and manage processes. The commands enable users to control the process execution and to view the process status and results. The commands are issued in a native command text format through the Sterling Connect:Direct API.
- ▶ **Sterling Connect:Direct browser user interface**
The user interface allows users to build, submit, and monitor processes from a web browser.
- ▶ **File Agent**
Sterling Connect:Direct File Agent is a feature of Sterling Connect:Direct that provides unattended file management. File Agent watches directories to detect new files. When a new file is detected, the File Agent submits a default process or evaluates the file using rules to override the default process. File Agent uses the rules to further determine which process to submit. Processes are selected for submission based on the following properties:
 - Specific or partial file names
 - File size
 - System events

2.2 IBM Sterling B2B Integrator

Sterling B2B Integrator is a transaction engine that runs user-defined processes and manages them according to business requirements. Its platform supports high-volume electronic message exchange, complex routing, translation, and flexible interaction with multiple internal systems and external business partners. Sterling B2B Integrator is a prerequisite for Sterling File Gateway and provides the underlying architecture and multiprotocol functionality that is used by the advanced file transfer routing features of Sterling File Gateway.

Sterling B2B Integrator provides the following functions:

- ▶ Ties together applications, processes, data, and people, both within and outside your organization.
- ▶ Offers flexible options for deployment, configuration, and customization, including the functionality to add capabilities one at a time.
- ▶ Complements, rather than disrupts, critical existing systems.

- ▶ Provides a robust security infrastructure.
- ▶ Includes visual management tools for easy configuration of and visibility into work flows, system and trading partner activities, translation maps, and business process implementation.
- ▶ Works with existing and emerging business and communication standards.

Together, these features enable organizations to configure the components, enabling them to meet evolving application integration requirements.

2.2.1 Capabilities

Sterling B2B Integrator includes the following capabilities:

- ▶ Support for any communication standard, protocol, including:
 - HTTP
 - HTTPS
 - FTP
 - FTPS
 - SMTP
 - SOAP
 - EDIINT
 - AS1
 - AS2
 - AS3
 - Data format
 - File
- ▶ Any-to-any data mapping and translation
- ▶ Securely and flexibly integrate trading communities
- ▶ Community self-provisioning, outsourced recruitment, onboarding, testing, and multi-lingual technical support
- ▶ Full range of encryption, certificate types, digital signatures, and identity management methods
- ▶ Extension of internal business processes to external partners
- ▶ Integration adapters for any ERP, messaging system, or data storage system
- ▶ Document conversion and processing:
 - Routing
 - Business rule validation
 - Delivery
 - Alerting
 - Exception handling, and
 - Archiving for end-to-end process automation
- ▶ Process modeling, execution, and orchestration with predefined process templates
- ▶ Scalability to meet fluctuations in transaction volume
- ▶ End-to-end order visibility for tracking transactions throughout the life cycle of an order and ensuring that orders are never lost
- ▶ Mailbox store-and-forward services

Sterling B2B Integrator is optionally used to facilitate e-business with trading partners for a vast assortment of transactions, including electronic data interchange (EDI), email, and

reliable bulk file management. Organizations can build human intervention points, such as approvals, into processes and set up self-service access to information across trading partner systems.

2.2.2 Terminology

This section explains of the Sterling B2B Integrator specific terminology that we use in this book.

Business processes

Sterling B2B Integrator's approach to integration is centered around business process management. A *business process* is a goal-driven, ordered flow of activities that accomplishes a business objective. Using Sterling B2B Integrator, organizations can integrate the activities that make up their business processes. Common examples of such activities include:

- ▶ XML, EDI, and proprietary file translation, transformation, and filtering
- ▶ Human interaction through a browser interface (such as reviewing and approving data)
- ▶ Content-based routing of messages
- ▶ Data publishing
- ▶ Extended process models that integrate the execution of a B2B protocol, such as AS2, with enterprise system integration, such as invoking the SAP adapter

Organizations can create and coordinate activities into business process models, extending the automation of processes and increasing the value of e-business operations.

An example of a simple business process is the fire-and-forget publishing of a business event to a group of interested participants. The steps that comprise the process trigger the process and the subsequent publishing of the event to the interested parties.

A complex business process might require multiple interactions among many applications in a start-and-stop, request-response mode, along with human interaction, occurring over a long period of time.

Services and adapters

A *service* is a set of instructions that the Business Process Engine uses to perform an activity in a business process. *Adapters* are services that connect the Business Process Engine and other system components to dissimilar systems and applications outside of the Sterling B2B Integrator environment. Business processes can send, pause, retrieve, and fully interact with adapters.

Services and adapters are reusable. Organizations can include them in multiple business process models.

Adapters either receive input from or provide output to outside systems. Adapters provide noninvasive integration with enterprise resource planning (ERP), supply chain management (SCM), customer relationship management (CRM), other packaged applications, enterprise applications, communication protocols, messaging solutions such as IBM WebSphere, and databases.

The following process summarizes the way adapters work within a business process:

1. The business process progresses to the application adapter step.
2. The adapter calls a third-party application to perform an activity.
3. The system records the modified state (context) of the process and related data.
4. The business process continues to the next service or adapter.

2.3 IBM Sterling File Gateway

Sterling File Gateway is an application for transferring files between partners using different protocols, file naming conventions, and file formats. It moves large and high-volume file transfers, with end-to-end visibility of file movement, in a process-oriented and highly-scalable framework. It also alleviates file transfer challenges, such as protocol and file brokering, automation, and data security.

Sterling File Gateway supports integration with Sterling B2B Integrator Mailbox, Sterling Control Center, Sterling Connect:Enterprise for UNIX server products, Sterling Secure Proxy and Sterling Connect:Direct. Sterling File Gateway is an add-on to the Sterling B2B Integrator platform with a unique application URL that provides single sign-on access to the Sterling B2B Integrator administrative console through menu selection.

Sterling File Gateway allows organizations to take complete control over file transfers with trading partners. Built on Sterling B2B Integrator, Sterling File Gateway offers a scalable architecture and a centralized file gateway with the capabilities necessary to monitor, administer, route, and transform high volumes of inbound and outbound files.

With Sterling File Gateway, the benefits of a standardized file transfer approach extend beyond the reliable and secure transmission of files with trading partners. A centralized gateway enables the consolidation of disparate file transfer activity. Intelligent routing and content-driven transformation capabilities help optimize file delivery processes. Subsequently, IT staff and users become more efficient, and platform consolidation helps reduce total cost of ownership.

2.3.1 How Sterling B2B Integrator and Sterling File Gateway work together

Within Sterling File Gateway, Sterling B2B Integrator is known as the *B2B Console* and is accessed from the Tools menu. Administrative functions, such as creating and managing user accounts, permission groups, and security keys for Sterling File Gateway, are handled in Sterling B2B Integrator.

Sterling File Gateway uses the following Sterling B2B Integrator communication adapters:

- ▶ FTP Server
- ▶ FTP Client
- ▶ SFTP Server
- ▶ SFTP Client
- ▶ HTTP Server
- ▶ HTTP Client
- ▶ Connect:Direct server
- ▶ Command Line adapter 2 (for PGP)

To install Sterling File Gateway, first install Sterling B2B Integrator. After installing Sterling File Gateway on an instance of Sterling B2B Integrator, future upgrades or new builds of Sterling B2B Integrator are included in the Sterling File Gateway upgrades and builds automatically. They are installed as part of the installation script.

2.3.2 Capabilities

Using the underlying Sterling B2B Integrator platform, the Sterling File Gateway has the following rich set of capabilities:

- ▶ Communication channels
 - Industry protocols, such as:
 - FTP
 - FTPS
 - SSH/SFTP
 - SSH/SCP
 - HTTP
 - HTTPS
 - Sterling Connect:Direct
 - Sterling Connect:Direct Secure Plus
 - WebDAV
 - SOAP
 - ODETTE
 - AS1, AS2, and AS3 support
 - EBICS (France) support
 - Extensible for custom protocols
 - Utilities to compress files
- ▶ Business processes
 - Create predefined event-driven business processes to minimize setup and administration of process flows.
 - Use rules-based processing to elevate file transfers to a business level activity.
- ▶ Security
 - Multiple encryption standards, such as PGP, SSL/TLS, and S/MIME
 - Single sign-on, LDAP, and user authentication
 - Encryption of both in-flight and at-rest files
 - Event logging for a complete audit trail of file transfer activities
- ▶ Management and visibility
 - myFileGateway provides trading partners secure access to initiate upload and download requests.
 - Choose from Internet Explorer, Safari, or Firefox.
 - Real-time monitoring and a self-service portal allows users visibility over in-flight file transfers.
 - Monitor file transfer activity on an *exception* basis using event management notifications.
 - Auditing and reporting provide metrics to verify regulatory compliance and adherence to service level agreements.

- ▶ Scalability
 - Native horizontal and vertical clustering support the consolidation of file transfer servers and growth in trading partner collaboration.
 - The technical architecture supports high volumes of file transfers and extremely large files without compromising performance.
- ▶ File processing and routing
 - Use mapping capabilities to manage file naming relationships.
 - Automate the replay, reprocess, and resend associated with failed file transfers.
 - Intelligently route files based on sender, file name, file type, and file contents.
 - Reusable templates reduce staff time to build and maintain file transfer processes.

With Sterling File Gateway, organizations can take advantage of reusable templates, create standardized processes, and use group-based controls to rapidly onboard and administer trading partner file transfers. Sterling File Gateway enables organizations to offer their trading partners a community self-service portal, event driven notifications, and support for industry-standard security and communication protocols.

2.3.3 Terminology

Table 2-2 contains terms that pertain to Sterling File Gateway that we use throughout this book.

Table 2-2 Sterling File Gateway terminology

Term	Definition
Arrived file	A message in a mailbox that Sterling File Gateway monitors that causes Sterling File Gateway to perform some activity on it.
Communication session	A record of the complete set of steps that are involved in a protocol level interaction with a remote client or a remote server, typically performed to facilitate a file transfer from connection to disconnection. Contains the authentication, authorization, file transfer, or non-file transfer records for all communication activities in which adapters participate, whether or not data actually is transferred.
Community	Represents a way to organize partners for the purposes of onboarding. In Sterling File Gateway, communities are used to limit or widen the selection of protocols available when creating partners. They are also used to enable listening or initiating modes of connection.
Consumer	Partner who receives files either directly delivered to them or delivered to a mailbox for them to retrieve.
Consumer file structure	Description of consumer expectations for file naming and format structure.
Dataflow	An aggregate of all documents that are related to each other by parent-child relationships, annotated with correlation entries and file transfer events.
Delivery	A record of the activities that Sterling File Gateway took to deliver a file to a specific consumer endpoint.

Term	Definition
Delivery channel	Consumer side of the routing channel that specifies a consumer file structure and a mailbox delivery destination. There can be more than one delivery channel for each routing channel.
Event	A distinct routing activity occurrence.
Event code	Generated for each activity during the progress of a file transfer. Displayed in the activity details to enable partners and operators to see the progress and to navigate to more details.
File layer	Description of format. A file can encapsulate one file or many other files. For example, a .zip file can contain a JPEG file. The .zip file is a container layer because it contains another file. The JPEG file is a non-container layer.
File structure	Description of a file's basic content structure and naming conventions. There are two types: <ul style="list-style-type: none"> ▶ Consumer file structure ▶ Producer file structure
Group	Usually refers to a partner group but can refer to a security group.
Integration Architect	Type of user who creates partner groups, communities, routing channel templates, producer file structures, and consumer file structures.
Mailbox	A repository for messages with a hierarchical structure similar to directories on Windows and UNIX.
Notification	Email sent to a subscriber when an event has occurred.
Mailbox Virtual Root	A position in the mailbox hierarchy that is associated with a user account that acts as the user's root directory.
myFileGateway	Web application portal where partners send and retrieve files.
Partner group	Represents a way to organize partners for purposes of applying templates that govern file transfer policies. Partner groups are separate and distinct from security groups, which are managed in Sterling B2B Integrator.
Partner	An organization such as a trading partner or a business unit. A partner can be a producer or a consumer or both a producer and a consumer.
Partner user	Type of user who uploads and downloads files from myFileGateway portal and works on behalf of the partner. Views his own activity, specifies events about which to receive notifications, and generates reports. Partner users are users that belong to a particular partner.
Payload	Either the arrived files (if non-container-type file layer) or files found inside a container type file layer.
Producer	Partner who creates and sends files.
Producer file structure	Description of producer requirements for file naming and structure.
Regular expression	An industry-standard pattern-matching language. Used in Sterling File Gateway to match the names of files from producers.

Term	Definition
Replay	The operation that an operator performs on a file to cause the system to reprocess that file again, as though it were sent again by the producer.
Route	A route is a record of all the activities performed on a payload after it is known whom the consumer is. Each payload is associated with a route. A replay of a route results in a new route and new file.
Route details	The details about a route, including the consumer, producer, list of deliveries that were attempted to the consumer, and events generated while processing the route, start and end times, any errors that occurred, and other details. Hyperlinks are provided to data flows, communication sessions, documents, and business processes that are related to the route.
Routing channel	<p>Matches incoming producer files to consumers according to the requirements in the routing channel template, then transforms and sends them to the correct consumer in the format and name specified in the consumer file structure.</p> <p>For a static routing channel, the consumer is explicitly identified during the configuration, and there is only one consumer.</p> <p>For a dynamic routing channel, no consumer is explicitly specified. The routing channel permits routing to any consumer belonging to the consumer group specified in the routing channel template.</p>
Routing channel template	The routing channel template (RCT) defines the structure through which routing occurs. The RCT specifies producer and consumer mailbox structures and file structures. It mandates which partners can participate in various file transfer scenarios and which file formats they must use. An RCT is required to create a routing channel, which establishes the producer-consumer relationship for file transfers.
Security group	Groups of users with the same privileges, as specified in Sterling B2B Integrator.
System administrator	Type of user who installs and maintains system software. Creates initial users. Configures services, adapters, perimeter servers, certificates, and the database for sending and receiving files.

Additional resource: For more information, see the Sterling File Gateway documentation, which is available at:

<http://help.sterlingcommerce.com/SFG20/index.jsp>

2.4 IBM Sterling Secure Proxy

Although organizations increasingly use the internet for file transfers, business communications over this unsecured channel are at risk. Sterling Secure Proxy is an application proxy that secures and shields a trusted network from external attacks by preventing direct communications between trading partners and internal servers. Sterling Secure Proxy provides DMZ-based authentication, session breaks, and SSL terminations

prior to allowing communications with the trusted network. It allows organizations to protect their trusted zone from unauthorized access by enforcing even tighter controls with multifactor authentication.

IT departments find it difficult to manage the need for tighter security as organizations increasingly use the internet as a lower-cost transmission channel for file transfers. Security experts are turning to *proxy* servers to shield applications and protocols in the trusted internal network from the public internet. In addition to encryption and authentication, organizations can implement a proxy server to terminate the file transfer session and to fully authenticate the trading partner before establishing a second session to the internal network. This feature prevents direct access to the organization's trusted zone and protects against increasing threats, like *man-in-the-middle* and *denial-of-service* attacks. Organizations need to protect their internal networks and file transfers from malicious attacks by incorporating the added security provided by proxy servers and a defense-in-depth strategy by utilizing multiple security layers.

Sterling Secure Proxy is used in this book in file transfer scenarios using the Sterling Connect:Direct protocol externally. Sterling Secure Proxy is the only mediation server available in the marketplace at the time of publication that supports the use of the Sterling Connect:Direct protocol. We do not discuss the details regarding the configuration of Sterling Secure Proxy in this book.

The Sterling Secure Proxy includes the following capabilities:

- ▶ Application proxy
 - Resides in the DMZ.
 - Supports Sterling Connect:Direct, Sterling Connect:Express, Sterling File Gateway, and Sterling B2B Integrator servers.
 - Provides support for multiple DMZ environments.
 - Supports FTP, FTPS, HTTP, HTTPS, SSH/SFTP, PeSIT, and Sterling Connect:Direct protocols.
- ▶ Firewall navigation best practices
 - Prevents inbound holes in the firewall.
 - Minimizes rich targets in the DMZ by ensuring that files, user credentials, and data are not stored in the DMZ.
 - Establishes sessions from more-trusted to less-trusted zones.
 - Enforces internal and external security policies.
- ▶ Perimeter security
 - Prevents direct communications between external and internal sessions by establishing secure session breaks in the DMZ using SSL or TLS.
 - Inspects protocol and sensitive control information, enabling configurable error handling for violations.
 - Provides session limits and data encryption to guard against denial-of-service attacks.
- ▶ Authentication Services
 - Customizable logon portal provides self-service password management for trading partners.
 - Supports single sign-on and integrates with existing security infrastructure, including Microsoft Active Directory and IBM Tivoli® user databases.

- Multifactor authentication enforces tight controls and validation of trading partner identity in the DMZ before information is passed to the trusted zone.
- Authentication options include IP address, user ID and password, digital certificates, SSH Keys, and RSA SecurID.
- ▶ Clustering
 - One central configuration manager pushes out configuration rules to multiple engines running in the DMZ, making it easy to scale.
 - Clustering for high availability and load balancing provides operational continuity and improved performance.

Sterling Secure Proxy promotes rapid expansion of an organization's trading partner community by allowing them to use the internet for secure file transfer. Sterling Secure Proxy provides secure internet communications for their file transfer infrastructure by blocking common URL exploits, using command filters, and providing configurable handling of protocol violations. In addition, it allows organizations to significantly reduce partner onboarding time and costs by consolidating security management in a single proxy while the customizable logon portal enables partners' self-service password administration. Authentication options allow organizations to integrate with existing security infrastructure to centralize user credentials and control user management costs. It is a highly scalable solution ideal for supporting rapid growth in a trading partner community.

Sterling Secure Proxy increases perimeter security for an organization's file transfer infrastructure. This increased security makes it easier to meet strict security audits and to comply with corporate requirements, industry mandates, and government regulations. Sterling Secure Proxy incorporates firewall navigation best practices to prevent direct communications between internal and external servers and exposes attacks at the perimeter.

2.5 WebSphere MQ File Transfer Edition

WebSphere MQ File Transfer Edition provides an enterprise-ready, managed file transfer capability that is both robust and easy-to-use. WebSphere MQ File Transfer Edition exploits the proven reliability and connectivity of WebSphere MQ to transfer files across a wide range of platforms and networks. In addition to using existing WebSphere MQ networks, you can integrate WebSphere MQ File Transfer Edition with existing file transfer systems.

WebSphere MQ File Transfer Edition offers the following benefits:

- ▶ Auditability

WebSphere MQ File Transfer Edition provides full logging of transfers at both the source and destination systems. File transfer audit logs are stored in WebSphere MQ queues and optionally in a relational SQL database.
- ▶ Ease-of-use

Using WebSphere MQ File Transfer Edition, you can initiate file transfers using the graphical user interface in WebSphere MQ Explorer, command-line commands, and scripts.
- ▶ Simplicity

WebSphere MQ File Transfer Edition has a low resource footprint and, apart from WebSphere MQ, has no other prerequisite software.

- ▶ Security

Access to files is controlled by file-system permissions. File transfers can be protected using SSL encryption and authentication.
- ▶ Automation

File transfers can be set up to occur at specified times or dates or can be repeated at specified intervals. File transfers can also be triggered by a range of system events, such as new files or updated files.

2.5.1 Architecture of WebSphere MQ File Transfer Edition

WebSphere MQ File Transfer Edition includes the following components that are all supported by one or more WebSphere MQ queue managers in the network:

- ▶ *FTE agents* are programs that perform the fundamental file transfer function (for example, they send and receive files from the local system).
- ▶ *Configuration commands* are used to control FTE from a command line. Configuration commands perform tasks, such as creating and deleting agents.
- ▶ *Administration commands* perform tasks, such as creating new file transfers.
- ▶ The *graphical user interface* provides a point-and-click graphical interface to configure and administer FTE.
- ▶ A *Database Logger* sends the contents of file transfer log messages to a database.

The components of WebSphere MQ File Transfer Edition use WebSphere MQ to communicate with each other, and the agents in particular use WebSphere MQ to transport the contents of files across the network to other agents (Figure 2-3).

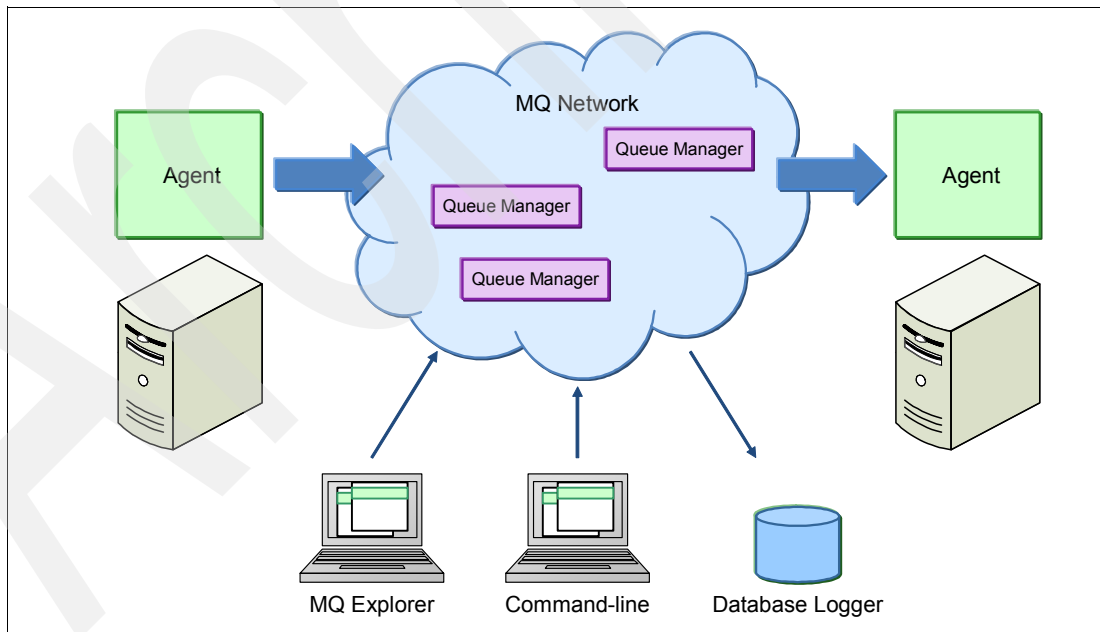


Figure 2-3 Overview of WebSphere MQ File Transfer Edition architecture

WebSphere MQ File Transfer Edition agents

Agents are Java-based MQ programs that form the endpoints for file transfer operations. Essentially, agents perform the fundamental task of transferring files across the network using the WebSphere MQ network as the back bone. When requested to send a file, an agent

reads the file's contents and sends it to the destination agent in the form of MQ messages. Often, those messages are carried by a WebSphere MQ *channel* across the network where another agent receives them. The receiving agent re-assembles the file on the destination system. There must be an FTE agent running on each host system that can transfer files to or from other systems.

A single agent can process more than one file transfer concurrently, and concurrent transfers might be to the same or different destination agents.

Agents use the WebSphere MQ network to send file information, so every agent needs a queue manager, which is called an *agent queue manager*. An agent queue manager can host more than one agent because each agent uses its own queues, which are separate from the queues that other agents use.

There are two types of FTE agents that correspond to the IBM WebSphere MQ File Transfer Edition *Server* product and the IBM WebSphere MQ File Transfer Edition *Client* product:

- ▶ **Server Edition agent**

The agent supplied with the FTE Server edition product can connect to a local queue manager using an *MQ bindings connection*. These agents can also connect to a local or remote queue manager using an *MQ client connection*.

- ▶ **Client Edition agent**

The agent that is supplied with the FTE Client edition product uses an *MQ client connection* to connect to a queue manager. Client agents can be located on the same system or on a different system than their agent queue manager.

Graphical user interface

You can administer WebSphere MQ File Transfer Edition with the MQ Explorer workbench, using the GUI plug-in. The GUI plug-in is part of the WebSphere MQ File Transfer Edition Remote Tools and Documentation product. WebSphere MQ Explorer is available for Windows and Linux platforms, is supplied with WebSphere MQ, and is available in stand-alone form with WebSphere MQ MS0T SupportPac.

Figure 2-4 shows the WebSphere MQ Explorer views for managing WebSphere MQ File Transfer Edition.

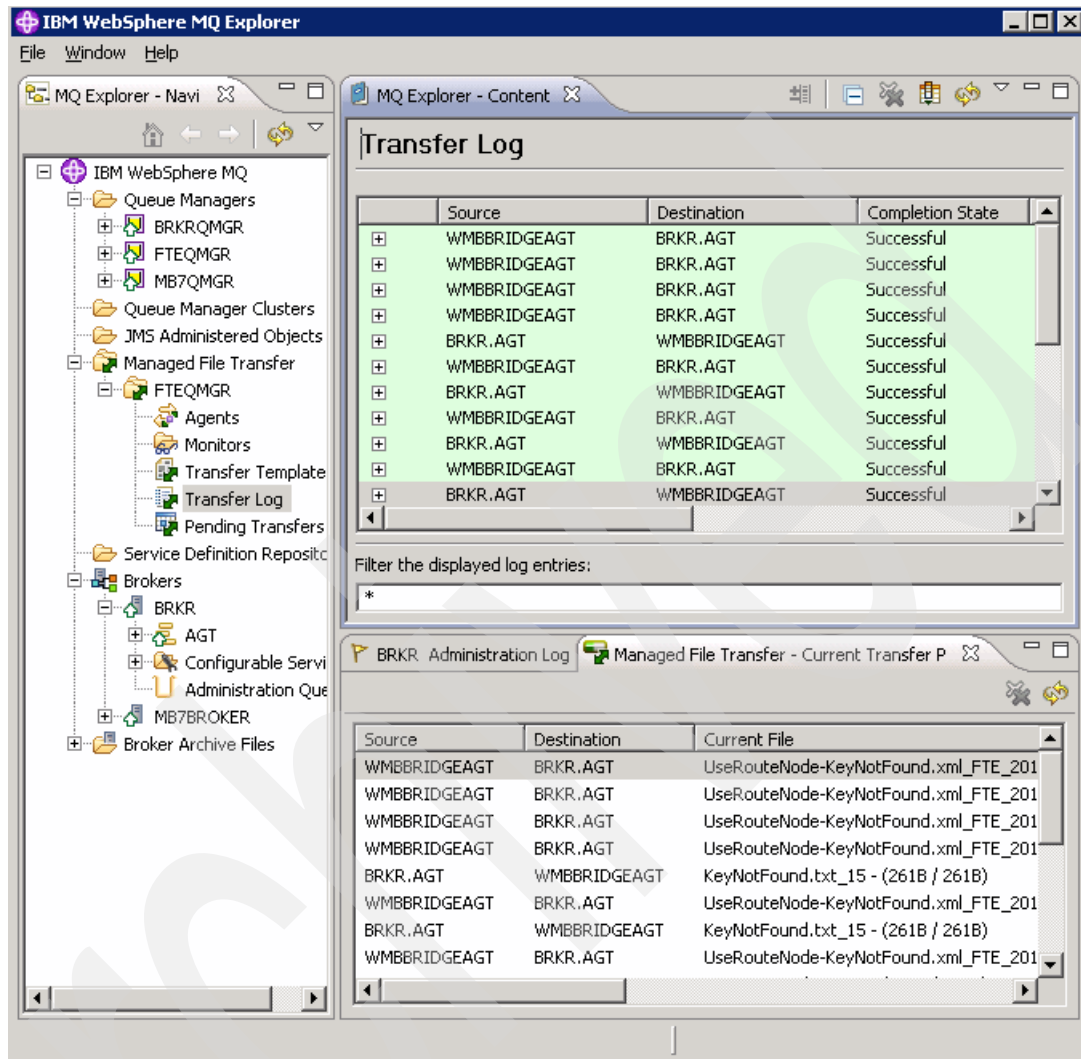


Figure 2-4 Administering WebSphere MQ File Transfer Edition using WebSphere MQ Explorer

Command-line tools

Use the command-line tools to configure WebSphere MQ File Transfer Edition and to operate it (for example, to submit and monitor file transfer requests). You can use the following configuration commands to set up WebSphere MQ File Transfer Edition when you create and configure a new installation:

- ▶ **fteCreateAgent**
- ▶ **fteDeleteAgent**
- ▶ **fteChangeDefaultConfigurationOptions**
- ▶ **fteSetupCoordination**
- ▶ **fteSetupCommands**
- ▶ **fteCreateBridgeAgent**
- ▶ **fteCreateWebAgent**

You can use the following administration commands to operate WebSphere MQ File Transfer Edition and to support tasks that are typically performed day-to-day in an installation:

- ▶ **fteStartAgent**
- ▶ **fteCreateTransfer**
- ▶ **fteCreateMonitor**
- ▶ **fteDeleteMonitor**
- ▶ **fteListAgents**
- ▶ **fteShowAgentDetails**
- ▶ **fteListScheduledTransfers**
- ▶ **fteDeleteScheduledTransfer**
- ▶ **fteCleanAgent**
- ▶ **fteStopAgent**
- ▶ **fteCancelTransfer**
- ▶ **fteSetAgentTraceLevel**
- ▶ **fteListMonitors**
- ▶ **fteAnt**
- ▶ **ftePingAgent**
- ▶ **fteStartDatabaseLogger**
- ▶ **fteStopDatabaseLogger**
- ▶ **fteCreateTemplate**
- ▶ **fteDeleteTemplates**
- ▶ **fteListTemplates**
- ▶ **fteDisplayVersion**
- ▶ **fteModifyAgent** (Windows Only)
- ▶ **fteModifyDatabaseLogger** (Windows Only)

Queue managers

WebSphere MQ File Transfer Edition uses WebSphere MQ to communicate between its agents, the WebSphere MQ Explorer plug-in, and the command-line commands. In addition, WebSphere MQ File Transfer Edition uses WebSphere MQ to transmit bulk file data between agents.

To do its job, each component needs to connect to a WebSphere MQ queue manager, of which there are the following roles:

- ▶ Coordination queue manager
- ▶ Agent queue managers
- ▶ Command queue managers

WebSphere MQ File Transfer Edition does not require that all three roles be physically separate queue managers, although there are usually good reasons to design your installation that way. A simple installation can designate a single queue manager to fill all three roles, but doing this setup in a production environment is not ideal from a performance and reliability point-of-view.

Production environments are best designed using separate coordination and agent queue managers.

The coordination queue manager

The coordination queue manager acts as a central collection point where information about file transfer activity is gathered. A WebSphere MQ File Transfer Edition network typically has a separate designated queue manager to be the coordination queue manager. Agents publish active file transfer status information to a topic that is hosted on this queue manager. Additionally, the coordination queue manager broadcasts file transfer audit information to

other components and to any interested and authorized parties who subscribed to WebSphere MQ File Transfer Edition information topics.

The coordination queue manager's primary role is to collect information about the network, and unless the coordination queue manager is also hosting WebSphere MQ File Transfer Edition agents, it does not participate in the transmission of file data (the agent queue managers perform that duty). Of course, it is possible to define a single queue manager that fills both the coordination queue manager role and the agent queue manager role, and in that case, the coordination queue manager also carries file data.

WebSphere MQ File Transfer Edition requires that the coordination queue manager be hosted using a WebSphere MQ V7 or later installation. Additionally, the coordination queue manager must be enabled for WebSphere MQ publish/subscribe.

The agent queue manager

Each agent connects to its agent queue manager and through it receives file transfer requests and publishes its own file transfer start and stop status events to the coordination queue manager. An agent queue manager hosts the queues that are used by the agents that it supports. Each agent uses its own uniquely-named set of queues so that an agent queue manager can support one or more *server agents* on its local system in addition to one or more *client agents* on remote systems.

The command queue manager

The command-line tools and the WebSphere MQ File Transfer Edition MQ Explorer GUI plug-in use the command queue managers to communicate with agents.

2.5.2 Using Apache Ant

Ant is an XML-based scripting tool, released by the Apache Software Foundation, that is widely used for building Java-based software suites. Although its original purpose was to manage building Java software, Ant is becoming popular as a general-purpose scripting tool. WebSphere MQ File Transfer Edition can integrate its file transfer functions using scripts that are run by Ant.

Ant accepts a script file that is coded in XML. Within the XML script are verbs, known as Ant *tasks*, that represent the actions that the script will perform. Ant itself provides many hundreds of tasks to address a wide range of scripting needs.

WebSphere MQ File Transfer Edition provides its own set of Ant tasks that can be used to integrate file transfer processing within an Ant script. The WebSphere MQ File Transfer Edition tasks can be combined with any of the other Ant tasks to address more complex file management needs.

WebSphere MQ File Transfer Edition provides the following Ant tasks:

- ▶ awaitoutcome
- ▶ call
- ▶ cancel
- ▶ filecopy
- ▶ filemove
- ▶ ignoreoutcome
- ▶ ping
- ▶ uuid

A number of the Ant tasks that WebSphere MQ File Transfer Edition provides use nested XML elements to further qualify the operations.

2.5.3 Using file transfer pre-processing and post-processing tasks

When you configure WebSphere MQ File Transfer Edition to send and receive files, it is possible to have WebSphere MQ File Transfer Edition run a task both before and after the transfer occurs. Pre-processing tasks are executed before the file transfer, and post-processing tasks are executed after the transfer.

Additionally, you can configure pre-processing and post-processing tasks for either or both the source agent and the destination agent.

2.5.4 Using WebSphere MQ Advanced Message Security

WebSphere MQ security for client applications, such as FTE client agents, relies on SSL or channel exits to authenticate connections. WebSphere MQ Advanced Message Security extends these capabilities by signing or encrypting messages to provide an additional layer of security. Whereas SSL channels encrypt the messages in transit on the network, WebSphere MQ Advanced Message Security encrypts them end-to-end, including while at rest on the queues. In addition, WebSphere MQ Advanced Message Security can enforce policies to ensure that only messages from authorized senders can be consumed by the FTE agent.

WebSphere MQ Advanced Message Security is a separately available product. For more information, see the product home page at:

<http://www-01.ibm.com/software/integration/wmq/advanced-message-security/>

2.6 WebSphere Message Broker

WebSphere Message Broker is a platform-independent based ESB that provides universal connectivity. It can be used to integrate disparate applications and is designed to transform various formats of data between any type of applications using a number of supported communications protocols or distribution methods. It is used where there is a need for high-performance and complex integration patterns.

2.6.1 Message flows with WebSphere Message Broker

Processing logic in WebSphere Message Broker is implemented using *message flows*. Through message flows, messages from business applications can be transformed and routed to other business applications. Message flows are created by connecting *nodes* together. A wide selection of built-in nodes is provided with WebSphere Message Broker. These nodes perform tasks that are associated with message routing, transformation, and enrichment. The base capabilities of WebSphere Message Broker are enhanced by SupportPacs that provide a wide range of additional enhancements.

Message routing

Packaged with WebSphere Message Broker is a variety of nodes through which connectivity is provided for both standards-based and non-standards-based applications and services. Routing can be point-to-point or based on matching the content of the message with a pattern that is specified in a node.

Aggregation is an advanced form of message routing. With aggregation, a request message is received, and multiple new request messages are generated. Each new message is routed to its destination using a request-reply interaction. WebSphere Message Broker tracks the process, collecting all responses and recomposing them into a single output message.

Transport protocol conversion

WebSphere Message Broker provides universal connectivity between applications that use disparate transport protocols. WebSphere Message Broker enables connectivity between applications or business processes that use transport protocols, such as web services (SOAP, REST), HTTP(S), Java Message Service (JMS), WebSphere MQ, CICS®, IMS™, TCP/IP, FTP, SCA, EIS (SAP, Siebel, and PeopleSoft), and user-defined transports.

WebSphere Message Broker supports integration with WebSphere Business Adapters. For more information about available adapters, see the WebSphere Adapters page at:

<http://www-01.ibm.com/software/integration/wbiadapters/>

Message transformation and enrichment

One of the key capabilities of WebSphere Message Broker is the transformation and enrichment of in-flight messages. This capability enables business integration without the need for any additional logic in the applications. For example, an application that generates messages in a custom format can be integrated with an application that recognizes only XML. This capability provides a powerful mechanism to unify organizations because business information can now be distributed to applications that handle completely separate message formats.

WebSphere Transformation Extender can be integrated into the WebSphere Message Broker ESB solution to extend the existing capabilities and to simplify transformation development.

2.6.2 Runtime architecture of WebSphere Message Broker

WebSphere Message Broker consists of a development environment on which message flows and message sets are designed and developed and of a runtime environment on which the message flows executes.

The *broker* is a set of application processes that host and run message flows. When a message arrives at the broker from a business application, the broker processes the message before passing it on to one or more other business applications. The broker routes, transforms, and manipulates messages according to the logic that is defined in its message flow applications. Each broker uses an internal repository on the local file system to store the message flows, configuration data, and the message sets that are deployed to it.

Execution groups are processes that host message flows. The execution groups facilitate the grouping of message flows within the broker with respect to functionality, load balancing, or other qualifications that are determined to be necessary.

2.6.3 Developing message flows with the WebSphere Message Broker Toolkit

The WebSphere Message Broker Toolkit is an integrated development environment (IDE) and GUI based on the Eclipse platform. Application developers use the WebSphere Message Broker Toolkit. Using it, they can create message flows and the associated artifacts and deploy them to the execution groups.

The Broker Application Development perspective is the default perspective that is displayed the first time that you start the WebSphere Message Broker Toolkit. Application developers work in this perspective to develop and modify message sets and message flows.

2.6.4 Deploying message flow applications

Message flow applications contain message flows and the message sets that are included in the message definitions that are used to model the messages within the message flows. Message flow applications can be deployed to the execution groups of the brokers by first adding the components of the message flow application to a broker archive file (bar file). Then, you can deploy the bar file to the broker's execution group. You can deploy the bar files using the command line or with the WebSphere Message Broker Toolkit. Alternatively, WebSphere Message Broker Toolkit provides the capability to deploy the message flows directly to an execution group without first adding them to a bar file, which results in the deployment of all of the message-flow-dependent resources.

2.6.5 Administration with WebSphere Message Broker Explorer

The WebSphere Message Broker Explorer is a tool for administrators and provides the capability for enhanced WebSphere Message Broker monitoring and management. WebSphere Message Broker Explorer is installed as a plug-in for WebSphere MQ Explorer. The WebSphere Message Broker view is added to the WebSphere MQ Explorer-Navigator pane. The broker administration tasks can then be performed from this view. Using the WebSphere Message Broker Explorer, the user can administer brokers and WebSphere MQ queue managers in the same toolkit.

In addition to the WebSphere Message Broker Explorer, there are command-line and API-based utilities that you can use to accomplish broker administration tasks. For example, the CMP API Exerciser sample that WebSphere Message Broker provides uses the Java administration API. You can use the CMP API Exerciser to view and manage brokers and their execution groups. You can also use it to create, modify, and delete configurable services and general broker administration tasks. The Java administration API is also available to users for scripting broker administration tasks.

Archived



Scenario topology overview

This book includes four file transfer scenarios, with additional variations, that illustrate how you can build a file transfer solution. The scenarios focus on the integration of IBM Sterling managed file transfer products with WebSphere MQ File Transfer Edition. They also demonstrate how Sterling File Gateway can work with HTTP and Secure File Transfer Protocol (Secure FTP) to move files in a multi-enterprise environment through the various network zones that exist in many organizations.

Each scenario addresses specific needs and environmental considerations. This chapter introduces these scenarios. It shows the systems topology used in our lab environment, the location of the products within that topology, and basic information about the configuration. Each scenario uses a subset of the components in the topology.

3.1 An introduction to the scenarios used in this book

The scenarios in this book illustrate how to design file transfer solutions that address a variety of situations. Each scenario uses technology to suit specific needs and, in certain cases, variations of the scenario are shown. You can use these solution designs as they are, or you can use them as the basis to design your own solution.

This section provides a basic overview of the scenarios that we use in this book.

3.1.1 Internal use of Sterling Connect:Direct and WebSphere MQ File Transfer Edition

This scenario illustrates how to use only Sterling Connect:Direct and how to integrate Sterling Connect:Direct with WebSphere MQ File Transfer Edition for internal managed file transfer. The Connect:Direct nodes and WebSphere MQ File Transfer Edition agents reside in the protected network. This scenario uses features in both products to integrate the two proprietary protocols.

The scenario addresses the flow of files from the following perspectives:

- ▶ Pushing a file from a Sterling Connect:Direct PNODE to a Sterling Connect:Direct SNODE
- ▶ Sterling Connect:Direct PNODE pulling a file from a Sterling Connect:Direct SNODE
- ▶ Pushing a file from Sterling Connect:Direct to WebSphere MQ File Transfer Edition
- ▶ Sterling Connect:Direct pulling a file from WebSphere MQ File Transfer Edition
- ▶ WebSphere MQ File Transfer Edition pushing a file to Sterling Connect:Direct
- ▶ WebSphere MQ File Transfer Edition pulling a file from Sterling Connect:Direct

3.1.2 Using Sterling Connect:Direct for multi-enterprise transfers

This scenario illustrates how to use Sterling Connect:Direct for multi-enterprise file transfers. The scenario features Connect:Direct nodes internal and external to the enterprise. Any external Sterling Connect:Direct transmission should use encryption of the payload.

This scenario uses the following topologies:

- ▶ In the first topology, Sterling Secure Proxy sits in the DMZ. Sterling Connect:Direct moves files into and out of the protected network through Sterling Secure Proxy. Sterling Secure Proxy is used as the mediation server that resides in the DMZ for this particular scenario, because it is the only product that is available at the time of publication that supports the use of the proprietary Sterling Connect:Direct protocol.
- ▶ In the second topology, files are transferred between the external Connect:Direct node and the protected network using Sterling File Gateway. Sterling Secure Proxy resides in the DMZ to allow for a secure external transmission. The addition of Sterling File Gateway to a Sterling Connect:Direct multi-enterprise file transfer scenario allows the external partner to have the ability to view transfers through a partner management interface and the ability to convert to other protocols if necessary.

The topologies address the flow of files from the following perspectives:

- ▶ Using Sterling Connect:Direct and Sterling Secure Proxy
 - Pushing a file from an external Sterling Connect:Direct PNODE through Sterling Secure Proxy to an internal Sterling Connect:Direct SNODE.
- ▶ Using Sterling Connect:Direct and Sterling File Gateway
 - Pushing a file from an external Connect:Direct node through Sterling Secure Proxy to Sterling File Gateway in the protected network, using a Sterling Connect:Direct adapter to receive the file and to send to another internal Connect:Direct node.
 - Pushing a file from an internal Sterling Connect:Direct to Sterling File Gateway, using a Sterling Connect:Direct adapter, which routes it to an external Connect:Direct node through Sterling Secure Proxy.

3.1.3 Multi-Enterprise file transfer using Sterling Connect:Direct, Sterling File Gateway, and WebSphere MQ File Transfer Edition

This scenario illustrates how files can be exchanged between enterprises using Sterling Connect:Direct and WebSphere MQ File Transfer Edition using Sterling File Gateway. Sterling Connect:Direct is used externally to send files through a mediation server in the DMZ. Any external Sterling Connect:Direct transmission should use encryption of the payload.

Sterling File Gateway takes advantage of Sterling B2B Integrator's ability to handle both the Sterling Connect:Direct and WebSphere MQ protocols. Through mailboxes and routing channels defined in Sterling File Gateway, Sterling File Gateway uses trading partner definitions in Sterling B2B Integrator to switch between protocols. Sterling B2B Integrator uses a WebSphere MQ adapter to place a message on a queue for an agent to initiate a transfer. The WebSphere MQ FTE backbone inside the protected network is used to move files securely between back-end systems.

This scenario addresses both inbound and outbound file transfers.

3.1.4 Integrating multi-enterprise transfers with an enterprise service bus

This scenario illustrates how to transfer files with external partners using application layer protocols with Sterling File Gateway, while at the same time taking advantage of the business-to-business partner profile capabilities in Sterling File Gateway. In addition, it illustrates how you can use an enterprise service bus (ESB) in the protected network to affect the file transfer. WebSphere Message Broker provides the ESB capabilities, allowing you to route data to the appropriate back-end application, to transform the transfer request to the protocol required for the target application, or to perform more advanced mediation using the wide range of capabilities of WebSphere Message Broker.

This scenario addresses both inbound and outbound file transfers.

3.2 Scenario architecture

The product components that we use throughout this book reside on multiple systems that are located in specific network locations within and external to the enterprise. Each file transfer solution is involved in the following network zones:

- ▶ The external partner, *Company A*
- ▶ The protected network, *Company B*, which contains the applications and data that belongs to the enterprise
- ▶ The DMZ, which acts as a security buffer between Company A and Company B

The zones are designed to simulate the common zones found in organizations. Figure 3-1 depicts a view of the complete scenario architecture.

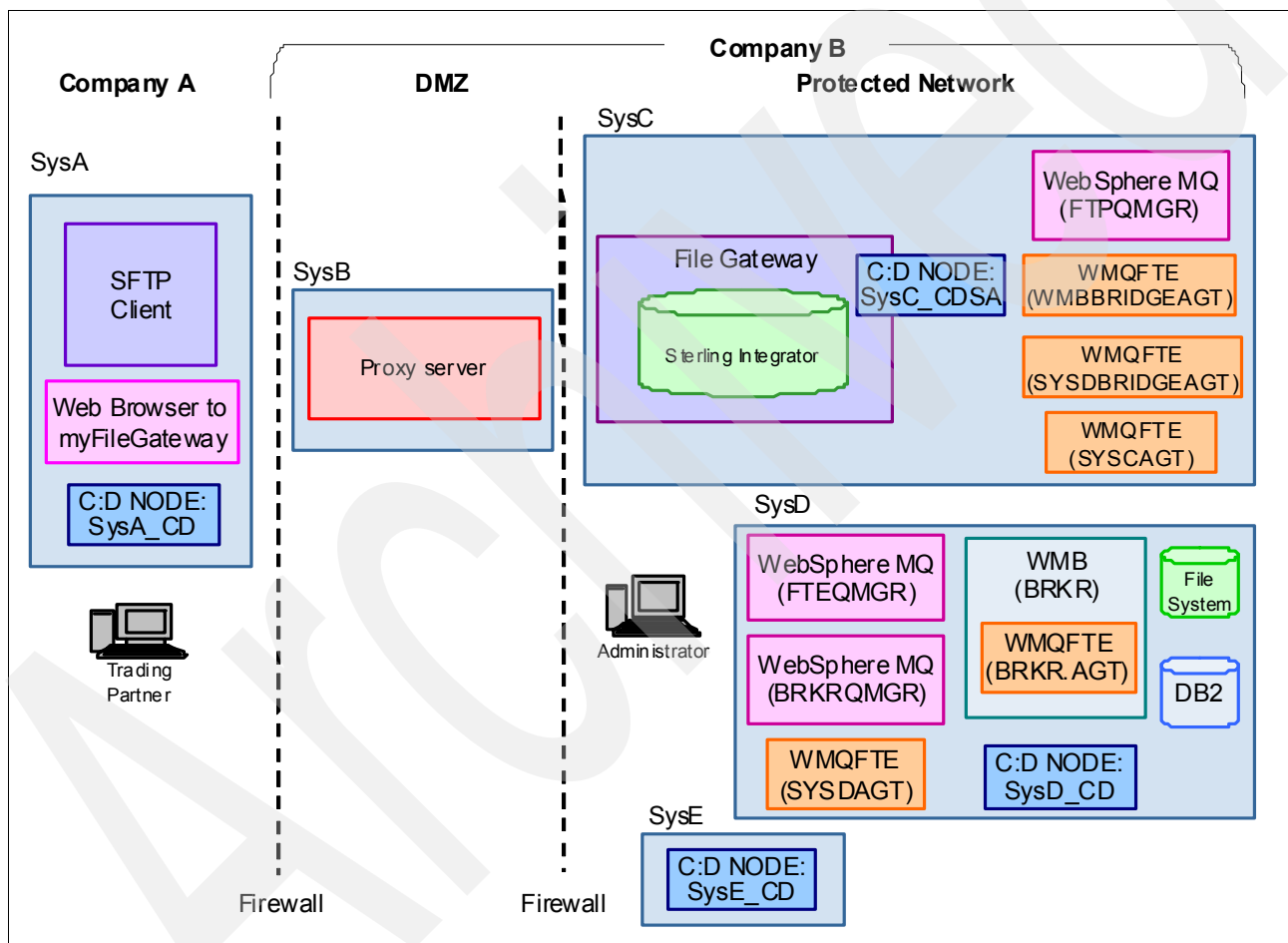


Figure 3-1 Multi-enterprise file transfer

In this chapter, we discuss the systems and components shown in Figure 3-1 at a high level and include basic implementation information where appropriate. Because specific alterations are made to the environment for each file transfer scenario, we discuss the specific modification or implementation details made to that system in the corresponding file transfer scenario chapter. Many of the middleware components reside on separate systems but use the same installation and configuration steps.

The following sections provide basic information about the topology configuration of each zone.

3.2.1 The external partner: Company A

In this book, *Company A* refers to anything outside Company B's span of control. We have placed a Secure FTP client, a Connect:Direct node, and an HTTP client at Company A to represent file transfer partners with unknown environments and networks. The Secure FTP client can be any publicly available Secure FTP client of your choosing.

An external partner is an independent entity that trades data with an entity in the protected network. The trading can take the form of business-to-consumer, business-to-business, or business-to-government transfers. For the purposes of this book, the *external partner* represents any location to which you move a file and over which you have no control. This location can be inside enterprise firewalls in a separate department or outside the firewalls at a separate organization.

Secure FTP client

For certain operating systems, the Secure FTP client is part of the base operating system. For our purposes, we used a freely available Secure FTP client on a Microsoft Windows server.

Sterling Connect:Direct

The Connect:Direct node is designated as both a *primary* node (PNODE) and a *secondary* node (SNODE), depending on the flow of the file transfer. As a PNODE, Sterling Connect:Direct originates the file transfer in the Connect:Direct process language. As a SNODE, the Connect:Direct node is designated to receive the file transfer in the Connect:Direct process language. The process language jobs can be built in the Connect:Direct requester on Windows and UNIX systems. The requester is used from a desktop to monitor jobs, view statistics logs, and manage jobs. Created for centralized management, the Sterling Connect:Direct Control Center, a desktop application, can also be used for these same administrative and development purposes. Control Center contains the ability to pull logs files automatically from multiple, disparate systems and to generate reports.

HTTP client

Simply stated, the HTTP client is a web browser. This browser can be any web browser available today. The web browser is on a desktop where it can read or written to the file system using HTTP verbs.

3.2.2 The protected network: Company B

The following components are hosted in the protected network:

- ▶ Sterling File Gateway
- ▶ Sterling B2B Integrator
- ▶ Sterling Connect:Direct
- ▶ WebSphere MQ
- ▶ WebSphere Message Broker
- ▶ WebSphere MQ File Transfer Edition

Sterling File Gateway

Sterling File Gateway is used in many of the multi-enterprise file transfer scenarios. Used for transferring files between partners using different protocols, Sterling File Gateway allows trading partners to choose protocols for data exchange independently. Sterling File Gateway allows for movement of large and high-volume file transfers with end-to-end visibility of the file movement. Through the use of myFileGateway, Sterling File Gateway allows business

partners to view file transfers for better partner management. Sterling File Gateway uses Sterling B2B Integrator under its covers to perform various functions.

Sterling B2B Integrator

Sterling B2B Integrator is used under the covers of Sterling File Gateway for connectivity to back-end adapters, such as WebSphere MQ. Also providing transport security, Sterling B2B Integrator is used to exchange keys to allow for Secure FTP communication. Sterling B2B Integrator also allows for the use of B2B communication protocols, customizable partner configuration, a variety of adapters for back-end integration, and the ability to create, manage, and grow partner communities.

Sterling Connect:Direct

Connect:Direct nodes are designated as both a PNODE and a SNODE, depending on the flow of the file transfer. As a PNODE, Sterling Connect:Direct originates the file transfer in the Connect:Direct process language. As an SNODE, the Connect:Direct node is designated to receive the file transfer in the Connect:Direct process language. The process language jobs can be built in the Connect:Direct requester on Windows and UNIX systems. The Requester is used from a desktop to monitor jobs, view statistics logs, and manage jobs. Created for centralized management, the Sterling Connect:Direct Control Center, a desktop application, can also be used for these same administrative and development purposes. Control Center has the ability to pull logs files automatically from multiple, disparate systems and to generate reports.

WebSphere MQ

WebSphere MQ is a messaging product that is available on a wide range of platforms. In these scenarios, WebSphere MQ provides the messaging network for WebSphere MQ File Transfer Edition (WebSphere MQ FTE). For information about how we configured the WebSphere MQ infrastructure required for the WebSphere MQ FTE backbone, see “Configuring WebSphere MQ” on page 349.

WebSphere Message Broker

WebSphere Message Broker is a powerful information broker that allows business data, in the form of messages, to flow between disparate applications and across multiple hardware and software platforms. Rules can be applied to the data that is flowing through the message broker to route, store, retrieve, and transform the information.

WebSphere Message Broker provides a choice of transports that enable secure business to be conducted at any time by providing powerful integration, message, and data transformations in a single place. WebSphere Message Broker is built on WebSphere MQ. Therefore, it supports MQ transport. However, it also supports other transports, such as HTTP/HTTPS, SOAP, file, TCP/IP, EIS, and others that do not use MQ stack.

WebSphere Message Broker is used in multi-enterprise file transfer to transform and route files. For more information about its use and configuration, see Chapter 7, “External transfers using IBM WebSphere Message Broker and IBM Sterling File Gateway” on page 245.

WebSphere MQ File Transfer Edition

WebSphere MQ File Transfer Edition (WebSphere MQ FTE) is used in every scenario to provide a WebSphere MQ FTE backbone for files to traverse on their way to their destination. The agents reside exclusively in the protected network and are used to push and pull files from Sterling Connect:Direct or Sterling File Gateway. For information about how the WebSphere MQ File Transfer Edition agents are configured, see “Configuring WebSphere MQ File Transfer Edition” on page 356.

The topologies using WebSphere MQ File Transfer Edition extend and build on the basic single queue manager WebSphere MQ File Transfer Edition topologies that are discussed in *Getting Started with WebSphere MQ File Transfer Edition V7*, SG24-7760. Any of the WebSphere MQ File Transfer Edition topologies depicted in that book work with multi-enterprise file transfers.

Additionally, you can expand or couple the topologies shown in this book with the file transfer topologies discussed in *Multi-Enterprise File Transfer with WebSphere Connectivity*, SG24-7886.

3.2.3 DMZ

The DMZ hosts technologies that are designed specifically for placement in a DMZ. In our scenarios, we generally suggest the use of a server designed for DMZ usage such as an edge server, web server, or another form of a mediation server. In the scenarios using Sterling Connect:Direct externally, we highlight the usage of Sterling Secure Proxy because it is the only known mediation server that supports the Sterling Connect:Direct protocol that is available at the time of publication.

DMZ systems

The systems in the DMZ provide the ability to connect to external partners using Sterling Connect:Direct, FTP, Secure FTP, and HTTP servers or services. Systems in the DMZ are typically used as intermediaries for requests from clients. They forward the requests or files to and from servers inside the protected network zone. These systems are commonly referred to as *edge servers* and can be proxies, web servers, messaging servers, proprietary connectors, and even email servers.

Sterling Secure Proxy

Sterling Secure Proxy in this book acts as an application proxy between Connect:Direct nodes or between a Connect:Direct node and the Sterling File Gateway server. Providing a high level of data protection between the external connections and an internal network, Sterling Secure Proxy creates a session break in SSL sessions. This level of protection allows an organization to create firewall rules to prevent trading partners from obtaining direct access to protected, back-end systems.

Archived

Managed file transfer within an enterprise

This chapter demonstrates how file transfers within the enterprise can add value within an organization. In many companies, inter-departmental file transfers are a critical part of day-to-day operations. The type of data that is transferred between departments can include files that contain any type of data, including text, EDI, binary, digital content, and images. Automating the transfer of this data improves the productivity and reliability of new or existing business processes.

This chapter includes multiple scenarios that transfer files using WebSphere MQ File Transfer Edition and Sterling Connect:Direct. The managed file transfer solutions that we outline represent scenarios that might occur within an internal network or a single company. We also outline the integration of WebSphere MQ File Transfer Edition and Sterling Connect:Direct to demonstrate how an organization can combine existing or new implementations. The examples demonstrate that combining WebSphere MQ File Transfer Edition and Sterling Connect:Direct can provide end-to-end solutions for predictability, security, performance, and automation.

This chapter discusses the following topics:

- ▶ Solution overview
- ▶ Scenario details
- ▶ Configuring the solution components
- ▶ Testing the flows
- ▶ Troubleshooting tips

4.1 Solution overview

This chapter is an introduction to the basic benefits, concepts, and components of managed file transfer using IBM Sterling Connect:Direct and WebSphere MQ File Transfer Edition. These scenarios demonstrate how a file can be transferred from one system to another, either WebSphere MQ File Transfer Edition to Sterling Connect:Direct or Sterling Connect:Direct to WebSphere MQ File Transfer Edition.

Connect:Direct servers receive instructions about the work to be performed from Connect:Direct clients. The clients communicate with the servers through a command-line interface (CLI) or a desktop client, called a Connect:Direct requester. The client uses the Connect:Direct process language to provide instructions to the server. A process contains special statements and parameters that instruct the server to perform data movement by running the jobs or programs.

This chapter includes examples of uploading and downloading a file using WebSphere MQ File Transfer Edition and Sterling Connect:Direct. When a file is transferred to Sterling Connect:Direct, the Connect:Direct process instructs the Connect:Direct server to launch a WebSphere MQ File Transfer Edition transfer. Likewise, after a file is transferred to a WebSphere MQ File Transfer Edition agent, an ANT job launches a Connect:Direct process. With these solutions, organizations can connect the WebSphere and Sterling Connect:Direct infrastructures.

4.1.1 Using Sterling Connect:Direct and WebSphere MQ File Transfer Edition

The examples in this chapter simulate managed file transfers that occur on an internal network. For our purposes, an *internal network* is one that does not require additional software or hardware security devices. Any data that is sent to or received from an external company should always be encrypted. The examples in this chapter do not show the use of encryption, because the scenarios are examples of internal file transfers.

The examples demonstrate WebSphere MQ File Transfer Edition and Sterling Connect:Direct integration by launching a program or an ANT job. Sterling Connect:Direct uses the Connect:Direct process language to transfer a file and then executes `fteCreateTransfer.cmd` to start the WebSphere MQ File Transfer Edition transfer. The `fteCreateTransfer.cmd` program is included as part the installation of WebSphere MQ File Transfer Edition. WebSphere MQ File Transfer Edition uses an ANT job to transfer a file and then launches a Connect:Direct process to start the Sterling Connect:Direct transfer.

The scenarios make use of three separate systems. On one system, both WebSphere MQ File Transfer Edition and Sterling Connect:Direct must be installed to enable both applications local access to the tools that are required to launch a transfer on the other application. Access to `fteCreateTransfer.cmd` is required for Sterling Connect:Direct to start a WebSphere MQ File Transfer Edition transmission. Likewise, access to the Sterling Connect:Direct CLI is required for WebSphere MQ File Transfer Edition to start a Connect:Direct process.

WebSphere MQ File Transfer Edition and Sterling Connect:Direct have multiple methods of initiating transfers and executing programs after a successful file transfer. The following other methods are available, but we do not discuss these methods in this book:

- ▶ User exit programs
- ▶ A Software Development Kit (SDK)
- ▶ File monitoring applications
- ▶ Built-in job scheduler

- ▶ Scripts
- ▶ Ant jobs

Additional documentation: For more WebSphere MQ File Transfer Edition documentation see “Customizing WebSphere MQ File Transfer Edition with user exit routines” at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.doc/user_exits.htm

4.1.2 Business value

Sterling Connect:Direct is a point-to-point file-based integration solution that is designed for unattended operation. It provides assured delivery and high-volume, secure data exchange. It is designed to move files that contain any type of data across multiple platforms, disparate file systems, and disparate media, maintaining high performance levels and throughput. Sterling Connect:Direct eliminates the need for manual intervention in data delivery, improving productivity and the reliability of business processes.

WebSphere MQ File Transfer Edition exploits the proven reliability and connectivity of WebSphere MQ to transfer files across a wide range of platforms and networks. In addition to using existing WebSphere MQ networks, you can integrate WebSphere MQ File Transfer Edition with existing file transfer systems.

WebSphere MQ File Transfer Edition

WebSphere MQ File Transfer Edition provides an enterprise-ready, managed file transfer capability that is both robust and easy to use. WebSphere MQ File Transfer Edition exploits the proven reliability and connectivity of WebSphere MQ to transfer files across a wide range of platforms and networks. In addition to using existing WebSphere MQ networks, you can integrate WebSphere MQ File Transfer Edition with existing file transfer systems.

WebSphere MQ File Transfer Edition offers the following benefits:

- ▶ **Auditability**

WebSphere MQ File Transfer Edition provides full logging of transfers at both the source and destination systems. File transfer audit logs are stored in WebSphere MQ queues and optionally in a relational SQL database.
- ▶ **Ease of use**

Using WebSphere MQ File Transfer Edition, you can initiate file transfers using the GUI in WebSphere MQ Explorer, CLI, and scripts.
- ▶ **Simplicity**

WebSphere MQ File Transfer Edition has a low resource footprint and, apart from WebSphere MQ, has no other prerequisite software.
- ▶ **Security**

Access to files is controlled by file system permissions. File transfers can be protected using SSL encryption and authentication.
- ▶ **Automation**

File transfers can be set up to occur at specified times or dates or to repeat at specified intervals. File transfers can also be triggered by a range of system events, such as new files or updated files.

Sterling Connect:Direct

Sterling Connect:Direct is a solution for secure, point-to-point file transfers. It is optimized for high-volume, assured data delivery of files within and between enterprises and provides script-based automation, scheduling, and alert notifications for 24x7 unattended operations. Sterling Connect:Direct allows organizations to automate the data exchange between mission-critical applications, regardless of the platform. The event-based architecture enables high volumes and large files, with no product-defined limits on file sizes. Sterling Connect:Direct also supports various clustering technologies and IBM Sysplex on the mainframe. It provides built-in automation and checkpoint restart to ensure lights-out operations.

Automation and management

Sterling Connect:Direct includes the following automation and management features:

- ▶ Supports 24x7 unattended operations
- ▶ Schedules jobs on a one-time, recurring, or continuous basis
- ▶ Assigns and manages file transfer workload
- ▶ Provides event-driven alert notification
- ▶ Includes a process language that builds scripts to provide integration with back-end systems
- ▶ Supports API and SDK for programmatic access by other applications
- ▶ Supports checkpoint restart
- ▶ Includes automatic recovery from network interruptions
- ▶ Provides automated alert notifications for success or failure

Security and compliance

Sterling Connect:Direct includes the following security and compliance features:

- ▶ Standard Sterling Connect:Direct
 - Interfaces with operating system security for user authentication
 - Provides a complete audit trail of data movement through extensive statistics logs
- ▶ Sterling Connect:Direct Secure Plus
 - User authentication
 - X.509 certificates for authentication
 - Data encryption (SSL/TLS)
 - Certificate and Certificate Revocation List (CRL) checking
 - FIPS 140-2 and Common Criteria certification

Multiple platform support

Sterling Connect:Direct includes support for the following operating systems:

- ▶ z/OS and z/VSE
- ▶ OpenVMS
- ▶ i5/OS (OS/400)
- ▶ UNIX and Linux
- ▶ Windows
- ▶ HP NonStop
- ▶ Sterling Connect:Direct Select (Java version that can run on multiple platforms)

4.2 Scenario details

The scenarios that we discuss in this chapter demonstrate moving data between Sterling Connect:Direct and WebSphere MQ File Transfer Edition. We begin with an introductory example that shows a simple transfer from one Connect:Direct node to another Connect:Direct node. This introductory example is intended for readers who might be less familiar with Sterling Connect:Direct. As such, the first scenario demonstrates a simple peer-to-peer transfer using only two Connect:Direct nodes. The scenarios then build upon the introductory scenario by adding WebSphere MQ File Transfer Edition into the examples.

With WebSphere MQ File Transfer Edition added, the examples describe a file transfer initiated at a Connect:Direct node and provide an example file upload (push) to a remote WebSphere MQ File Transfer Edition agent. An example of a file download (pull) from WebSphere MQ File Transfer Edition that is initiated by Sterling Connect:Direct is also included.

Lastly, the examples describe a file transfer that is initiated by a WebSphere MQ File Transfer Edition agent. An example file upload (push) to a remote Connect:Direct node is shown. An example of a file download (pull) from Sterling Connect:Direct that is initiated by WebSphere MQ File Transfer Edition is also described.

4.2.1 Solution components

This section describes the components that are associated with each product in this solution (Figure 4-1). Certain components require a specific configuration for the solution to work. We discuss the configuration steps that are required as necessary.

For Sterling Connect:Direct, each data transfer involves a local and a remote Connect:Direct server (also referred to as *nodes*). The two servers work together to perform the work in a peer-to-peer relationship. The server initiating the connection is the *primary node* (PNODE) for the connection, and the server receiving the connection is the *secondary node* (SNODE). A Connect:Direct server can manage multiple concurrent connections with other Connect:Direct servers and can simultaneously act as both a PNODE and an SNODE.

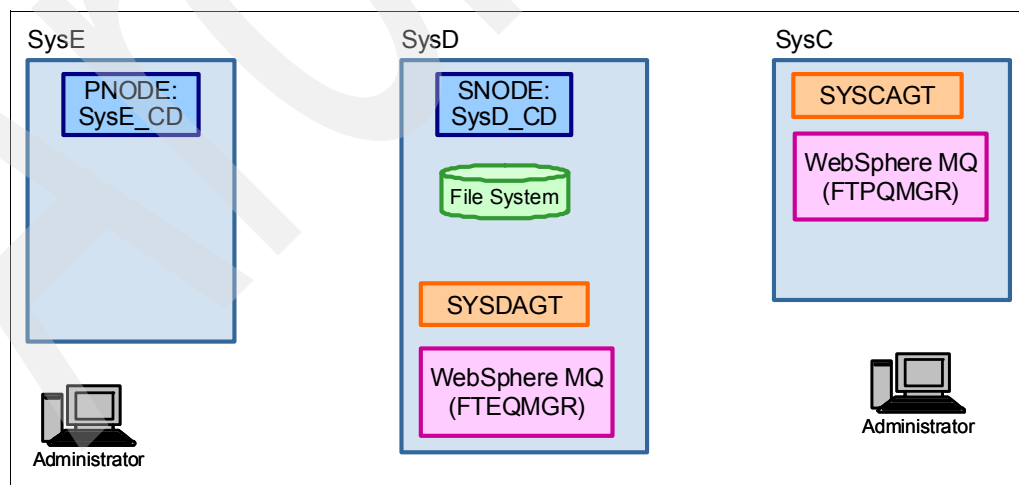


Figure 4-1 The WebSphere MQ File Transfer Edition and Sterling Connect:Direct internal systems

Sterling Connect:Direct

Sterling Connect:Direct moves files point to point (peer to peer) using the Sterling Connect:Direct protocol. A Sterling Connect:Direct client is used to communicate with a

Connect:Direct server regarding the work that will be performed. The Connect:Direct requester, a GUI, and the CLI are used to communicate with the Connect:Direct server.

Connect:Direct node SysE_CD

As illustrated in Figure 4-1 on page 55, the Connect:Direct node installed on SysE is named SysE_CD. For the scenarios in this chapter, SysE_CD represents an internal trading partner at one end point. The files transferred in the scenarios in this chapter start and end at this endpoint.

Connect:Direct node SysD_CD

The Connect:Direct node installed on SysD is named SysD_CD. For the scenarios in this chapter, SysD_CD represents an external trading partner at one end-point (Figure 4-1 on page 55). The files transferred in the scenarios in this chapter both start and end at this endpoint.

Connect:Direct Requester for Windows

Connect:Direct Requester for Windows is a GUI for Connect:Direct servers on Windows, UNIX, and Open VMS platforms. You can connect to Connect:Direct servers to perform file transfers, run remote programs or batch jobs, and create, submit, and monitor Connect:Direct processes. You can also perform Sterling Connect:Direct administrative functions, such as setting up and maintaining Sterling Connect:Direct users and network maps. In this chapter, the Connect:Direct Requester is used to configure and launch Connect:Direct processes on SysD_CD.

Sterling Connect:Direct CLI

The Sterling Connect:Direct CLI is a CLI to Connect:Direct servers on the UNIX platform. You can connect to Connect:Direct servers and perform file transfers, run remote programs or batch jobs, and create, submit, and monitor Connect:Direct processes. In this chapter, the Sterling Connect:Direct CLI is used to configure and launch Connect:Direct processes on SysE_CD.

WebSphere MQ Command and Agent Queue Manager (FTPQMGR)

WebSphere MQ Queue Manager (FTPQMGR) acts as the command and agent queue manager for the SYSCAGT WebSphere MQ File Transfer Edition agent.

WebSphere MQ Coordination Queue Manager (FTEQMGR)

WebSphere MQ Queue Manager (FTEQMGR) is the WebSphere MQ File Transfer Edition coordination queue manager. The coordination queue manager publishes status messages that are received from the agents, showing the state of transfers and the agent's status. FTEQMGR also acts as the command and agent queue manager for the SYSDAGT WebSphere MQ File Transfer Edition agent.

WebSphere MQ File Transfer Edition Server Agent (SYSDAGT)

WebSphere MQ File Transfer Edition Server Agent (SYSDAGT) is the WebSphere MQ File Transfer Edition agent that connects to the local queue manager FTEQMGR in bindings mode. This type of agent is referred to as a *server agent*. SYSDAGT reads files from and writes files to the local file system. This agent is the target agent for SYSCAGT in the scenarios in this chapter.

WebSphere MQ File Transfer Edition Server Agent (SYSCAGT)

WebSphere MQ File Transfer Edition Server Agent (SYSCAGT) is the WebSphere MQ File Transfer Edition agent that connects to the local queue manager FTPQMGR in bindings mode. This type of agent is referred to as a *server agent*. SYSCAGT reads files from and

writes files to the local file system. This agent is the target agent for SYSDAGT in the scenarios in this chapters.

WebSphere MQ Explorer and WebSphere MQ File Transfer Edition Explorer

WebSphere MQ Explorer is used to view and administer the WebSphere MQ queue managers and queue manager objects, such as queues, topics, and channels. WebSphere MQ Explorer is built on an Eclipse-integrated development environment. The Eclipse-based platform allows plug-ins to be added to the base platform.

The WebSphere MQ File Transfer Edition Explorer is a plug-in to WebSphere MQ Explorer. It is used to schedule file transfer requests and view the status of current requests. The tool includes a transfer log view that subscribes to the coordination queue manager to obtain audit information. The audit information is displayed in the transfer log view for every transfer that occurs in the given topology. Beginning in WebSphere MQ File Transfer Edition Version 7.0.3, WebSphere MQ File Transfer Edition Explorer also includes the ability to view the status of an agent.

4.2.2 The Sterling Connect:Direct to Sterling Connect:Direct scenario

Figure 4-2 portrays the simple transfer from one Connect:Direct node to another Connect:Direct node. A Connect:Direct node named SysE_CD is installed on the system named SysE. Additionally, a Connect:Direct node named SysD_CD is installed on SysD.

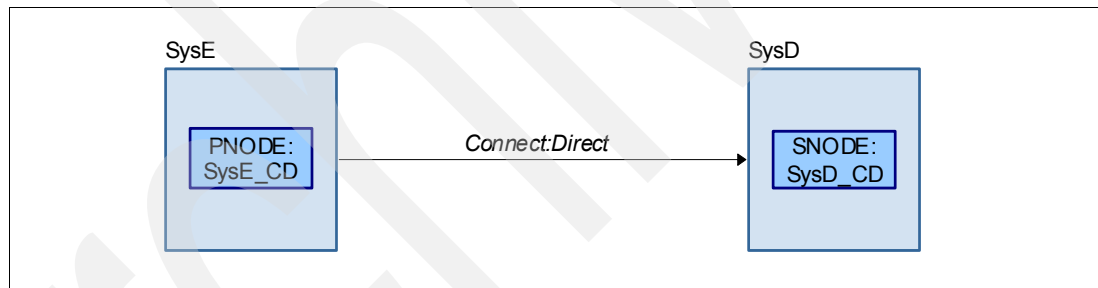


Figure 4-2 Connect:Direct node to Connect:Direct node

A pre-existing file on SysE flows from the SysE_CD node to the SysD_CD node. The file is written to disk on SysD. The sequence of events is as follows:

1. A plain-text Connect:Direct process file is created on SysE. A Connect:Direct process contains the simple command statements that describe the work for SysE_CD to perform. The name of the file on SysE is `sample.cd`.
2. On SysE_CD, a user logs in to the Sterling Connect:Direct CLI with a user ID that has privileges to log in to Sterling Connect:Direct.
3. Within the CLI, the user enters the command to instruct SysE_CD to read and parse the `sample.cd` Connect:Direct process file that was created in step 1.
4. SysE_CD establishes a network connection to SysD_CD.
5. The file is transferred over the network from SysE_CD to SysD_CD.
6. The Connect:Direct node SysD_CD writes the file to the SysD system.
7. SysE_CD terminates the network connection to SysD_CD.

4.2.3 Sterling Connect:Direct push to WebSphere MQ File Transfer Edition

In Figure 4-3, we build on the basic scenario shown in 4.2.2, “The Sterling Connect:Direct to Sterling Connect:Direct scenario” on page 57 by adding WebSphere MQ File Transfer Edition to the file transfer. Here, we portray a transfer from Sterling Connect:Direct to Sterling Connect:Direct to WebSphere MQ File Transfer Edition on SysD to WebSphere MQ File Transfer Edition on SysC. Sterling Connect:Direct initiates the transfer.

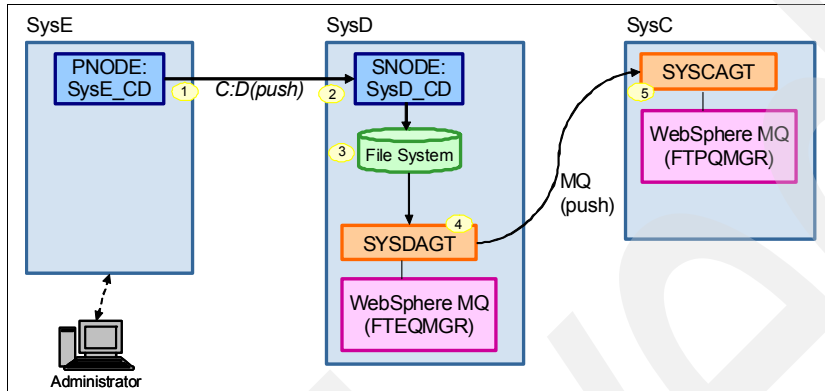


Figure 4-3 Sterling Connect:Direct to Sterling Connect:Direct to WebSphere MQ File Transfer Edition to WebSphere MQ File Transfer Edition

The Connect:Direct process language accepts multiple steps in a Connect:Direct process. For this example, we copy the file in the first step just as we did in 4.2.2, “The Sterling Connect:Direct to Sterling Connect:Direct scenario” on page 57. A second step is added to the Connect:Direct process to execute a program. The program executed is `fteCreateTransfer.cmd`. Running `fteCreateTransfer.cmd` on SysD launches the WebSphere MQ File Transfer Edition transmission. The WebSphere MQ File Transfer Edition transmission sends the file from the WebSphere MQ File Transfer Edition agent on SysD to the WebSphere MQ File Transfer Edition agent on SysC.

The sequence of events is:

1. The file is transferred using Sterling Connect:Direct from SysE_CD to SysD_CD in the first step of the Connect:Direct process.
2. The Sterling Connect:Direct SysD_CD node receives the file and writes it to the file system.
3. After the file is transferred to SysD, Sterling Connect:Direct runs the second step in the Connect:Direct process. This second step executes the `fteCreateTransfer.cmd` program.
4. SYSDAGT reads the file from the local file system and sends it to SYSCAGT.
5. SYSCAGT receives the file and saves it on the SysC system.

4.2.4 Sterling Connect:Direct pulling from WebSphere MQ File Transfer Edition

In Figure 4-4, we portray a file download (pull) that is initiated by Sterling Connect:Direct. A file is transferred from WebSphere MQ File Transfer Edition on SysC to WebSphere MQ File Transfer Edition on SysD, and then from Sterling Connect:Direct (SysD) to Sterling Connect:Direct (SysE).

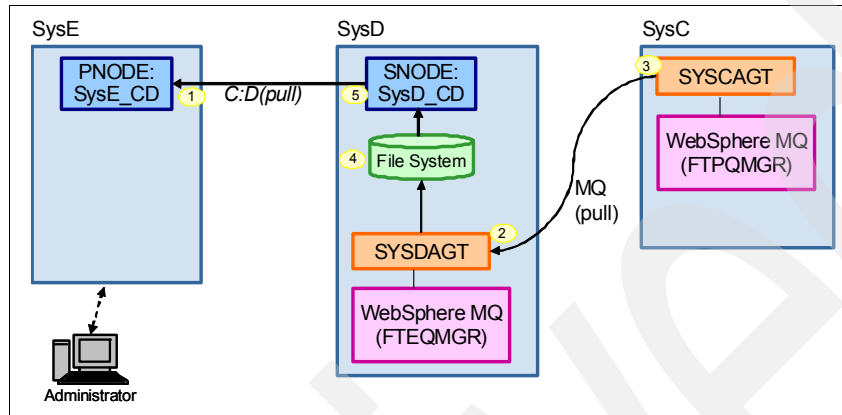


Figure 4-4 C:D pull from C:D pull from WebSphere MQ File Transfer Edition pull from WebSphere MQ File Transfer Edition

For this example Sterling Connect:Direct initiates the WebSphere MQ File Transfer Edition transfer in the first step of the Connect:Direct process. The first step of the process executes `fteCreateTransfer.cmd` on SysD. The first step completes and the second step runs the file transfer to pull the file from SysD_CD to SysE_CD.

The sequence of events is:

1. The Connect:Direct node SysE_CD contacts SysD_CD. In the first step of the Connect:Direct process, SysE_CD instructs SysD_CD to execute `fteCreateTransfer.cmd`.
2. SYSDAGT contacts SYSCAGT and requests that a file be sent back to it.
3. SYSCAGT sends the file back to SYSDAGT.
4. SYSDAGT writes the file to the SysD file system.
5. The Connect:Direct node SysE_CD instructs SysD_CD to send the file back to SysE_CD.

4.2.5 WebSphere MQ File Transfer Edition pushing to Sterling Connect:Direct

Figure 4-5 shows the flow for this scenario when files are transferred (push) from WebSphere MQ File Transfer Edition (SysC) to WebSphere MQ File Transfer Edition (SysD) to Sterling Connect:Direct (SysD) to Sterling Connect:Direct (SysE).

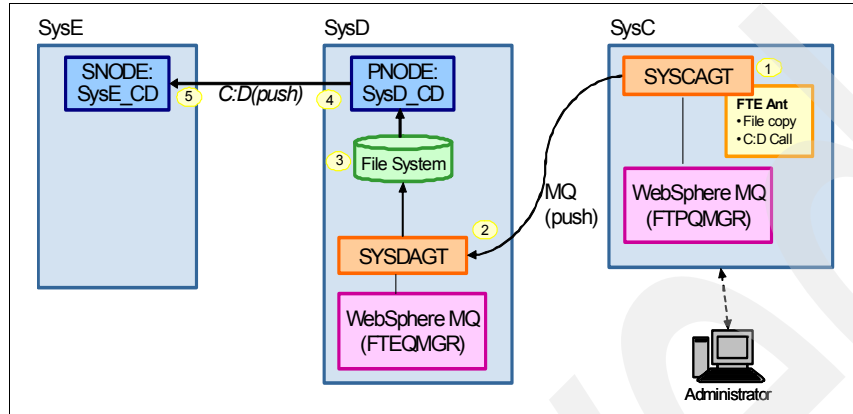


Figure 4-5 WebSphere MQ File Transfer Edition push files to WebSphere MQ File Transfer Edition to C:D to C:D

For this example, we show a file transfer flow that is initiated by WebSphere MQ File Transfer Edition. The WebSphere MQ File Transfer Edition uses an Ant script on SysC to run a Connect:Direct process on SysD as post-processing after the WebSphere MQ File Transfer Edition transfer occurs. To transfer the file between Sterling Connect:Direct, we use a Connect:Direct process, as we did in 4.2.2, “The Sterling Connect:Direct to Sterling Connect:Direct scenario” on page 57.

The sequence of events is:

1. The file transfer is initiated by an Ant script on SysC.
2. The file is transferred from SYSCAGT to SYSDAGT using WebSphere MQ File Transfer Edition.
3. SYSDAGT writes the file to the file system and launches a Connect:Direct process.
4. The file is transferred using Sterling Connect:Direct from SysD_CD to SysE_CD.
5. SysE_CD receives the file.

4.2.6 WebSphere MQ File Transfer Edition pulling from Sterling Connect:Direct

Figure 4-6 shows the flow for this scenario when files are transferred (pull) from Sterling Connect:Direct (SysE) from Sterling Connect:Direct (SysD) from WebSphere MQ File Transfer Edition (SysD) from WebSphere MQ File Transfer Edition (SysC).

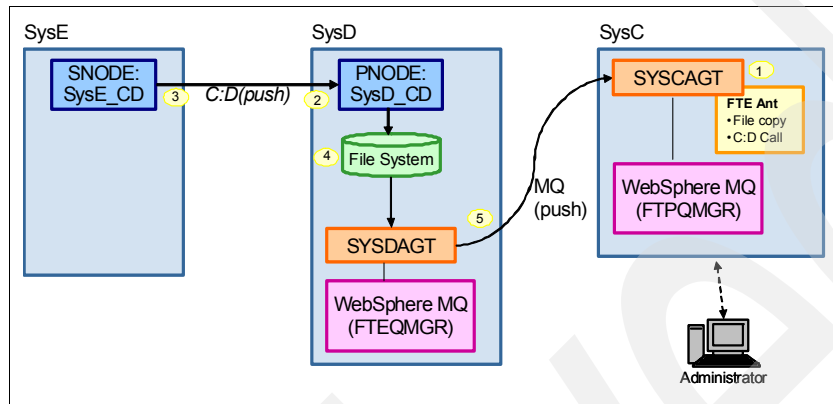


Figure 4-6 WebSphere MQ File Transfer Edition pull files from C:D from C:D from WebSphere MQ File Transfer Edition

For this example, we show a file transfer flow that is initiated by WebSphere MQ File Transfer Edition. The WebSphere MQ File Transfer Edition uses Ant script on SysC to run a Connect:Direct process on SysD as preprocessing before the WebSphere MQ File Transfer Edition transfer occurs. To transfer the file between Sterling Connect:Direct, we use a Connect:Direct process, as we did in 4.2.2, “The Sterling Connect:Direct to Sterling Connect:Direct scenario” on page 57.

The sequence of events is:

1. The file transfer is initiated by an Ant script on SysC.
2. SYSDAGT launches a Connect:Direct process to instruct SysE_CD to send the file to SysD_CD.
3. SysE_CD sends the file to SysD_CD using Sterling Connect:Direct.
4. SysD_CD receives the file to file system.
5. The file is transferred from SYSDAGT to SYSCAGT using WebSphere MQ File Transfer Edition.

4.2.7 Protocols

These scenarios use the Sterling Connect:Direct protocol to transfer data between SysE_CD and SysD_CD. The WebSphere MQ File Transfer Edition uses WebSphere MQ to move files from one WebSphere MQ File Transfer Edition agent to the next.

4.2.8 Security

To perform work in your enterprise, Sterling Connect:Direct relies on building blocks of information that define the local and remote nodes, users who can access those nodes, and the functions that users can perform.

Local node definition

During installation, you define a local node for Sterling Connect:Direct. The local node definition specifies information, such as the operating system, default user ID, TCP/IP address, and port number. After installation, you can change the local node's settings and define remote nodes. In addition to the default user ID that you specify for a local node, you can add other users who will access that node.

Local user authorities

After you define a user ID for each user who has access to the local node, you can restrict the ability of each user to perform certain tasks by defining user authorities for each user ID. For example, you can permit a user to submit a process, but not to monitor or delete processes.

Sterling Connect:Direct has two types of users:

- ▶ Administrators
- ▶ General users

Each type has a set of default privileges. You can use these user templates to assign user authorities and to restrict user privileges. Local user authorities provide one type of authentication in Sterling Connect:Direct. An alternative method of authentication is available using remote user proxies. For a listing of the default authorities for each type, see the product documentation for your Sterling Connect:Direct platform.

Remote user proxies

User proxy definitions (referred to as *secure point of entry* on the mainframe) contain remote user information for operations that are initiated from remote Connect:Direct nodes. These definitions identify a proxy relationship between a user ID at a remote Connect:Direct node and a local user ID. This mapping of remote and local user IDs enables users at remote Connect:Direct nodes to submit work to the local Connect:Direct node without explicitly defining user IDs and passwords in the processes, eliminating the need to share passwords with your trading partners. User proxies also define what each user ID can do on the local Connect:Direct node.

Configuration settings for the local node

Initialization parameters determine various Sterling Connect:Direct settings that control system operation. The initialization parameters file is created when you install Sterling Connect:Direct and can be updated as needed. Some of these settings can be overwritten in the netmap, user authorities, user proxies, and processes.

Remote node definitions

The network map (or *netmap*) is a file created during the Sterling Connect:Direct installation that identifies the remote nodes with which each local node can communicate and the communication information that is needed to establish a connection. You create a remote node entry in the network map for each remote node with which the local node communicates. Each network map entry contains information about the remote node, such as the remote node name, operating system, session characteristics for a protocol, and transfer and protocol information about the available communications paths and their attributes.

Netmap checking

In addition to defining the remote nodes that communicate with the Connect:Direct node, the network map can be used to perform a security function. Netmap checking verifies that inbound sessions are from a node defined in the network map. If the node is not in the network map, the connection fails.

WebSphere MQ File Transfer Edition security

For any file transfer request, the agent process requires a certain level of access to its local file systems. In addition, both the user identifier that is associated with the agent process and the user identifiers that are associated with users who perform file transfer operations must have the authority to use certain WebSphere MQ objects.

Commands are issued by users who might be in operational roles, in which they typically start file transfers. Alternatively, they might be in administrative roles, in which they can additionally control when agents are created, started, deleted, or cleaned (when messages from all agent system queues are removed). Messages that contain command requests are placed on an agent's SYSTEM.FTE.COMMAND queue when a user issues a command. The agent process retrieves messages that contain command requests from the SYSTEM.FTE.COMMAND queue. The agent process also uses the following system queues:

- ▶ SYSTEM.FTE.DATA.*agent_name*
- ▶ SYSTEM.FTE.EVENT.*agent_name*
- ▶ SYSTEM.FTE.REPLY.*agent_name*
- ▶ SYSTEM.FTE.STATE.*agent_name*

WebSphere MQ File Transfer Edition supports finer-grained checking of users authorities, which permits access to be granted (or denied) to specific product functions for each user. For example, you can choose which users have the authority to schedule transfer operations to happen at a future time. Because users who issue commands use the queues in different ways from the agent process, assign different WebSphere MQ authorities to the user identifiers or user groups that are associated with each. For more information, see *Using groups to manage authorities for resources specific to WebSphere File Transfer Edition* at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/group_resource_access.htm

The agent process has additional queues that can be used to grant users the authority to perform certain actions. The agent does not put or get messages on these queues. However, you must ensure that the queues are assigned the correct WebSphere MQ authorities, both for the user identifier that is used to run the agent process and for the user identifiers that are associated with users who are being authorized to perform certain actions. The authority queues are:

- ▶ SYSTEM.FTE.AUTHADM1.*agent_name*
- ▶ SYSTEM.FTE.AUTHAGT1.*agent_name*
- ▶ SYSTEM.FTE.AUTHMON1.*agent_name*
- ▶ SYSTEM.FTE.AUTHOPS1.*agent_name*
- ▶ SYSTEM.FTE.AUTHSCH1.*agent_name*
- ▶ SYSTEM.FTE.AUTHTRN1.*agent_name*

The agent process also publishes messages to the SYSTEM.FTE topic on the coordination queue manager using the SYSTEM.FTE queue. Depending on whether the agent process is in the role of the source agent or the destination agent, the agent process might require authority to read, write, update, and delete files.

You can create and modify authority records for WebSphere MQ objects using the WebSphere MQ Explorer. Right-click the object, and then click **Object Authorities** → **Manage Authority Records**. You can also create authority records using the `setmqaut` command.

Instead of granting authority to individual users for all of the various objects that might be involved, configure two security groups for the purposes of administering WebSphere MQ File Transfer Edition access control:

- ▶ FTEUSER
- ▶ FTEAGENT

Additionally, WebSphere MQ File Transfer Edition has security features to protect the files and file systems from unauthorized access. These features allow organizations to control who can read and write files that are transferred and how to protect the integrity of files. You can achieve this security using the following methods:

- ▶ Manage authorities to access file systems.

The user ID that is running the agent process must have access to the local file system to read or write files during file transfer. This access is controlled through the local operating system.

- ▶ Use sandboxes.

The access of an agent to the file system can be restricted by defining the `sandboxRoot` property in an agent's properties file. This property restricts the agent's access to a certain directory or a certain area of the file system, the *sandbox*. For more information about sandboxing, see:

<http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/sandboxes.htm>.

- ▶ Use the agent's `commandPath` property.

The `commandPath` property in an agent's property file restricts the locations from which an agent can run commands. By default, the `commandPath` is empty, so an agent cannot call any commands. Take extreme care when setting this property, because any command in one of the specified `commandPath` settings can be called from a remote client system that is able to send commands to the agent. For more information about this property, see:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/command_path.htm

- ▶ Configure SSL encryption for WebSphere MQ File Transfer Edition.

The file data when transferred between WebSphere MQ File Transfer Edition agents can be protected by establishing Secure Sockets Layer (SSL) on the WebSphere MQ channel connections.

- ▶ Provide authority to publish log and status messages.

Agents issue various log, progress, and status messages to the coordination queue manager for publication. The publication of these messages can be secured using WebSphere MQ security.

- ▶ Use the MD5 checksum.

This setting is the default setting on file transfers in WebSphere MQ File Transfer Edition. This setting is set to keep files from being manipulated after the transfer has been initiated or while the transfer is in progress.

- ▶ Authentication.

Client FTE agents must be authenticated. The authentication can be performed by WebSphere MQ using SSL or exits.

4.3 Configuring the solution components

This section describes the software configuration that is required for the scenarios that we describe in this chapter. Sterling Connect:Direct must be configured to communicate with a remote node with which it has not previously communicated. Additionally, configuration changes for WebSphere MQ File Transfer Edition are made to enable WebSphere MQ File Transfer Edition and Sterling Connect:Direct to pass files to each other. This section documents these configuration changes.

4.3.1 Software prerequisites

This scenario uses the following software:

- ▶ WebSphere MQ V7.0.1
- ▶ WebSphere MQ File Transfer Edition V7.0.3
- ▶ Sterling Connect:Direct for Microsoft Windows V4.5.0.1
- ▶ Sterling Connect:Direct for Linux V4.0

4.3.2 Configuration prerequisites

The following software is required for the scenario:

- ▶ Sterling Connect:Direct for Linux is installed and operating on SysE.
- ▶ Sterling Connect:Direct for Microsoft Windows is installed and operating on SysD.
- ▶ WebSphere MQ File Transfer Edition is installed and operating on SysD.
- ▶ WebSphere MQ File Transfer Edition is installed and operating on SysC.
- ▶ An operating system user ID named cdadmin is created on SysE and SysD.

The cdadmin user ID is given the default basic user rights on each system.

You can use another operating system user ID in place of the cdadmin user ID. In that case, replace cdadmin with the user ID of your choice in the examples that we show in this chapter.

These scenarios use the following preconfigured components for WebSphere MQ File Transfer Edition:

- ▶ WebSphere MQ queue manager FTPQMGR, FTEQMGR
- ▶ WebSphere MQ File Transfer Edition agent SYSCAGT, SYSDAGT

“Configuring WebSphere MQ File Transfer Edition” on page 356 provides instructions for creating these components.

Security prerequisites: Review your local security policy and practices to determine what is appropriate for your production environment. While the scenarios in this book do not implement security, it is important that you take security into consideration when implementing these scenarios in your own environment.

4.3.3 Configuring Sterling Connect:Direct on SysD

The following sections describe how to configure the Connect:Direct nodes so that they are aware of each other. To simplify the examples, both nodes use the same operating system user ID. Both nodes also use the Sterling Connect:Direct preset upload and download directories for transferred files. Preset upload and download directories enable a process to be submitted without using full path and file names. Given only a file name, both Connect:Direct nodes will push or pull any file from the preset directory.

We describe the procedure to configure preset upload and download directories in this section.

Before the Connect:Direct nodes are configured, add a new user named `cdadmin` on both `SysD_CD` and `SysE_CD`. The `cdadmin` user ID on both systems only needs basic user rights. The Sterling Connect:Direct examples in this chapter use the `cdadmin` user ID.

To configure the Sterling Connect:Direct for Microsoft Windows node `SysD_CD`:

1. Select **Start** → **All Programs** → **Sterling Commerce Sterling Connect:Direct vX.X** → **CD Requester**.
2. Expand the entry for `SysD_CD` and double-click **Netmap** (Figure 4-7).

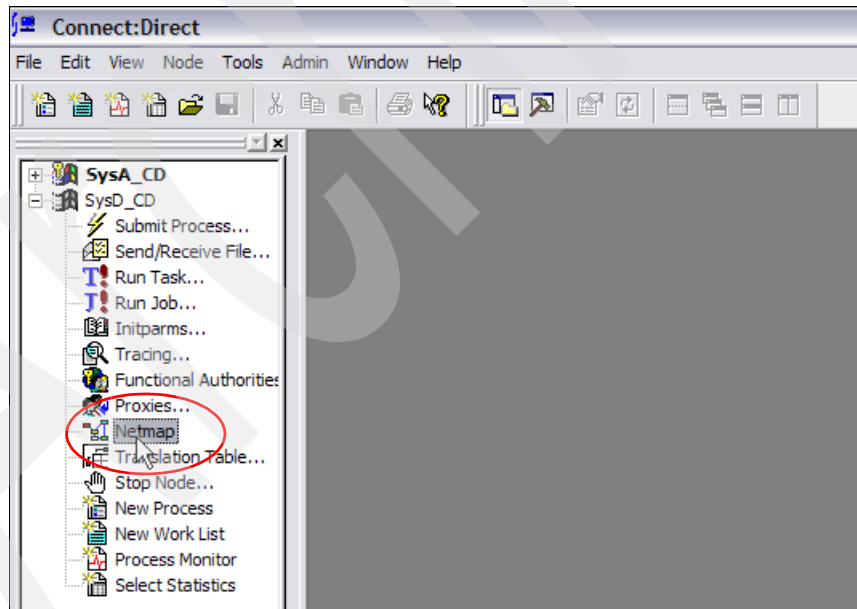


Figure 4-7 Launching the Connect:Direct requester netmap configuration

3. From the menu, select **Netmap** → **Insert** (Figure 4-8).

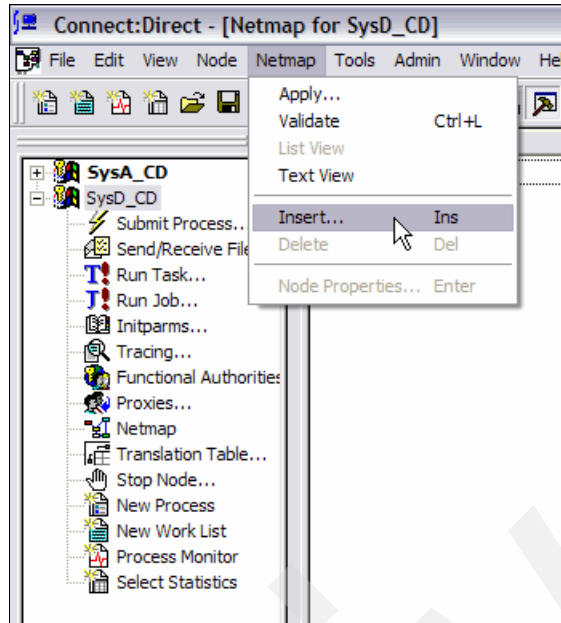


Figure 4-8 Inserting a new netmap entry

4. Under the Main tab, complete the values as shown in Figure 4-9.

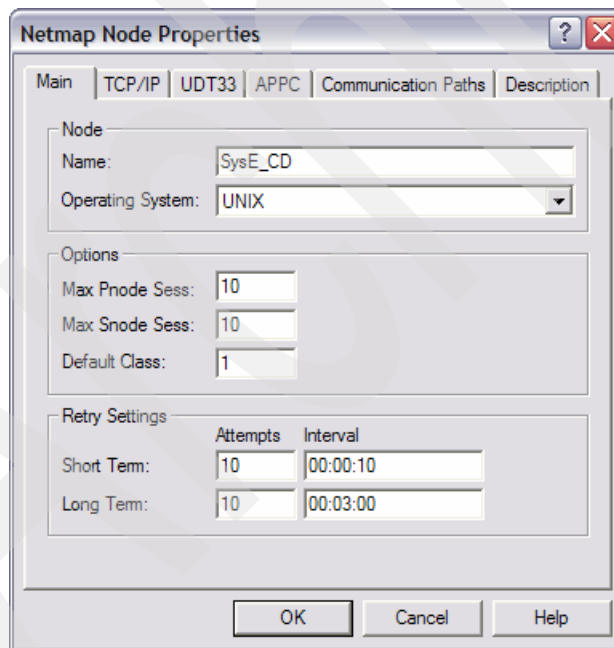


Figure 4-9 netmap Main tab

- Under the TCP/IP tab, complete the values as shown in Figure 4-10.

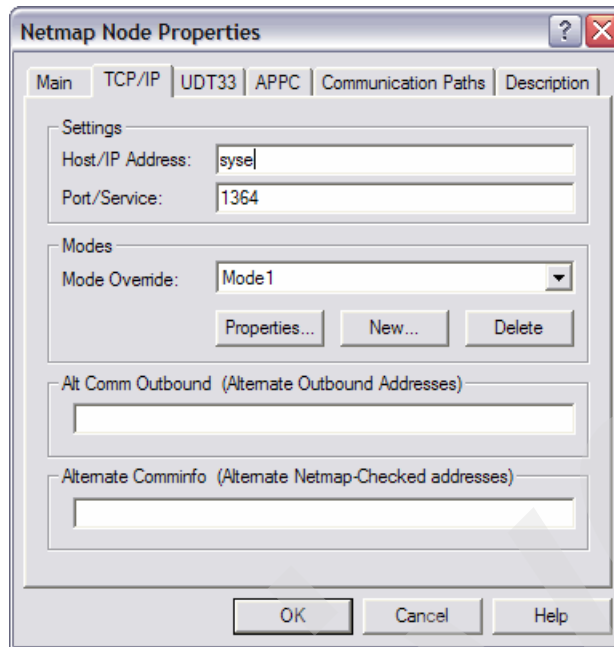


Figure 4-10 Netmap TCP/IP tab

- Under the Communications Paths tab, select and highlight **TCPCommPath**. Then click the right arrow to add TCPCommPath to the Selected Paths box (Figure 4-11).

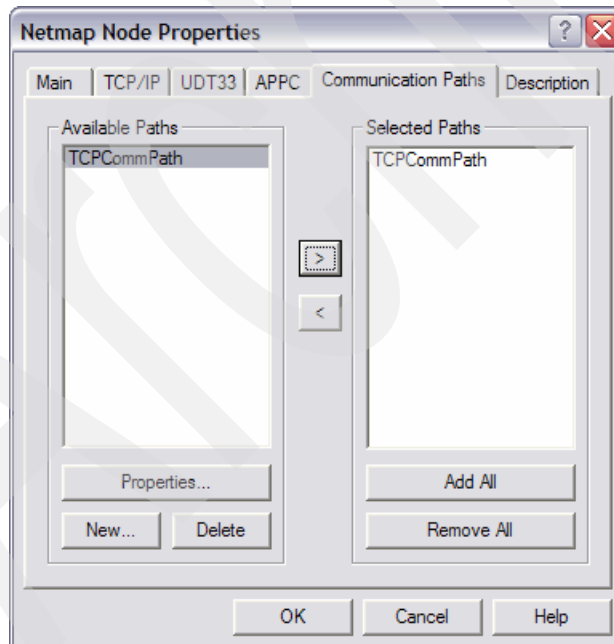


Figure 4-11 Netmap Communication Paths tab

- Click **OK** to stage the new entry.

8. Back on the main Netmap window, select **Netmap** → **Apply** (Figure 4-12).

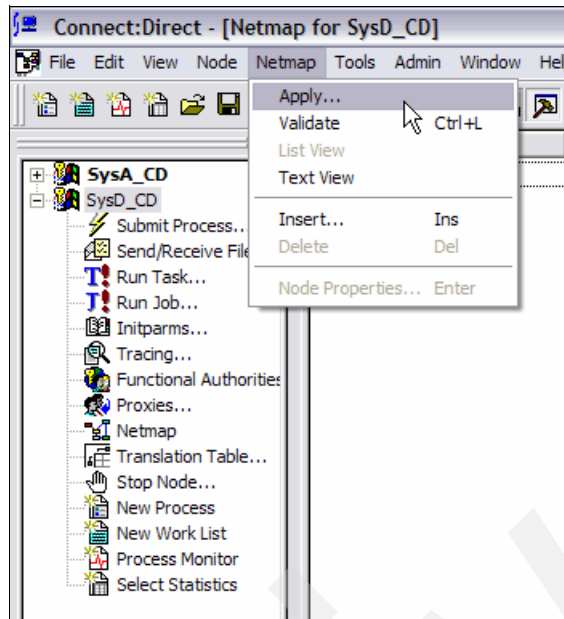


Figure 4-12 Applying a new netmap entry

9. If prompted to select a node to apply the changes to, select **SysD_CD** (Figure 4-13).

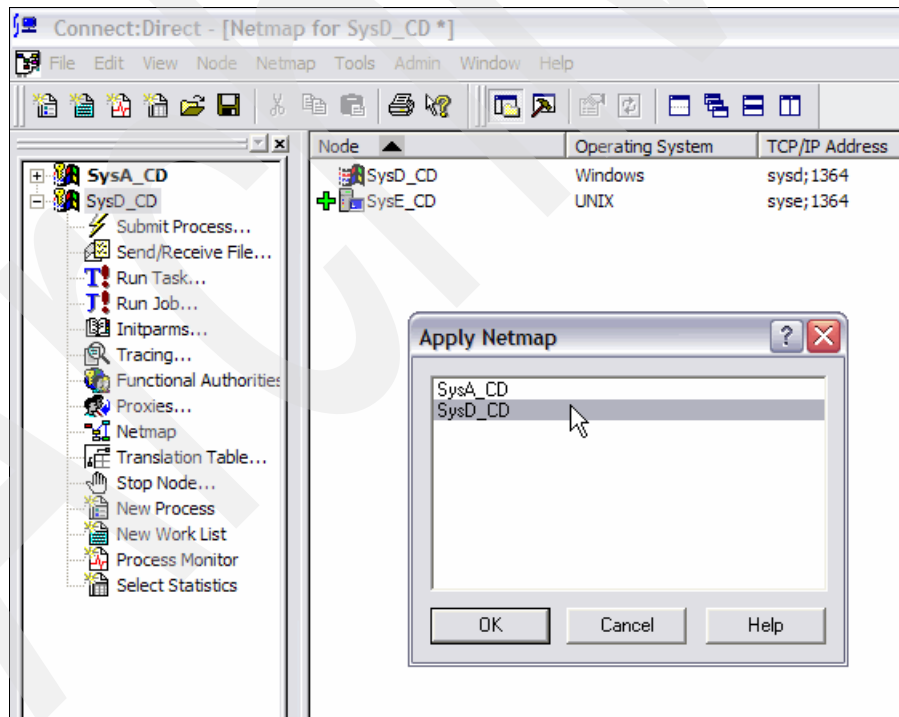


Figure 4-13 Applying a new netmap entry to multiple nodes

10. Create two directories on SysD_CD where Sterling Connect:Direct for Microsoft Windows is installed. Ensure that the cdadmin user ID has operating system permissions to read and write to the following directories:

- C:\CDWindows_files\upload
- C:\CDWindows_files\download

11. Expand the entry for **SysD_CD** and double-click **Functional Authorities** (Figure 4-14).

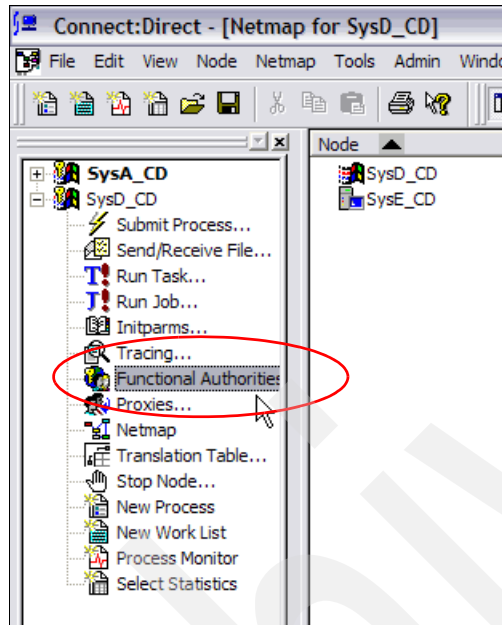


Figure 4-14 Launching the Connect:Direct requester Functional Authorities configurative

12. In the Functional Authorities dialog box, click **New Admin** (Figure 4-15).

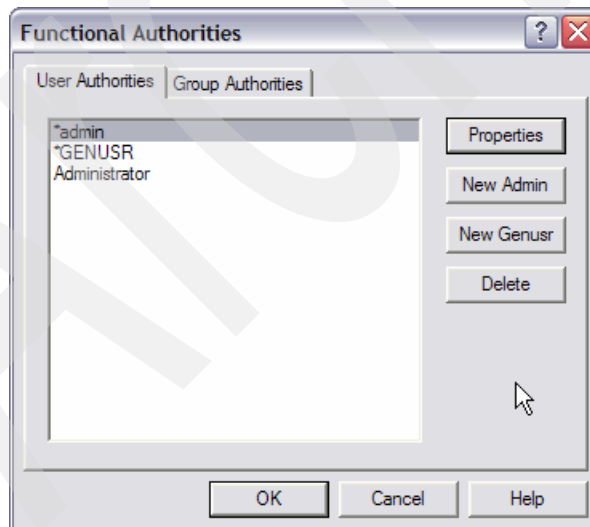


Figure 4-15 Functional Authorities dialog box

13. On the Main tab, enter `cdadmin` in the Name field (Figure 4-16).

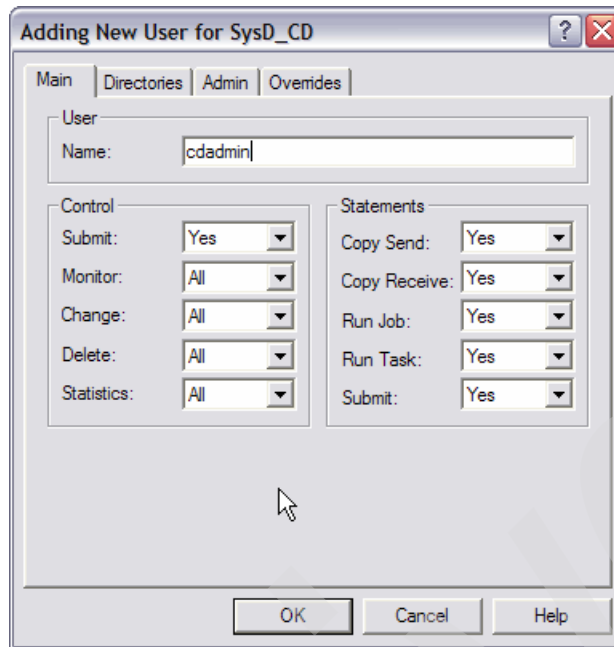


Figure 4-16 Functional Authorities new user Main tab

14. On the Directories tab, enter the upload and download directories that were created on SysD_CD in step 10 on page 70 (Figure 4-17). Click **OK** to save your changes.

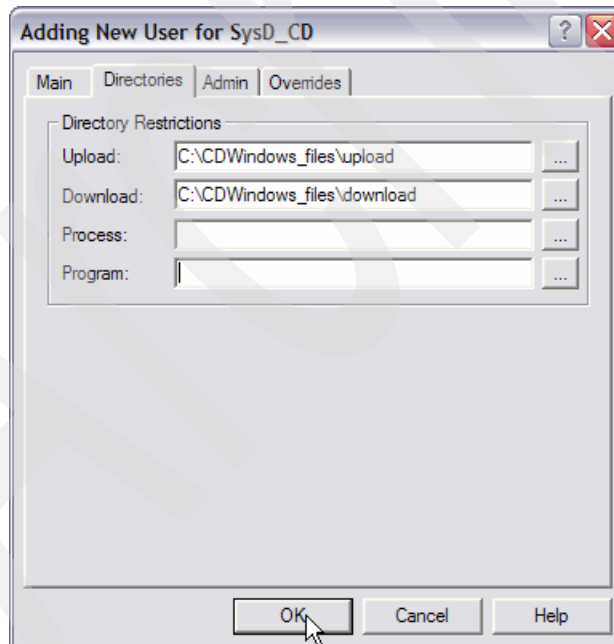


Figure 4-17 Functional authorities new user Directories tab

15. Expand the entry for **SysD_CD**, and then double-click **Proxies** (Figure 4-18).

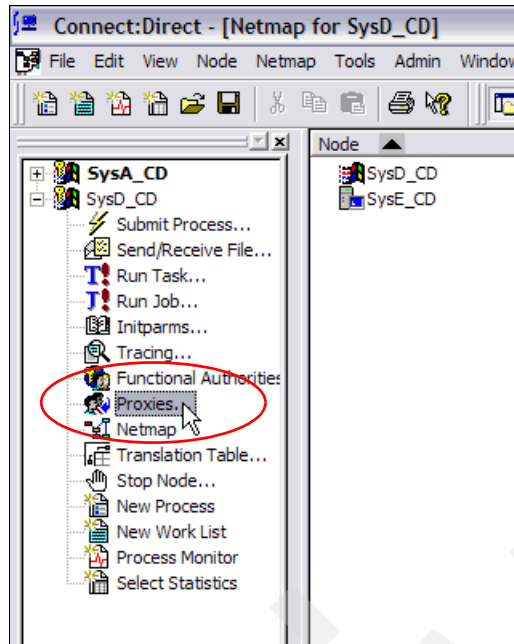


Figure 4-18 Launching the Connect:Direct requester Proxies configurative

16. Click **Insert** (Figure 4-19).

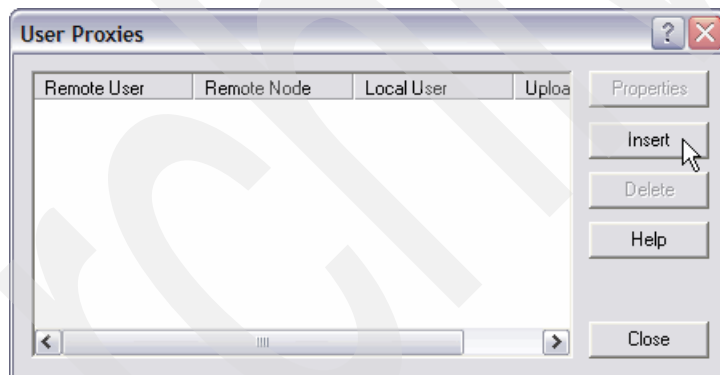


Figure 4-19 User Proxies dialog box

17. Under the Main tab, enter the text as shown in Figure 4-20.

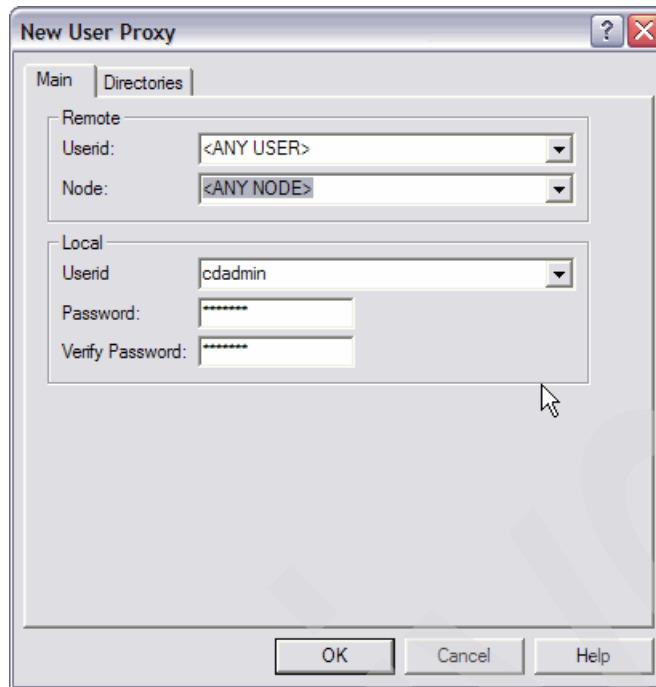


Figure 4-20 New User Proxy Main tab

18. On the Directories tab (Figure 4-21), change the copy send and copy receive permissions to **Yes**. Enter the upload and download directories that were created on SysD_CD in step 10 on page 70. Select **OK** to save.

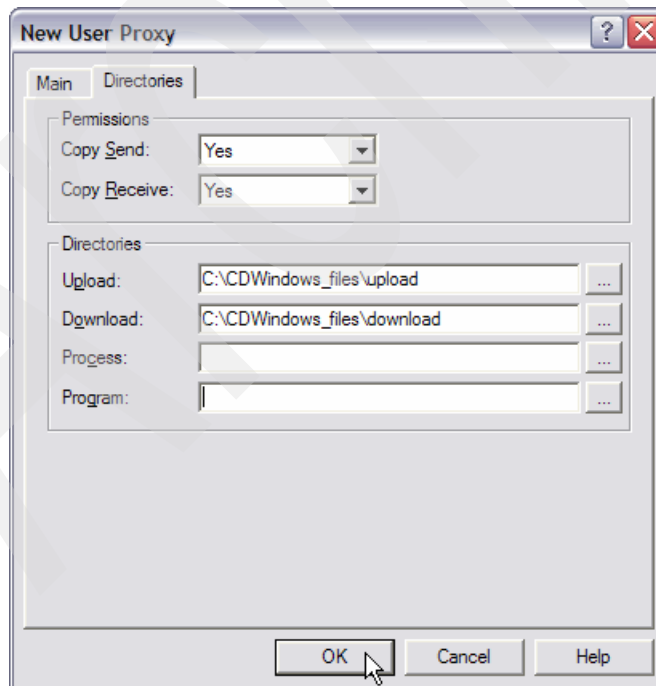


Figure 4-21 New User Proxy Directories tab

19. To enable the cdadmin user to launch a WebSphere MQ File Transfer Edition command, put the cdadmin user in the mqm group (Figure 4-22).

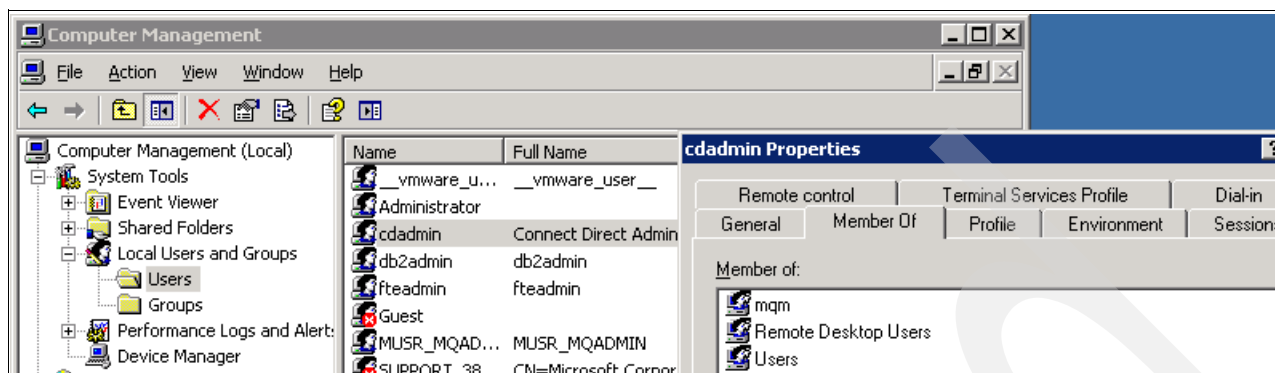


Figure 4-22 Putting the cdadmin user in the mqm group

4.3.4 Configuring Sterling Connect:Direct on SysE

The following sections describe how to configure the Connect:Direct nodes so that they are aware of each other. To simplify the examples, both nodes use the same operating system user ID. Both nodes also use the Sterling Connect:Direct preset upload and download directories for transferred files. Preset upload and download directories enable a process to be submitted without using the full path and file names. Given only a file name, both Connect:Direct nodes will push or pull any file from the preset directory. We describe the procedure to configure preset upload and download directories here.

Before the Connect:Direct nodes are configured, add a new user named cdadmin on both SysD_CD and SysE_CD. The cdadmin user ID on both systems needs only basic user rights. All the examples in this section use the cdadmin user ID.

To configure Sterling Connect:Direct SysE:

1. Append the netmap entry shown in Example 4-1 to the `/opt/cdunix/ndm/cfg/SysE_CD/netmap.cfg` file.

Example 4-1 Netmap entry on SysE_CD that points to SysD_CD

```
SysD_CD:\
:conn.retry.stwait=00.00.30:\
:conn.retry.stattempts=3:\
:conn.retry.ltwait=00.10.00:\
:conn.retry.ltattempts=6:\
:tcp.max.time.to.wait=180:\
:runstep.max.time.to.wait=0:\
:contact.name=\
:contact.phone=\
:descrip=\
:sess.total=255:\
:sess.pnode.max=255:\
:sess.snode.max=255:\
:sess.default=1:\
:comm.info=sysd;1364:\
:comm.transport=tcp:\
:comm.bufsize=65536:\
:pacing.send.delay=0:\
:pacing.send.count=0:
```

2. Append the userfile entry shown in Example 4-2 to the /opt/cdunix/ndm/cfg/SysE_CD/userfile.cfg file. Notice an upload and download directory are applied to the cdadmin entry.

Example 4-2 User file entry on SysE_CD for inbound connections by the cdadmin user ID

```
*@*:\  
:local.id=cdadmin:\  
:pstmt.upload=y:\  
:pstmt.upload_dir=:\  
:pstmt.download=y:\  
:pstmt.download_dir=:\  
:pstmt.run_dir=:\  
:pstmt.submit_dir=:\  
:descrip=:  
  
cdadmin:\  
:admin.auth=y:\  
:pstmt.copy.ulimit=y:\  
:pstmt.upload=y:\  
:pstmt.upload_dir=/opt/cdunix/upload:\  
:pstmt.download=y:\  
:pstmt.download_dir=/opt/cdunix/download:\  
:pstmt.run_dir=:\  
:pstmt.submit_dir=:\  
:name=:\  
:phone=:\  
:descrip=:
```

4.3.5 Configuring WebSphere MQ File Transfer Edition

This section details the Ant scripts that enable the following scenarios:

- ▶ 4.2.5, “WebSphere MQ File Transfer Edition pushing to Sterling Connect:Direct” on page 60
- ▶ 4.2.6, “WebSphere MQ File Transfer Edition pulling from Sterling Connect:Direct” on page 61

Creating Ant scripts

In the scenarios, a file transfer is requested at SysC, and a file is transferred automatically between SysC and SysE through SysD. Specifically, we want to initiate a WebSphere MQ File Transfer Edition file transfer on SysC, and before or after the transfer, we want to launch a Connect:Direct process to pull or push files without operating. In addition, the processes need to run on a remote system (SysD), which is not the system (SysC) where the Ant script is initiated. To meet these requirements, we make use of Ant scripts to make the Connect:Direct process be a pre-processing or post-processing task.

The Ant script created uses the following Ant tasks, which are provided by WebSphere MQ File Transfer Edition:

► call

This task allows you to send a call request to an agent, including a remote agent. The agent processes this request by running a script or program and returning the outcome. The script needs to be accessible to the agent that processes it.

► filecopy

This task copies files between WebSphere MQ File Transfer Edition agents. The file is not deleted from the source agent.

To demonstrate the scenarios that are described in 4.2.5, “WebSphere MQ File Transfer Edition pushing to Sterling Connect:Direct” on page 60, and 4.2.6, “WebSphere MQ File Transfer Edition pulling from Sterling Connect:Direct” on page 61, we created the following Ant scripts:

► Push_to_CD.xml

This script is used in the scenario in 4.2.5, “WebSphere MQ File Transfer Edition pushing to Sterling Connect:Direct” on page 60.

This script performs the following steps:

- Transfers a file from SYSCAGT to SYSDAGT using the filecopy task.
- Calls the InvokeCD.xml file on SysD as post-process using the call task. Uses the InvokeCD.xml file to run a Connect:Direct process to push a file from SysD_CD to SysE_CD.

You can find a description of this script in “Push_to_CD.xml script” on page 402.

► Pull_from_CD.xml

This script is used in the scenario that is described in 4.2.6, “WebSphere MQ File Transfer Edition pulling from Sterling Connect:Direct” on page 61.

This script performs the following steps:

- Calls the InvokeCD.xml file on SysD as pre-processing using call task. The InvokeCD.xml file runs the Connect:Direct process to pull the file from SysE_CD to SysD_CD.
- Transfer the file from SYSDAGT to SYSCAGT using the filecopy task.

You can find a description of this script in “Pull_from_CD.xml script” on page 404.

► InvokeCD.xml

This script is called by both the Push_to_CD.xml and Pull_from_CD.xml scripts and runs a Connect:Direct process, which runs a Connect:Direct process to pull the file from SysE_CD to SysD_CD, creating a temporary file to differentiate files for each job.

You can find a description of this script in “InvokeCD.xml script” on page 406.

The InvokeCD.xml file uses template process files for each scenario, which are used to launch the Connect:Direct process regardless of the source and destination file names. We created the Template_process_for_push.cdp for Push_to_CD.xml and Template_process_for_pull.cdp for Pull_from_CD.xml template process files.

Disclaimer: These Ant scripts are written to demonstrate file transfer between Sterling Connect:Direct and WebSphere MQ File Transfer Edition. We tested the scripts in basic scenarios, but the code is supplied as is and contains no error handling. We suggest that you analyze the solution thoroughly to ensure that it meets the requirements of your setup and to ensure that the solution is fully tested before you use it in any production environment.

Create and locate the scripts according to the information listed in Table 4-1.

Table 4-1 Scripts locations

Script name	System	Directory
Push_to_CD.xml	SysC	Locate it anywhere that you want.
Template_process_for_push.cdp	SysD	The directory that is set in the Push_to_CD.xml file as the template directory for the Connect:Direct process. See the cdTemplate property in Example 4-3 on page 78. We set this location to the following directory: C:\CDWindows_files\processes
Pull_from_CD.xml	SysC	Locate it anywhere that you want.
Template_process_for_pull.cdp	SysD	The directory that is set in the Pull_to_CD.xml file as the template directory for the Connect:Direct process. See the cdTemplate property in Example 4-5 on page 80. We set this location to the following directory: C:\CDWindows_files\processes
InvokeCD.xml	SysD	The directory that is set by commandPath in the agent.properties file. See Example 4-5 on page 80. We set the C:\FTE_CD directory as commandPath.

The Push_to_CD.xml script consists of the following procedures:

- ▶ **init**
 - Set the global properties that are used in the copy and post procedure using the property task, including the following settings:
 - The transfer file name
 - Temporary directory to locate the file, for Sterling Connect:Direct to send
 - Template C:D process file name
 - WebSphere MQ File Transfer Edition file transfer configuration
 - Configuration for InvokeCD.xml
 - A unique ID for the file transfer job using the uuid task
- ▶ **copy**
 - Transfer the file using the filecopy task.
 - The information of the WebSphere MQ File Transfer Edition agents is set by the attributes of the filecopy task.

- The metadata that is related to the file transfer is set by the metadata task.
- The setting for the transferring file, such as filename, is set by the filespec task.
- ▶ **post**
 - Request to SYSDAGT to run the Connect:Direct process, which is wrapped by the InvokeCD.xml script.
 - The information about the WebSphere MQ File Transfer Edition agents that launches the InvokeCD.xml is set by the attributes of the call task.
 - The command task sets the InvokeCD.xml command information, such as the file path for the command.
 - The nested property task sets the configurations that are used by InvokeCD.xml.
 - The metadata related to the calling task is set by the metadata task.

Example 4-3 shows the code for the Template_process_for_push.cdp script.

Example 4-3 Template_process_for_push.cdp

```

SUBMIT
PUSHTEST PROCESS
  SNODE=SysE_CD

STEP01 COPY
  from( file = "#{srcfile}"
        sysopts="datatype(text) xlate(no) strip.blanks(no)"
        pnode
        )

  ckpt = 2M
  compress extended

  to( file = "#{dstfile}"
      snode
      sysopts=":datatype=text:xlate=no:strip.blanks=no:"
      disp = rpl
      )

PEND;
```

The Template_process_for_push.cdp script is a Connect:Direct process file to launch a Connect:Direct process using the InvokeCD.xml file. This process file makes the PNODE SysD_CD push a file to SNODE SysE_CD. To send a file regardless of source and destination file names, the source file name is set as #{srcfile} and the destination file name is set as #{dstfile}. These file names are replaced by the InvokeCD.xml file, depending on the actual file name. For more details about the Connect:Direct process file, see 4.4.1, “Sterling Connect:Direct push to Sterling Connect:Direct” on page 81.

The Pull_from_CD.xml file consists of the following procedures:

- ▶ **init**
 - Sets the global properties that are used in the pre and copy procedure using the property task, including the following settings:
 - The transfer file name
 - Temporary directory to locate the file for Sterling Connect:Direct to send forward the file
 - Template C:D process file name
 - WebSphere MQ File Transfer Edition file transfer configuration
 - Configuration for the InvokeCD.xml file
 - A unique ID for the file transfer job using the uuid task
- ▶ **pre**
 - Request to SYSDAGT to run Connect:Direct process, which is wrapped by the InvokeCD.xml script.
 - The information about the WebSphere MQ File Transfer Edition agents that launch the InvokeCD.xml file is *settd* by the attributes of the call task.
 - The command task sets the InvokeCD.xml command information, such as the file path for the command and the nested property task that is set for the configurations that are used by the InvokeCD.xml file.
 - The metadata that is related to the calling task is set by the metadata task.
- ▶ **copy**
 - Request to transfer the file using the filecopy task.
 - The information about the WebSphere MQ File Transfer Edition agents is *settd* by the attributes of the filecopy task.
 - The metadata that is related to the file transfer is set by the metadata task.
 - The setting for the transferring file, such as filename, is set by the filespec task.

Example 4-4 shows code from the Template_process_for_pull.cdp script.

Example 4-4 Template_process_for_pull.cdp

```
SUBMIT
PULLTEST PROCESS
  SNODE=SysE_CD

STEP01 COPY
  from( file = "#{srcfile}"
        sysopts=":datatype=text:xlate=no:strip.blanks=no:"
        snode
      )

  ckpt = 2M
  compress extended

  to(file = "#{dstfile}"
     pnode
     sysopts="datatype(text) xlate(no) strip.blanks(no)"
```

```
        disp = (rpl)
    )
```

PEND;

The `Template_process_for_pull.cdp` script is a Connect:Direct process file to launch a Connect:Direct process using the `InvokeCD.xml` file. This process file makes the PNODE `SysD_CD` pull a file from SNODE `SysE_CD`. To send a file regardless of source and destination file names, the source file name is set as `#{srcfile}` and the destination file name is set as `#{dstfile}`. These file names are replaced by the `InvokeCD.xml` file, depending on the actual file name. For more details about the Connect:Direct process file, see Example 4-5.

The `InvokeCD.xml` file is launch by SYSDAGT as post-processing before WebSphere MQ File Transfer Edition file transfer or pre-processing after transfer. The `InvokeCD.xml` file consists of the following procedures:

- ▶ Copying the template Connect:Direct process file to the `${UUID}.cdp` file
 - `${UUID}` is a unique ID for the file transfer job.
 - `${UUID}.cdp` is used temporarily to launch a Connect:Direct process.
- ▶ Replacing the source file name and destination file name to match the actual file names
- ▶ Launching the Connect:Direct process
- ▶ Deleting the `${UUID}.cdp` file

Configuring WebSphere MQ File Transfer Agent

We assume that the queue manager, SYSDAGT, and SYSCAGT are already implemented.

For the Ant scripts to work properly, make the following configuration changes:

- ▶ Set the SYSDAGT `commandPath` property on SysD. Ensure that the `commandPath` property value in the `C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDAGT\agent.properties` file includes the location of the Ant script, `InvokeCD.xml`, to call. Any path information that is specified by the `command` nested element must be relative to the locations that are specified by the `commandPath` property. By default, `commandPath` is empty so that the agent cannot call any commands.

We configured the `agent.properties` file as shown in Example 4-5. The separator character, a backslash (`\`), must be escaped and appear as a double backslash (`\\`).

Example 4-5 The agent.properties file of SYSDAGT

```
#
#Mon Nov 15 14:04:58 EST 2010
agentQMgr=FTEQMGR
agentDesc=
agentName=SYSDAGT
commandPath=C:\\FTE_CD
```

- ▶ Restart SYSDAGT as shown in Example 4-6 using the **fteStopAgent** and **fteStartAgent** commands.

Example 4-6 Restart SYSDAGT

```
C:\Documents and Settings\fteadmin>fteStopAgent SYSDAGT
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED
BFGCL0034I: Stop request issued to agent 'SYSDAGT'.
BFGCL0198I: The agent has processed the stop request and will end once all
current transfers have completed.
```

```
C:\Documents and Settings\fteadmin>fteStartAgent SYSDAGT
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED
BFGCL0030I: The request to start agent 'SYSDAGT' on this machine has been
submitted.
BFGCL0031I: Agent log files located at: C:\Documents and Settings\All
Users\Application Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDAGT
```

4.4 Testing the flows

The following sections demonstrate how to transfer files between Sterling Connect:Direct for Linux and Sterling Connect:Direct for Microsoft Windows.

4.4.1 Sterling Connect:Direct push to Sterling Connect:Direct

Create and add the Sterling Connect:Direct plain-text process shown in Example 4-7 to the `/opt/cdunix/sample_push.cd` file on SysE_CD.

Example 4-7 Process to push a file to a remote Sterling Connect:Direct

```
/*
 * This sample process copies a text file "From_syse.txt"
 * from pnode SysE_CD to snode SysD_CD.
 */

ToSysD process snode=SysD_CD

    step01 copy
        from( file = From_syse.txt
              sysopts=":datatype=text:xlate=no:"
              pnode
            )

        ckpt = 2M
        compress extended

        to( file = From_syse.txt
           sysopts="datatype(text) xlate(no)"
           snode
           disp = (rpl)
         )

pend;
```

Sterling Connect:Direct identifies the node that initiates a process submission as the PNODE. The remote node is referred to as the SNODE. A PNODE is always the node that started or initiated the process. Sterling Connect:Direct can send or push files and can download or pull files from another Sterling Connect:Direct. Regardless of whether the file is pushed or pulled, the node that initiated the session is referred to as the PNODE.

The statements in the process

The following process statements are shown in Example 4-7 on page 81:

- ▶ C-style comments can be added to a Connect:Direct process using `/* comment */` notation.
- ▶ `SysE_CD` is the PNODE because it initiated the process. The Connect:Direct process specifies the remote node to connect to using the `snode=SysD_CD` statement.
- ▶ A Connect:Direct process can have multiple steps. Example 4-7 on page 81 only has one step.
- ▶ The `sysopts` statement:
 - The `datatype=text` informs Sterling Connect:Direct that text file line endings need to be adjusted to correspond to the remote systems line endings.
 - The `xlate=no` statement informs Sterling Connect:Direct that ASCII to EBCDIC translation does not need to occur for this transfer.
 - The `ckpt=2M` statement informs Sterling Connect:Direct that a checkpoint should be taken every two megabytes. If the file transfer has to be restarted for any reason, then the transfer will be resumed from the last known checkpoint.
 - The `disp=rpl` statement informs Sterling Connect:Direct that if the file already exists on the remote system then it should be replaced.

To submit the Connect:Direct process execute the Sterling Connect:Direct command-line interface (`direct`). Type the `submit` command (Example 4-8).

Example 4-8 Launching a process from the Direct> prompt

```
$ /opt/cdunix/ndm/bin/direct
Direct> submit file=/opt/cdunix/sample_push.cd;
Process Submitted, Process Number = 31
Direct> quit;
```

You can obtain the log messages for Connect:Direct for Linux by typing `Select Statistics` at the same `Direct>` prompt, for example:

```
select statistics pnumber=31;
```

Alternatively, you can use the abbreviated form:

```
sel stat pnum=31;
```

You can use the `quit;` command to exit the `Direct>` prompt. Every command statement must be terminated with a semicolon.

Create and add the Connect:Direct process shown in Example 4-9 to the /opt/cdunix/sample_pull.cd. Save the file on SysE_CD.

Example 4-9 Process to pull a file from a remote Sterling Connect:Direct

```
/*
 * This sample process pulls a text file "msgfile.cfg"
 * from "snode" SysD_CD to "pnode" SysE_CD.
 */
```

ToSysE process snode=SysD_CD

```
step01 copy
  from( file = FromSysD.me
        sysopts="datatype(text) xlate(no)"
        snode
      )

  ckpt = 2M
  compress extended

  to( file = msgfile.cfg
      sysopts=":datatype=text:xlate=no:"
      pnode
      disp = rpl
    )
```

pend;

Execute the pull process using the same instructions as shown in Example 4-8 on page 82.

4.4.2 Sterling Connect:Direct push file to WebSphere MQ File Transfer Edition

Create and add the Connect:Direct process shown in Example 4-10 to the /opt/cdunix/sample_push_runtask_E-D.cd.txt file.

Example 4-10 Push a file and trigger a WebSphere MQ File Transfer Edition file transfer

```
/*
 * This sample process copies a text file "FromE.txt"
 * from "pnode" SysE_CD to "snode" SysD_CD.
 */
```

ToSysD process snode=SysD_CD

```
step01 copy
  from( file = FromE.txt
        sysopts=":datatype=text:xlate=no:"
        pnode
      )

  ckpt = 2M
  compress extended

  to( file = FromE.txt
      sysopts="datatype(text) xlate(no)"
```

```

        snode
        disp = (rpl)
    )

    if ( step01 == 0 ) then
        step02 run task snode
            sysopts="pgm(C:\Program
Files\IBM\WMQFTE\bin\fteCreateTransfer.cmd)args(-sa SYSDAGT -dm FTEQMGR -da
SYSCAGT -dm FTPQMGR -df C:\downloads\FromE.txt
C:\CDWindows_files\download\FromE.txt)"
        eif

```

pend;

4.4.3 Sterling Connect:Direct pull file from WebSphere MQ File Transfer Edition

Create and add the Connect:Direct process shown in Example 4-11 to the /opt/cdunix/sample_pull_runtask_E-D.cd.txt file.

Example 4-11 Trigger a WebSphere MQ File Transfer Edition transfer and pull using Sterling Connect:Direct

```

/*
 * This sample process pulls a text file "FromSysC_CD.txt"
 * from "snode" SysD_CD to "pnode" SysE_CD.
 */

FromSysC process snode=SysD_CD

    step01 run task snode
        sysopts="pgm(C:\Program
Files\IBM\WMQFTE\bin\fteCreateTransfer.cmd)args(-sa SYSCAGT -sm FTPQMGR -da
SYSDAGT -dm FTEQMGR -df C:\CDWindows_files\upload\FromSysC.txt
C:\downloads\FromSysC.txt)"

    if ( step01 == 0 ) then
        step02 copy
            from
                ( file = FromSysC.txt
                  sysopts="datatype(text) xlate(no)"
                  snode
                )
            to
                ( file = FromSysC_CD.txt
                  sysopts=":datatype=text:xlate=no:"
                  pnode
                  disp = rpl
                )

```

```
)  
    eif  
  
pend;
```

4.4.4 WebSphere MQ File Transfer Edition push to Sterling Connect:Direct

This section details how to run the file transfer where WebSphere MQ File Transfer Edition pushes files to Sterling Connect:Direct. The file transfer is initiated by an Ant script on SysC. A file is sent from the WebSphere MQ File Transfer Edition agent SYSCAGT to SYSDAGT. SYSDAGT writes the file to the file system and launches a Connect:Direct process. The Connect:Direct node SysD_CD pushes the file to SysE_CD.

To initiate the file transfer, open a command window and run the command shown in Example 4-12 on SysC.

We use the **fteAnt** command to launch the Ant script ("Push_to_CD.xml script" on page 402).

Addition resource: For more information about the **fteAnt** command, see the WebSphere MQ File Transfer Edition Information Center at:

<http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.doc/fteant.htm>

Example 4-12 Launch the Ant script Push_to_CD.xml

```
C:\FTE_CD>fteAnt -f Push_to_CD.xml  
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED  
BFGCL0211I:  
BFGCL0211I: init:  
BFGCL0211I:  
BFGCL0211I: copy:  
BFGAN0046I: Issuing request to copy file from 'SYSCAGT@FTPMGR' to 'SYSDAGT@FTEQ  
MGR'.  
BFGAN0048I: Copy operation assigned transfer ID: 414d5120465450514d475220202020  
02952dc4c213cd503  
BFGAN0050I: Successfully completion of copy operation: 414d5120465450514d4752202  
02020202952dc4c213cd503  
BFGCL0210W: [Info] uuid for copy=414d5120465450514d475220202020202952dc4c213cd50  
3  
BFGCL0211I:  
BFGCL0211I: post:  
BFGAN0070I: Issuing call request to 'SYSDAGT@FTEQMGR'.  
BFGAN0071I: Call operation assigned transfer ID: 414d5120465450514d475220202020  
02952dc4c213cd80c  
BFGCL0211I:  
BFGCL0211I: job:
```

The flow of the Ant script file transfer is:

1. The source file is located in the C:\sysctmp\TestFile_from_SysC.txt directory on SysC.
2. The file is sent to SysD and is located in the C:\CDWindows_files\upload directory as a temporary file, the file name of which is the job number.

The C:\CDWindows_files\upload directory is a directory that is configured in Connect:Direct (Figure 4-21 on page 73).

3. The file is sent to SysE and located in the /opt/cdunix/download directory as the TestFile_from_SysC.txt file.

The /opt/cdunix/download directory is a directory that is configured in Connect:Direct (Example 4-2 on page 75).

Make sure that the TestFile_from_SysC.txt file arrived at the /opt/cdunix/download directory on SysE.

The file transfer consists of two parts:

- ▶ The Sterling Connect:Direct file transfer
- ▶ The WebSphere MQ File Transfer Edition file transfer

You have two kinds of transfer logs. The log messages for Sterling Connect:Direct can be obtained through a CLI or using the desktop client called Connect:Direct Requester. For instructions on how to obtain these logs, see “Sterling File Gateway and Sterling B2B Integrator log files” on page 384.

Figure 4-23 shows the log accessed by the Connect:Direct Requester.

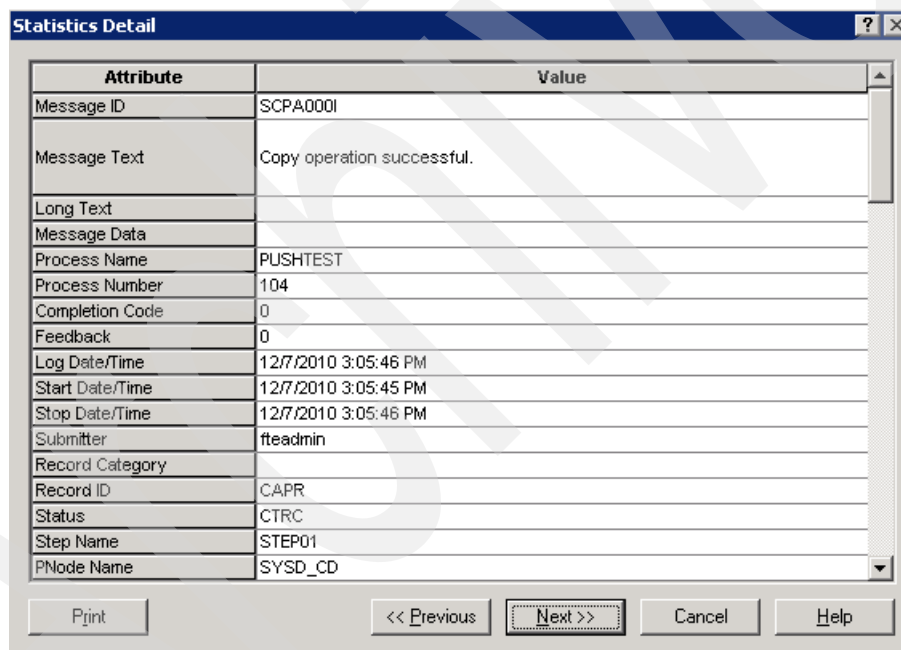


Figure 4-23 Connect:Direct Requester Statistics Detail dialog box for Push_to_CD.xml

Example 4-13 shows the log, accessed by the CLI.

Example 4-13 The log for Push_to_CD.xml access by the CLI

```
> select statistics pnum=104;
=====
                          Select Statistics
=====
P  RECID  LOG TIME                PNAME    PNUMBER  STEPNAME  CCOD  FDBK   MSGID
E  RECID  LOG TIME                MESSAGE TEXT
-----
P  SUBP   12/07/2010 15:05:45  PUSHTEST  104             0    0     LCCA013I
```

```

E QCEX 12/07/2010 15:05:45

P PSTR 12/07/2010 15:05:45 PUSHTEST 104          0  0      LSMG200I
E SPCA 12/07/2010 15:05:45 Secure+ bypassed for remote node &NODE.

E SSTR 12/07/2010 15:05:45 PNODE session started - remote node SysE_CD

P LSST 12/07/2010 15:05:46 PUSHTEST 104      STEP01
P CTRC 12/07/2010 15:05:46 PUSHTEST 104      STEP01  0  0      SCPA000I
P PRED 12/07/2010 15:05:46 PUSHTEST 104      0  0      LSMG252I

```

The status of the transfer is also available in the WebSphere MQ Explorer Content view and progress view at the bottom of the panel (Figure 4-24).

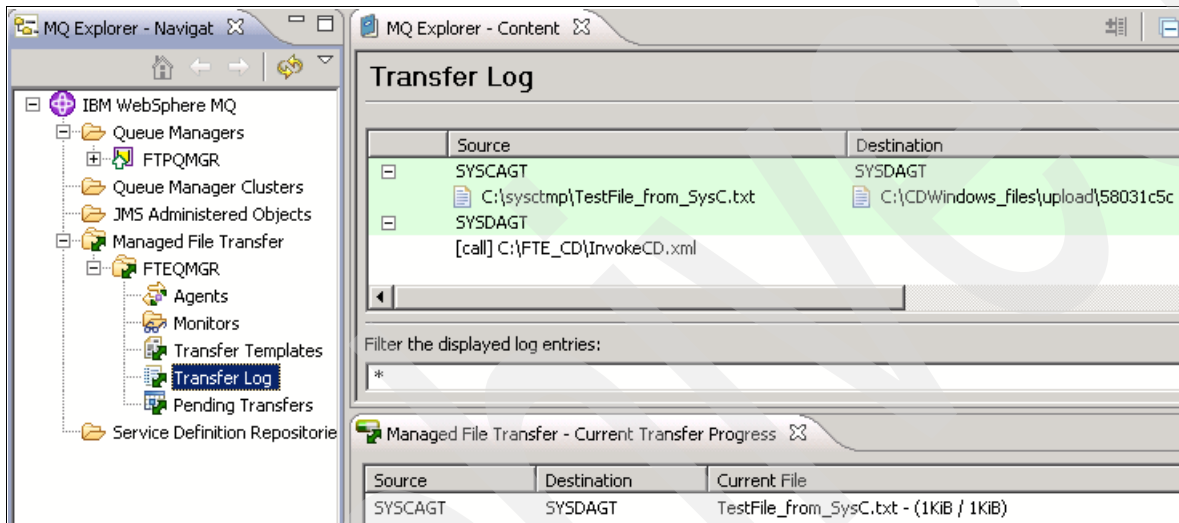


Figure 4-24 The File transfer logs for Push_to_CD.xml

4.4.5 WebSphere MQ File Transfer Edition pull to Sterling Connect:Direct

This section describes how to run a file transfer where WebSphere MQ File Transfer Edition pulls files to Sterling Connect:Direct. The file transfer is initiated by an Ant script on SysC. The WebSphere MQ File Transfer Edition SYSDAGT launches a Connect:Direct process that instructs the Connect:Direct node SysE_CD to send the file to SysD_CD. SysE_CD sends the file to SysD_CD using Sterling Connect:Direct. The file is transferred from SYSDAGT to SYSCAGT using WebSphere MQ File Transfer Edition.

To initiate the file transfer, open a command prompt and run the command shown in Example 4-14 on SysC. We used the `fteAnt` command to launch the Ant script ("Pull_from_CD.xml script" on page 404).

Example 4-14 Launch the Ant script Pull_from_CD.xml

```

C:\FTE_CD>fteAnt -f Pull_from_CD.xml
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED
BFGCL0211I:
BFGCL0211I: init:
BFGCL0211I:
BFGCL0211I: pre:
BFGAN0070I: Issuing call request to 'SYSDAGT@FTEQMGR'.
BFGAN0071I: Call operation assigned transfer ID: 414d5120465450514d475220202020

```

02952dc4c2092d403
BFGCL0211I:
BFGCL0211I: copy:
BFGAN0046I: Issuing request to copy file from 'SYSDAGT@FTEQMGR' to 'SYSCAGT@FTPQ
MGR'.
BFGAN0048I: Copy operation assigned transfer ID: 414d5120465450514d4752202020202
02952dc4c2092d407
BFGAN0050I: Successfully completion of copy operation: 414d5120465450514d4752202
02020202952dc4c2092d407
BFGCL0210W: [Info] uuid for copy=414d5120465450514d475220202020202952dc4c2092d40
7
BFGCL0211I:
BFGCL0211I: job:

The flow of the Ant script is:

1. The source file is located in the /opt/cdunix/upload directory on SysE.
2. The file is sent to SysD and located in the C:\CDWindows_files\download directory as a temporary file. The file name is the job number.

The C:\CDWindows_files\download directory is a directory that is configured in Sterling Connect:Direct (Figure 4-21 on page 73).

3. The file is sent to SysC and is located in the C:\Documents and Settings\fteadmin directory as the TestFile_from_SysE.txt file.

The C:\Documents and Settings\fteadmin directory is the home directory for the fteadmin user that started the agent process.

Make sure that the TestFile_from_SysE.txt file arrived at the C:\Documents and Settings\fteadmin directory on SysE.

The file transfer consists of two parts:

- ▶ The Sterling Connect:Direct file transfer
- ▶ The WebSphere MQ File Transfer Edition file transfer

There are two kinds of transfer logs. The log messages for Sterling Connect:Direct can be obtained through a CLI or using the desktop client called Connect:Direct Requester. For instructions on obtaining these logs, see “Sterling File Gateway and Sterling B2B Integrator log files” on page 384.

Figure 4-25 shows the log accessed by Connect:Direct Requester.

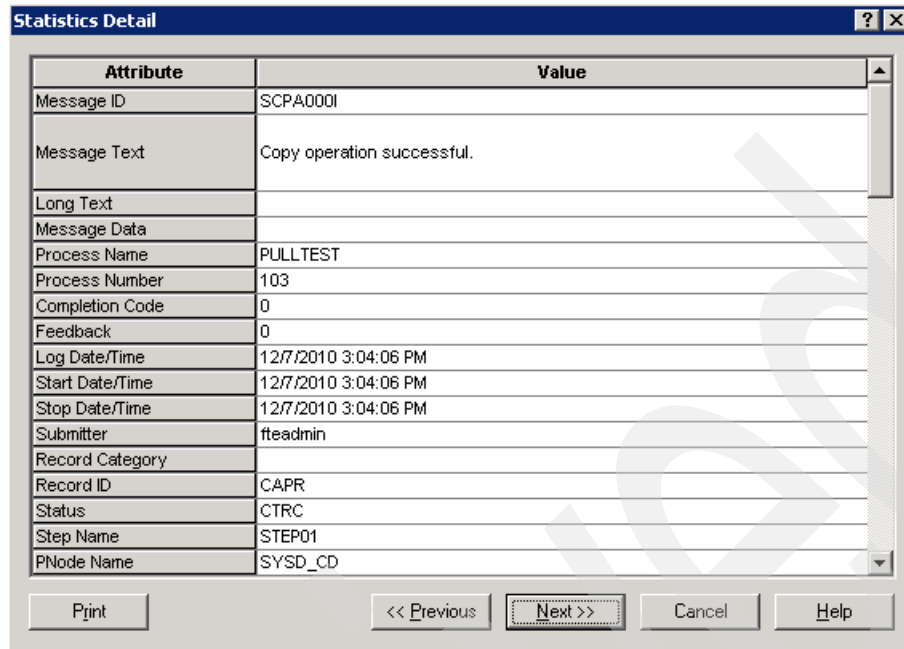


Figure 4-25 Connect:Direct Requester Statistics Detail dialog for Pull_to_CD.xml

Example 4-15 shows the log access by the CLI.

Example 4-15 The log for Pull_to_CD.xml access by the CLI

```
> select statistics pnum=103;
=====
                          Select Statistics
=====
P RECID  LOG TIME          PNAME    PNUMBER STEPNAME CCOD  FDBK  MSGID
E RECID  LOG TIME          MESSAGE TEXT
-----
P SUBP   12/07/2010 15:04:05 PULLTEST 103          0    0    LCCA013I
E QCX    12/07/2010 15:04:05
P PSTR   12/07/2010 15:04:05 PULLTEST 103          0    0    LSMG200I
E SPCA   12/07/2010 15:04:05 Secure+ bypassed for remote node &NODE.
E SSTR   12/07/2010 15:04:06 PNODE session started - remote node SysE_CD
P LSST   12/07/2010 15:04:06 PULLTEST 103    STEP01
P CTRC   12/07/2010 15:04:06 PULLTEST 103    STEP01    0    0    SCPA000I
P PRED   12/07/2010 15:04:06 PULLTEST 103          0    0    LSMG252I
=====
```

You can also view the transfer in the WebSphere MQ Explorer Content view and view the progress at the bottom of the panel (Figure 4-26).

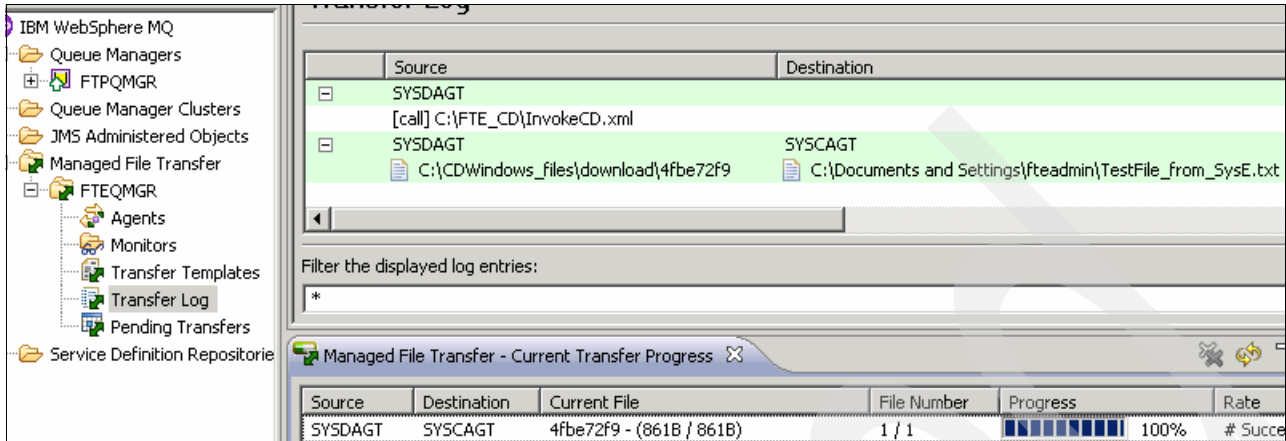


Figure 4-26 The file transfer logs for the Pull_from_CD.xml file

4.5 Troubleshooting tips

If you experience difficulty running the scenario, see Appendix C, “Troubleshooting” on page 379.

External transfers using IBM Sterling Connect:Direct and IBM Sterling File Gateway

This chapter shows how external file transfers using Sterling Connect:Direct can have enhanced business value with the addition of Sterling File Gateway. Organizations often receive files that need transformation, parsing, and routing. By combining Sterling File Gateway with a Sterling Connect:Direct managed file transfer solution, a file's journey into, throughout, and out of an organization to external trading can function seamlessly.

To demonstrate the use of Sterling File Gateway combined with external managed file transfer in this chapter, we integrate Sterling File Gateway and Sterling Connect:Direct as the protocol for managed file transfer. This chapter includes the following topics:

- ▶ Solution overview
- ▶ Scenario details
- ▶ Configuring the solution components
- ▶ Testing the flows
- ▶ Troubleshooting

5.1 Solution overview

This scenario shows how a file can come into an organization through Sterling File Gateway. Files are transferred using the Sterling Connect:Direct protocol over the public internet through a secure, hardened interface in the DMZ into an organization's secure, internal network.

Sterling File Gateway is an application for transferring files between partners using different protocols, file naming conventions, and file formats. Sterling File Gateway also supports integration with Sterling Connect:Direct. Sterling File Gateway is delivered as part of the Sterling B2B Integrator platform with a unique application URL that provides single sign-on (SSO) access to the Sterling B2B Integrator administrative console through menu selection.

In this chapter, we demonstrate both a file upload and a file download example using Sterling File Gateway and Sterling Connect:Direct. When a file is received by Sterling File Gateway and placed in the partner user ID mailbox, it is routed throughout the organization. With this type of configuration, organizations can interact with any external Sterling Connect:Direct trading partner and then route the file throughout the organization.

5.1.1 Appropriate use of the scenario

This scenario demonstrates the Sterling Connect:Direct communication between an external trading partner and Sterling File Gateway. The Sterling Connect:Direct communication is shown by using the Sterling Connect:Direct Adapter to allow the external trading partner to access Sterling File Gateway to upload and download files. Sterling Connect:Direct is one of many application layer protocols that you can use with Sterling File Gateway to trade data with external trading partners. The following possible protocols are supported:

- ▶ FTP
- ▶ FTP/S
- ▶ SSH/SCP
- ▶ AS2
- ▶ AS3
- ▶ Odette FTP

Encryption note: You need to encrypt any data that flows between internal and external trading partners. Some of the scenarios that we show in this chapter represent file transfers to and from an external entity using Sterling Connect:Direct. You can encrypt the Sterling Connect:Direct protocol using the IBM Sterling Connect:Direct Secure Plus option. Alternatively, you can encrypt the channel between two trading partners with a separate solution.

To modify the scenario to use another protocol, you need to verify that the appropriate adapter is installed and configured in IBM Sterling B2B Integrator and to configure partners and routing channels in Sterling File Gateway that might be needed.

5.1.2 Business value

Sterling File Gateway is an application for transferring files between partners using different protocols, file naming conventions, and file formats. This solution uses the Sterling B2B foundation, which includes IBM Sterling B2B Integrator, IBM Sterling Standards, and the IBM Sterling platform, to deliver capabilities similar to those found in IBM Sterling File Transfer

Service and IBM Sterling Connect:Enterprise for UNIX, while adding new features and functionality.

Use Sterling File Gateway for movement of large and high-volume file transfers, with end-to-end visibility of file movement in a process-oriented and highly scalable framework that alleviates file transfer challenges, such as protocol and file brokering, automation, and data security.

Sterling File Gateway supports integration with Sterling B2B Integrator Mailbox, IBM Sterling Control Center, Sterling Connect:Enterprise for UNIX server products, and Sterling Connect:Direct. Sterling File Gateway is delivered as part of the Sterling B2B Integrator platform with a unique application URL and provides SSO access to the Sterling B2B Integrator administrative console through menu selection.

5.1.3 Sterling File Gateway features

Sterling File Gateway provides the following features:

- ▶ File and file name transformations

Input is mapped to:

- Output file names
- System-wide, group, and partner-specific policies
- Common file processing tasks, such as compression and decompression
- Pretty Good Privacy (PGP) encryption and decryption
- Signing

- ▶ File transfer visibility

Events are recorded for monitoring and reporting. Detailed tracking is available for input-output file structure processing and dynamic route determination. You can view and filter Sterling File Gateway data flows for all users.

- ▶ Replay and redeliver

One-click replay and redeliver allows users to reprocess a transmission from the beginning or to resend just the processed file to a specific delivery destination.

- ▶ Notifications

Partners and operators can subscribe to be notified about events by email.

- ▶ Predefined business processes

Common behaviors are defined in file-transfer scenarios, reducing the need for customization.

- ▶ Extensibility

You can add custom event codes, protocols, and consumer identification policies to support unique scenarios.

- ▶ Broad communications protocol support

FTP, FTP/S, SSH/SFTP, SSH/SCP, and Sterling Connect:Direct are supported upon installation. You can also configure additional protocols (such as AS2, AS3, or Odette FTP) using the extensibility feature.

- ▶ Partner Interface (myFileGateway)

The web browser-based interface enables partners to upload and download files, to subscribe to notifications of events, to manage passwords, to search and view file transfer activity, and to generate reports about file transfer activity.

- ▶ Flexible mailbox structures

You can specify mailbox structures that use pattern matching policies and specify attributes that must be true of all partners or a subset of partners.

- ▶ Dynamic routing

Consumers are derived at run time, either through mailbox structure, file name, business process-derived consumer name, or map-derived consumer name.

- ▶ Partner onboarding

An easy-to-use GUI is available to onboard partners and to configure the various combinations of communication protocols to enable Sterling File Gateway operations.

Sterling File Gateway represents the next generation for enterprise-level file transfer. It includes all the features of managed file transfer and adds the following new capabilities:

- ▶ A *partner* belongs to exactly one community, but can belong to more than one partner group, which is a way to combine partners for business purposes.
- ▶ An *integration architect* can configure the Sterling File Gateway mailbox hierarchy to match the configuration with which *partners* are already familiar.
- ▶ The structure for mailboxes is defined flexibly.
- ▶ Sterling File Gateway can perform format unwrapping and wrapping for the ZIP, GZIP, and PGP formats.
- ▶ Sterling File Gateway can extract facts from file names and use that information for routing and delivery and as input for generating the file name that the consumer sees.
- ▶ Producer and consumer mailboxes are not tightly constrained. Both producer and consumer mailbox patterns can be built from facts that are available when a routing channel is provisioned. Consumer mailbox patterns can also include facts that are available only when a file is being routed.

5.2 Scenario details

In this scenario, we demonstrate that multiple enterprises can exchange files to each other between Sterling File Gateway and Sterling Connect:Direct. Data is transferred through a proxy server in the DMZ to Sterling File Gateway. Sterling File Gateway is used to perform the routing of the file and protocol conversion between Sterling Connect:Direct.

We show how to configure and set up Sterling File Gateway to enable Sterling Connect:Direct to be enabled as an available transfer protocol. Specifically, configuration of Sterling B2B Integrator to use the Connect:Direct server adapter is needed for the scenario to work successfully.

In an inbound scenario, with Sterling Connect:Direct, the external trading partner uploads a file to Connect:Direct server adapter through a proxy server and puts the file to the mailbox in Sterling File Gateway. The file is then directed through Sterling File Gateway routing channels to an internal Connect:Direct node.

In an outbound scenario, an internal trading partner uploads a file to the Sterling File Gateway inbox with Sterling Connect:Direct. The file is then directed to the Connect:Direct server adapter mailbox in the same manner as the inbound scenario and is then sent to the destination using the proxy server to the external Sterling Connect:Direct.

5.2.1 Solution components

This section describes the components that are associated with each product in this solution (Figure 5-1). Certain components require specific configuration for the solution to work. We discuss the configuration steps that are required as necessary.

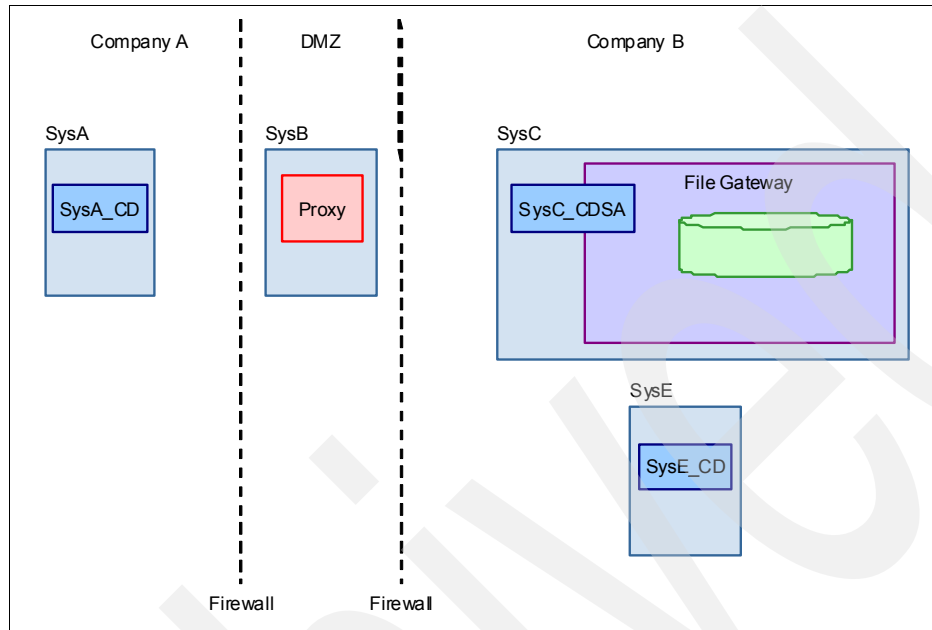


Figure 5-1 Components for external transfers using Sterling Connect:Direct and Sterling File Gateway

Sterling Connect:Direct

Sterling Connect:Direct moves files point-to-point (peer-to-peer) using the Sterling Connect:Direct protocol. A Sterling Connect:Direct client is used to communicate with a Connect:Direct server regarding the work that will be performed. The Connect:Direct requester GUI and CLI are used to communicate with the Connect:Direct server.

The scenario uses the following components:

- ▶ Connect:Direct node SysA_CD

The Connect:Direct node that is installed on SysA is named *SysA_CD*. For the scenarios in this chapter, *SysA_CD* represents an external trading partner at one endpoint.

SysA_CD is a Connect:Direct node in Company A (Figure 5-1). The files transferred in the scenarios in this chapter both begin and end at this endpoint.

- ▶ Connect:Direct node SysE_CD

The Connect:Direct node installed on SysE is named *SysE_CD*. For the scenarios in this chapter, *SysE_CD* represents an internal trading partner at one endpoint. *SysE_CD* is a Connect:Direct node in Company B (Figure 5-1). The files transferred in the scenarios in this chapter both begin and end at this endpoint.

- ▶ **Connect:Direct requester for Windows**

Connect:Direct requester for Windows is a GUI for Connect:Direct servers on the Microsoft Windows, UNIX, and Open VMS platforms. You can connect to Connect:Direct servers to perform file transfers, to run remote programs or batch jobs, and to create, submit, and monitor Connect:Direct processes. You can also perform Sterling Connect:Direct administrative functions, such as setting up and maintaining Sterling Connect:Direct users and network maps. In this chapter, the Connect:Direct requester is used to configure and launch Connect:Direct processes on SysA_CD.

- ▶ **Connect:Direct CLI**

The Connect:Direct CLI is a text-based user interface to Connect:Direct servers on the UNIX platform. You can connect to Connect:Direct servers to perform file transfers, run remote programs or batch jobs, and create, submit, and monitor Connect:Direct processes. In this chapter, the Sterling Connect:Direct CLI is used to configure and launch Connect:Direct processes on SysE_CD.

Connect:Direct server adapter

The Connect:Direct server adapter is a component of Sterling B2B Integrator that allows Sterling B2B Integrator to act like a Connect:Direct server in sending and receiving data using the Connect:Direct protocol.

Proxy server

The proxy server can be any proxy server or other DMZ-hardened security mechanism. This security piece validates that the external connection is coming from an approved domain, over the port that is specified for the protocol that is used. The proxy server terminates the external session, begins a secure session to continue back to the protected network, and authenticates users.

Sterling File Gateway

Sterling File Gateway routes incoming and outgoing files based on defined partners and their mailboxes. Sterling File Gateway uses Sterling B2B Integrator to switch protocols in this scenario. The mailboxes are created based on partner definitions. Sterling File Gateway can be administered in the following ways:

- ▶ Sterling File Gateway Administration Console is a web-based GUI that allows administrators to create partners, routes, and channel templates.
- ▶ Sterling B2B Integrator Administration Console is a web-based GUI that allows administrators to perform actions such as creating business processes, defining trading partners, creating server adapters, and configuring protocols.
- ▶ Sterling B2B Integrator Import/Export Configuration allows administrators to configure one Sterling File Gateway instance that can be exported and then imported into another Sterling File Gateway instance.

In our scenario, we created additional mailboxes for each unique routing. Table 5-1 lists the use for each mailbox and to which mailbox it sends files.

Table 5-1 Sterling File Gateway mailboxes

External Trading Partner mailbox	Usage	Routing direction	Internal mailbox	Use
/SysA_CD_Partner/ To_SysE_Partner	Receives a file from SysA_CD_Partner that is sent from the SysA_CD Connect:Direct node. The routing channel is configured to route the file to SysE_Partner.	Inbound	/SysE_Partner/Inbox	The file is never placed in SysE_Partner/Inbox. The file is routed directly from SysA_CD_Partner/To_SysE_Partner mailbox to SysE_Partner and then to SysE using the Connect:Direct Server Adapter.
/SysE_Partner/	Receives a file from the internal Connect:Direct node SysE_CD that is running on SysE. The file is then routed to the external partner defined by SysA_CD_Partner.	Outbound	/SysA_CD_Partner/Inbox	The file is never placed into the SysA_CD_Partner/Inbox mailbox. The file is routed from the SysE_Partner root mailbox directly to the SysA_CD node (as defined by SysA_CD_Partner) using the Connect:Direct Server Adapter.

Sterling B2B Integrator

Sterling B2B Integrator runs under Sterling File Gateway and provides protocol switching. Sterling B2B Integrator is available as a stand-alone product. It provides the following features:

- ▶ The ability to manage and grow trading partner communities
- ▶ Adapters for back-end applications
- ▶ Role-based data access and system operation
- ▶ Data transport security and data encryption support
- ▶ Digital signature support
- ▶ Identity management, including authorization and authentication

5.2.2 Sterling Connect:Direct to Sterling Connect:Direct using Sterling Secure Proxy

Figure 5-2 portrays a transfer from one Connect:Direct node to another Connect:Direct node using Sterling Secure Proxy. A Connect:Direct node named SysA_CD is installed on the system named SysA. Additionally, a Connect:Direct node named SysE_CD is installed on SysE. In between the Connect:Direct node is an instance of Sterling Secure Proxy in the DMZ.

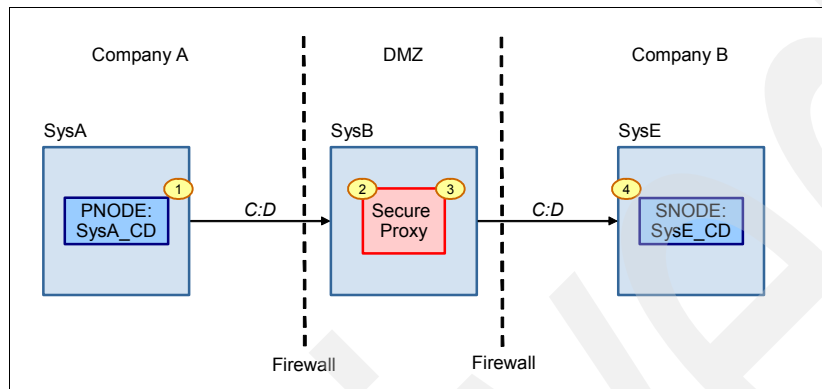


Figure 5-2 Sterling Connect:Direct to Sterling Secure Proxy to Sterling Connect:Direct

A pre-existing file on SysA flows from the SysA_CD node to the SysE_CD node. The file is written to disk on SysE. The sequence of events is as follows:

1. A plain-text Connect:Direct process file is created on SysA. A Connect:Direct process contains the simple command statements that describe the work for SysA_CD to perform. The name of the file on SysA is `sample.cd`. On SysA_CD, a user logs in to Sterling Connect:Direct using Connect:Direct requester with a user ID that has privileges to log in to Sterling Connect:Direct. Within the Connect:Direct requester, the user clicks the menu options to instruct SysA_CD to read and parse the `sample.cd` Connect:Direct process file.
2. SysA_CD establishes a network connection to SysB, the Sterling Secure Proxy.
3. SysB establishes another network connection to SysE_CD.
4. The file is transferred over the network from SysA_CD to SysB and then on to SysE_CD. The Connect:Direct node SysE_CD writes the file to the SysE system. SysA_CD terminates the network connection to SysB, which in turn terminates the connection to SysE_CD.

5.2.3 External Sterling Connect:Direct push to Sterling File Gateway using Sterling Secure Proxy

Figure 5-3 portrays a transfer from one Connect:Direct node to Sterling File Gateway using Sterling Secure Proxy. At Sterling File Gateway, a routing channel built for SysA_CD is processed and the file is forwarded to SysE_CD. A Connect:Direct node named SysA_CD is installed on SysA. A Sterling File Gateway instance named SysC_CDSA resides on SysC. A Connect:Direct node named SysE_CD is installed on SysE. Between the Connect:Direct node SysA_CD and the Sterling File Gateway instance named SysC_CDSA is an instance of Sterling Secure Proxy in the DMZ. Between the Sterling File Gateway instance named SysC_CDSA and the Connect:Direct node SysE_CD is an instance of Sterling Secure Proxy in the DMZ.

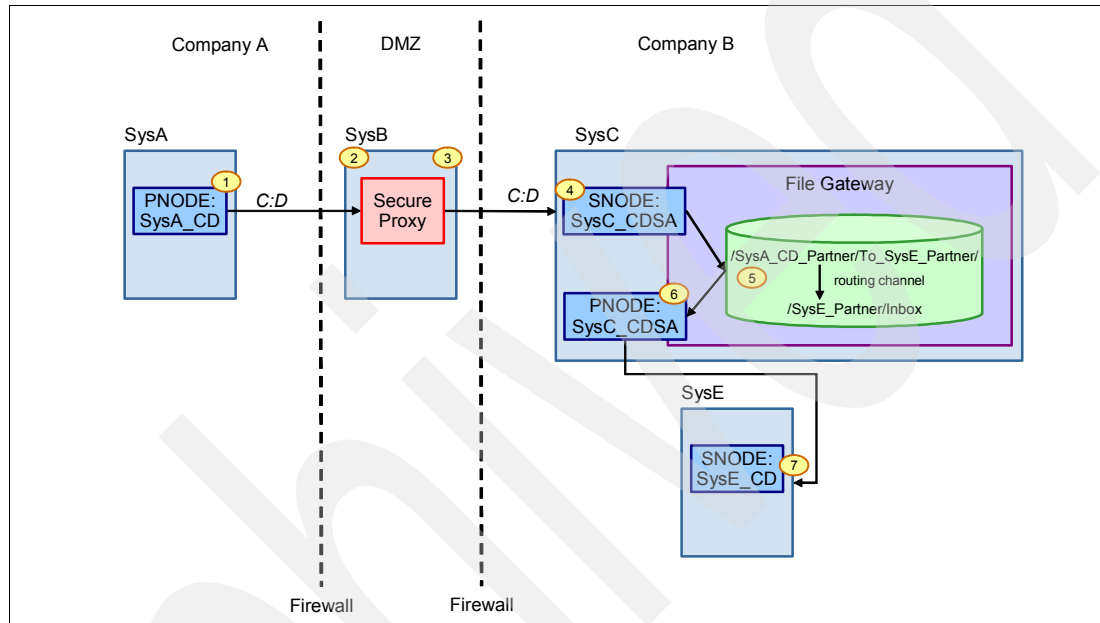


Figure 5-3 Sterling Connect:Direct to Sterling Secure Proxy to Sterling Connect:Direct to Sterling File Gateway to Sterling Connect:Direct to Sterling Connect:Direct

A pre-existing file on SysA flows from the SysA_CD node through the Sterling Secure Proxy, then to the SysC_CDSA instance, and finally to SysE_CD node. The file is written to disk on SysE. The sequence of events is as follows:

1. A plain-text Connect:Direct process file is created on SysA. A Connect:Direct process contains the simple command statements that describe the work for SysA_CD to perform. The name of the file on SysA is `sample.cd`. On SysA_CD, a user logs in to Sterling Connect:Direct using Connect:Direct requester with a user ID that has privileges to log in to Sterling Connect:Direct. Within the Connect:Direct requester, the user clicks the menu options to instruct SysA_CD to read and parse the `sample.cd` Connect:Direct process.
2. SysA_CD establishes a network connection to SysB, the Sterling Secure Proxy.
3. SysB establishes another network connection to SysE_CD.
4. The file is transferred over the network from SysA_CD to SysC_CDSA.
5. Sterling File Gateway processes the routing channel rules based on files that are received from SysA_CD. These rules dictate that the file should be sent to SysE_CD.
6. Sterling File Gateway triggers SysC_CDSA to establish a network connection to SysE_CD and to transfer the file to it.
7. The Connect:Direct node SysE_CD receives the file and writes the file to the SysE system.

5.2.4 Internal Sterling Connect:Direct push to Sterling File Gateway using Sterling Secure Proxy

Figure 5-4 portrays a transfer from an internal Connect:Direct node to Sterling File Gateway. At Sterling File Gateway a routing channel is processed built for SysE_CD, which forwards the file to the external SysA_CD node. Sterling File Gateway sends the file to SysA_CD using Sterling Secure Proxy on SysB.

A Connect:Direct node named SysA_CD is installed on SysA. A Sterling File Gateway instance named SysC_CDSA resides on SysC. A Connect:Direct node named SysE_CD is installed on SysE. Between the Connect:Direct node SysA_CD and the Sterling File Gateway instance named SysC_CDSA is an instance of Sterling Secure Proxy in the DMZ.

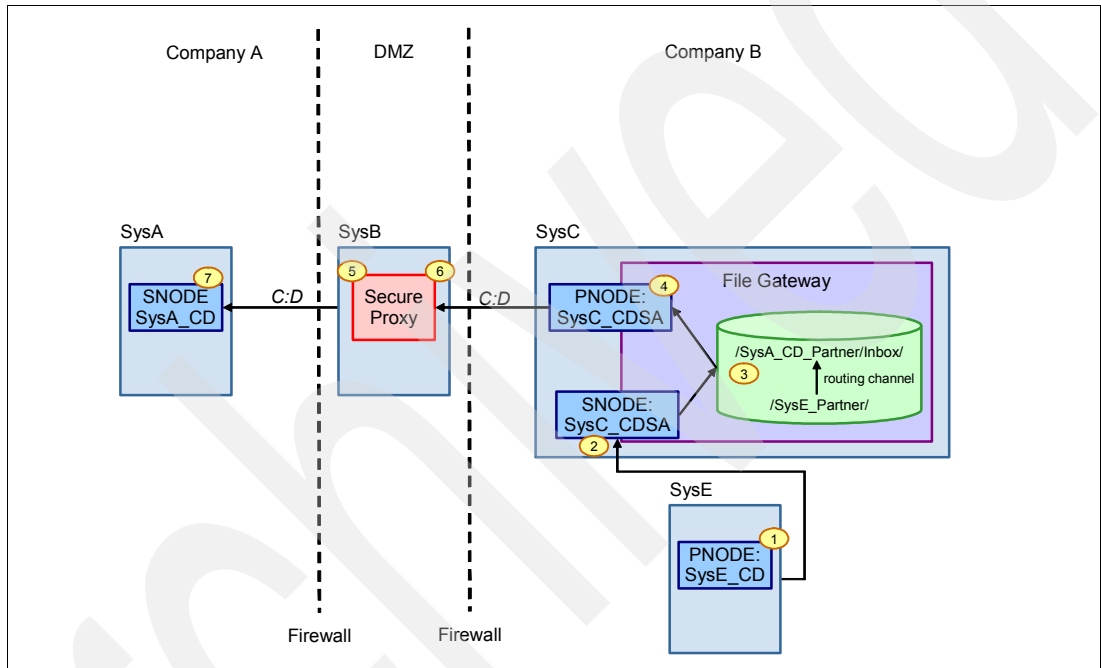


Figure 5-4 Sterling Connect:Direct to Sterling Connect:Direct to Sterling File Gateway to Sterling Connect:Direct to Sterling Secure Proxy to Sterling Connect:Direct

A pre-existing file on SysE flows from the SysE_CD node to Sterling File Gateway (SysC_CDSA). From Sterling File Gateway, the file travels from SysC_CDSA through Sterling Secure Proxy to SysA_CD. The sequence of events is as follows:

1. A plain-text Connect:Direct process file is created on SysE. A Connect:Direct process contains the simple command statements that describe the work for SysE_CD to perform. The name of the file on SysE is `sample.cd`. On SysE_CD, a user logs in to Sterling Connect:Direct using the Sterling Connect:Direct CLI with a user ID that has privileges to log in to Sterling Connect:Direct. Within the CLI, the user types the commands to submit and parse the `sample.cd` Connect:Direct process file.
2. SysE_CD establishes a network connection to SysC_CDSA and transfers the file. The Connect:Direct node SysC_CDSA receives the file.
3. Sterling File Gateway processes the routing channel rules based on files that are received from SysE_CD. These rules dictate that the file is sent to SysA_CD.
4. Sterling File Gateway triggers SysC_CDSA to establish a network connection to SysE_CD and transfer the file to it.

5. Sterling Secure Proxy on SysB receives the network connection request from SysC_CDSA.
6. Sterling Secure Proxy establishes another network connection to SysA_CD.
7. The file is transferred over the network from SysC_CDSA to SysA_CD.

5.2.5 Protocols

This scenario uses Sterling Connect:Direct protocol to transfer files between the external trading partner Company A (SysA_CD) to the systems in Company B. The Sterling Connect:Direct protocol is used to communicate with Sterling File Gateway in Company B's protected zone.

5.2.6 Security

To perform work in your enterprise and with external trading partners, Sterling Connect:Direct and Sterling File Gateway rely on building blocks of information that define the local and remote nodes, users who can access those nodes, and the functions that users can perform.

Sterling File Gateway and Sterling B2B Integrator

Sterling File Gateway and Sterling B2B Integrator provide the following security functions:

- ▶ Identity management, including authorization and authentication
- ▶ Perimeter security at DMZ traversal
- ▶ Role-based data access and system operation
- ▶ Secured mailboxing repository
- ▶ Data transport security (SSL and SFTP/SSH) and data encryption (S/MIME and PGP) support
- ▶ Non-repudiation using the AS2 or AS3 protocol
- ▶ Digital signature support
- ▶ Message-level and transport-level security based on WS-Security 1.0 compliance, including WS-I Basic Profile 1.1 and Basic Security Profile 1.0

Sterling Connect:Direct

To perform work in your enterprise, Sterling Connect:Direct and the Connect:Direct Adapter on Sterling B2B Integrator rely on building blocks of information that define the local and remote nodes, users who can access those nodes, and the functions that these users can perform.

Local node definition

During installation, you define a local node for Sterling Connect:Direct. The local node definition specifies information such as the operating system, default user ID, TCP/IP address, and port number. After installation, you can change the local node's settings and define remote nodes. In addition to the default user ID that you specify with a local node, you can add other users who can access that node.

Local user authorities

After you define a user ID for each user who has access to the local node, you can restrict the ability of each user to perform certain tasks by defining user authorities for each user ID. For example, you can permit a user to submit a process but not to monitor or delete processes.

Sterling Connect:Direct has administrators and general users, and each type of user has a set of default privileges. You can use these user templates to assign user authorities and to restrict user privileges. Local user authorities provide one type of authentication in Sterling Connect:Direct. An alternative method of authentication is available using remote user proxies. For a listing of the default authorities for each type, see the product documentation for your Sterling Connect:Direct platform.

Remote user proxies

User proxy definitions (referred to as *secure points of entry* on the mainframe) contain remote user information for operations that are initiated from remote Connect:Direct node. These definitions identify a proxy relationship between a user ID at a remote Connect:Direct node and a local user ID. This mapping of remote and local user IDs enables users at remote Connect:Direct node to submit work to the local Connect:Direct node without explicitly defining user IDs and passwords in the processes, thus eliminating the need to share passwords with trading partners. User proxies also define activities that each user ID can perform on the local Connect:Direct node.

Configuration settings for the local node

Initialization parameters determine various Sterling Connect:Direct settings that control system operation. You can create the initialization parameters file when you install Sterling Connect:Direct and can update it as needed. Some of these settings can be overwritten in the netmap, user authorities, user proxies, and processes.

Remote node definitions

You create the network map, or *netmap*, when you install Sterling Connect:Direct. This file identifies the remote nodes with which each local node can communicate and the communication information that is needed to establish a connection. You create a remote node entry in the network map for each remote node with which the local node communicates. Each network map entry contains information about the remote node, such as the remote node name, operating system, session characteristics for a protocol, and transfer and protocol information about the available communications paths and their attributes.

Netmap checking

In addition to defining the remote nodes that communicate with the Connect:Direct node, you can use the network map to perform a security function. Netmap checking verifies that inbound sessions are from a node that is defined in the network map. If the node is not in the network map, the connection fails.

The following methods of authenticating users outside Sterling File Gateway and Sterling B2B Integrator are supported:

- ▶ **Single sign-on (SSO)**

This method provides access control that enables a user to log in once to a company network or portal site to gain access to multiple software systems without logging in again. SSO bypasses the built-in authentication process in Sterling File Gateway and, instead, trusts that a user is authenticated by a third-party software.

- ▶ **Lightweight Directory Access Protocol (LDAP)**

This method provides a network protocol for accessing directories where user credentials are authenticated against an external LDAP directory instead of against the Sterling B2B Integrator database user table for access to Sterling File Gateway.

Firewall security

Firewall configuration plays an important role in securing connections to and from external partners and in protecting the internal network. The external firewall must allow incoming

requests from all of the trading partner's source IP addresses or range of IP addresses. You can configure firewall security in the firewall rules configuration. The method used to configure firewall rules depends on the model and type of firewall that you use.

The DMZ is a termination point at the edge of the protected network that typically is used to house internet-facing systems.

Setting up tight rules on the inner firewall is important for protecting internal systems. Typically, you set up inner firewall rules to allow traffic only from a mediation server in the DMZ that terminates the connection from the internet. The mediation server then re-establishes the connection through the inner firewall to a system for which the files are destined or to a system that moves the files to a back-end application.

For this scenario, we need to open a Sterling Connect:Direct port, 1354, both inside and outside the firewall. These ports vary based on the mediation server configuration in the DMZ and your organization's security policies and practices.

5.2.7 Software prerequisites

This scenario uses the following software:

- ▶ Sterling Connect:Direct for Microsoft Windows Version 4.5.01
- ▶ Sterling Connect:Direct for Linux Version 4.0
- ▶ Sterling File Gateway 2.1
- ▶ Sterling B2B Integrator 5.1.01

5.2.8 Configuring the solution components

This section describes the software configuration that is required for the scenarios that we describe in this chapter. You must configure Sterling Connect:Direct to communicate with a remote node and you must configure the Sterling File Gateway to route the files to their destination.

This section documents these configuration changes.

5.2.9 Configuration prerequisites

For the scenarios that we describe in this chapter, we make the following assumptions:

- ▶ Sterling Connect:Direct for Linux is installed and operating on SysE. Sterling Connect:Direct is installed in the `/opt/cdunix/` directory.
- ▶ Sterling Connect:Direct for Microsoft Windows is installed and operating on SysA. Sterling Connect:Direct is installed in the `C:\Program Files\Sterling Commerce\Connect Direct v4.5.01\` directory.
- ▶ An operating system user ID named `cdadmin` is created on SysE and SysA. The `cdadmin` user ID is given the default basic user rights on each system.
You can use another operating system user ID in place of the `cdadmin` user ID. If you use another user ID, replace `cdadmin` with that user ID in the examples that follow.
- ▶ Sterling B2B Integrator and Sterling File Gateway are installed on SysC.
- ▶ The 1364 port for Sterling Connect:Direct is open in the firewall (internal and external).

Security prerequisites: Review your local security policy and practices to determine what is appropriate for your production environment. While the scenarios in this book do not implement security, it is important that you take security into consideration when implementing these scenarios in your own environment.

5.2.10 Configuring Sterling Connect:Direct on SysA

The following sections describe how to configure the Connect:Direct nodes so that they are aware of each other. To simplify our examples, both nodes use the same operating system user ID. Both nodes also use the Sterling Connect:Direct preset upload and download directories for transferred files. Preset upload and download directories enable a process to be submitted without using full path and file names. Given only a file name, both Connect:Direct nodes will push or pull any file from the preset directory. We describe the procedure to configure preset upload and download directories in this section.

Before you configure the Connect:Direct nodes, add a new user named `cdadmin` on both SysA_CD and SysE_CD. The `cdadmin` user ID on both systems needs only basic user rights. The examples that follow use the `cdadmin` user ID.

To configure Sterling Connect:Direct on SysA:

1. On SysA, select **Start** → **All Programs** → **Sterling Commerce Connect:Direct v4.5.01** → **CD Requester**.
2. Expand the entry for **SysA_CD**, and double-click **Netmap** (Figure 5-5).

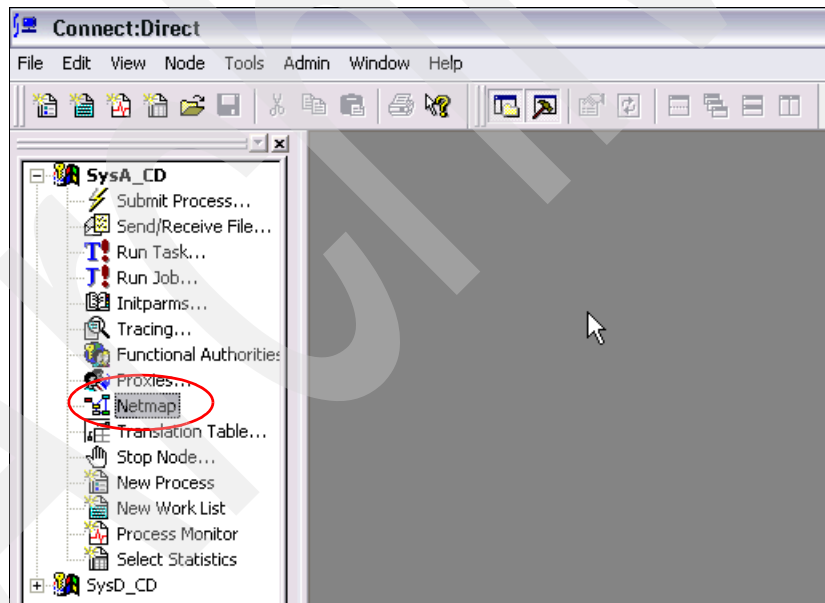


Figure 5-5 Launching the Connect:Direct requester netmap configuration

3. From the menu, select **Netmap** → **Insert** (Figure 5-6).

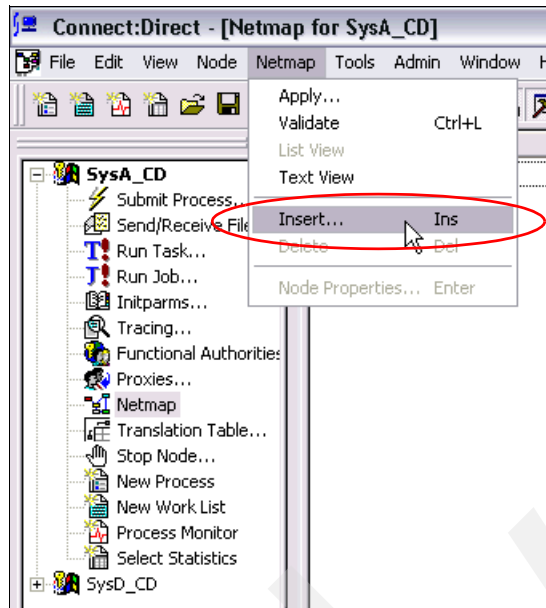


Figure 5-6 Inserting a new netmap entry

4. In the Netmap Node Properties window:
 - a. On the Main tab, complete the values shown in Figure 5-7.

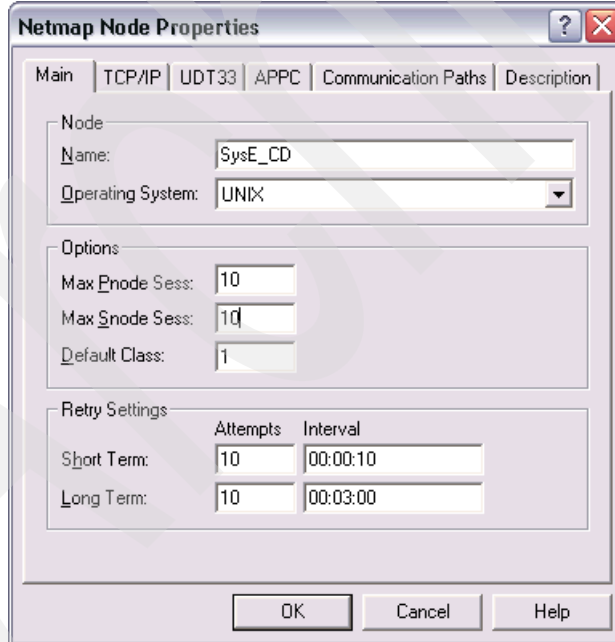


Figure 5-7 netmap Main tab

- b. On the TCP/IP tab, complete the values shown in Figure 5-8. Note that the configuration of the Connect:Direct nodes requires that you add the host address of the other system. However, in this scenario, the address of the other system actually points to the proxy instance on SysB. The host name of the proxy instance is *sysb*.

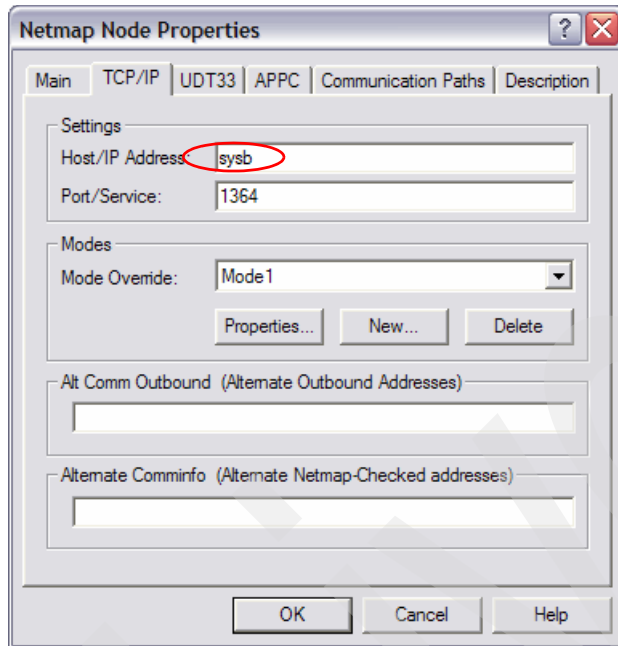


Figure 5-8 netmap TCP/IP tab

- c. On the Communications Paths tab, select **TCPCommPath**. Click the right arrow to add TCPCommPath to the Selected Paths box (Figure 5-9).

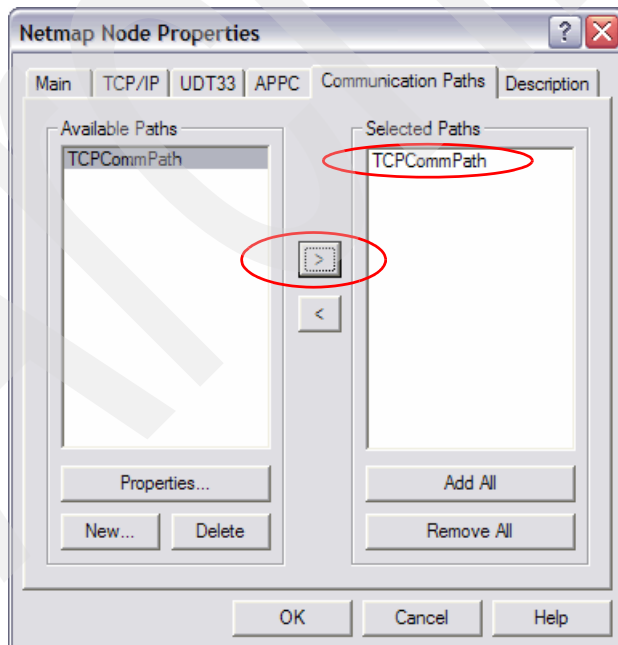


Figure 5-9 netmap Communication Paths tab

- d. Click **OK** to load the new entry.

5. Back at the main Netmap window, select **Netmap** → **Apply** (Figure 5-10).

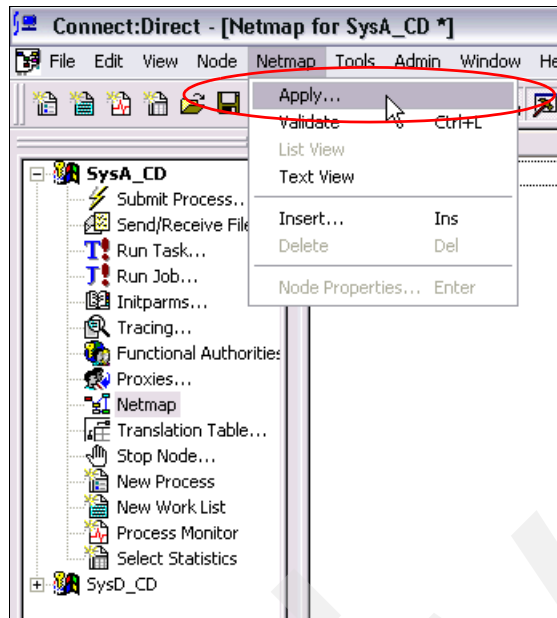


Figure 5-10 Applying a new netmap entry

6. If prompted to select a Connect:Direct node, select **SysA_CD** (Figure 5-11).

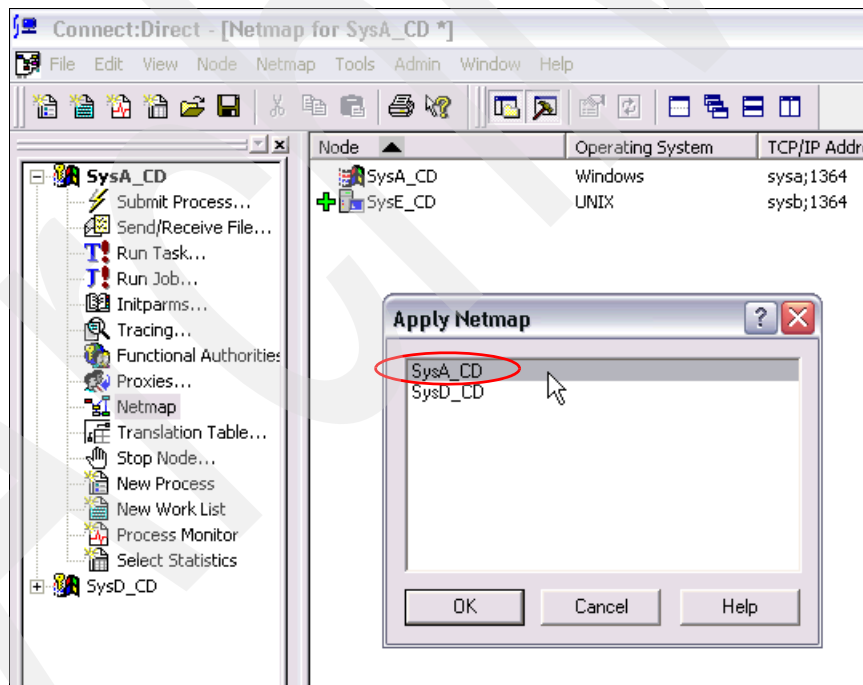


Figure 5-11 Applying a netmap entry to multiple nodes

7. From the menu select **Netmap** → **Insert** (Figure 5-12).

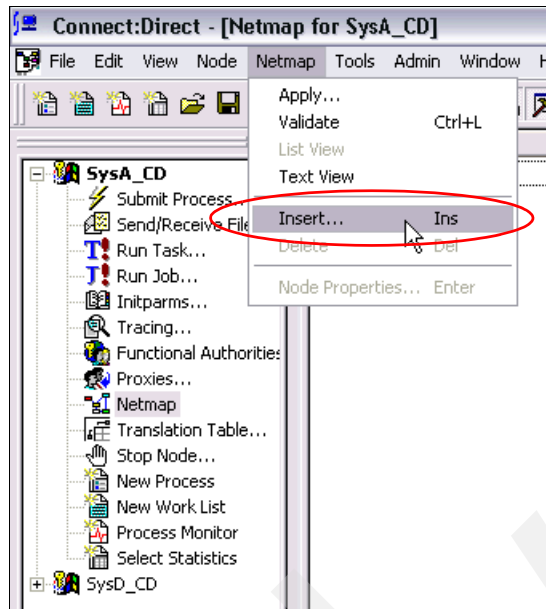


Figure 5-12 Inserting a new netmap entry

8. In the Netmap Node Properties window:
 - a. On the Main tab, complete the values as shown in Figure 5-13.

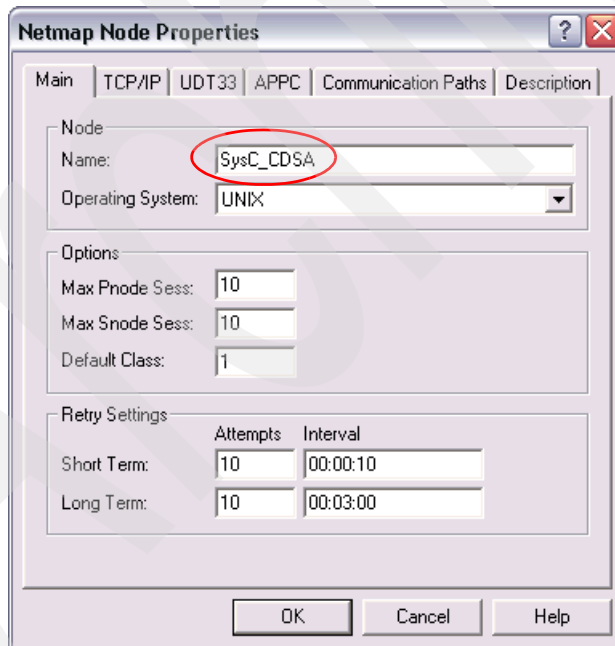


Figure 5-13 netmap Main tab

- b. On the TCP/IP tab, complete the values as shown in Figure 5-14.

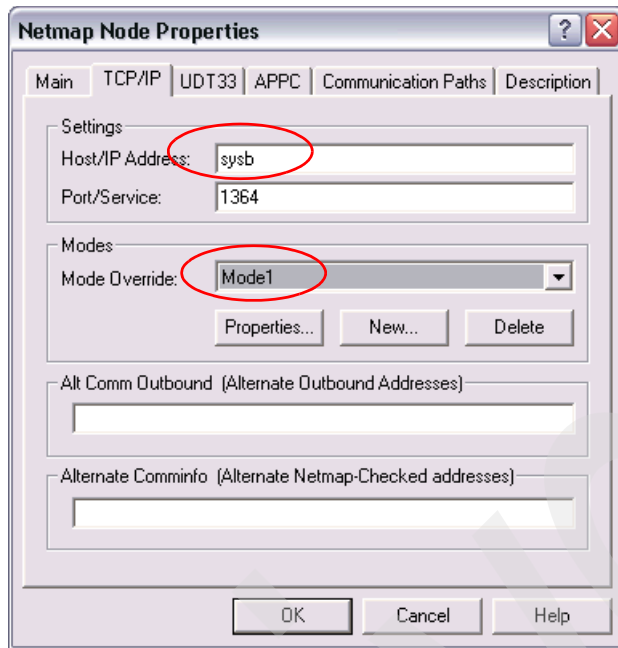


Figure 5-14 netmap TCP/IP tab

- c. On the Communications Paths tab, select **TCPCommPath**. Click the right arrow to add TCPCommPath to the Selected Path box (Figure 5-15).

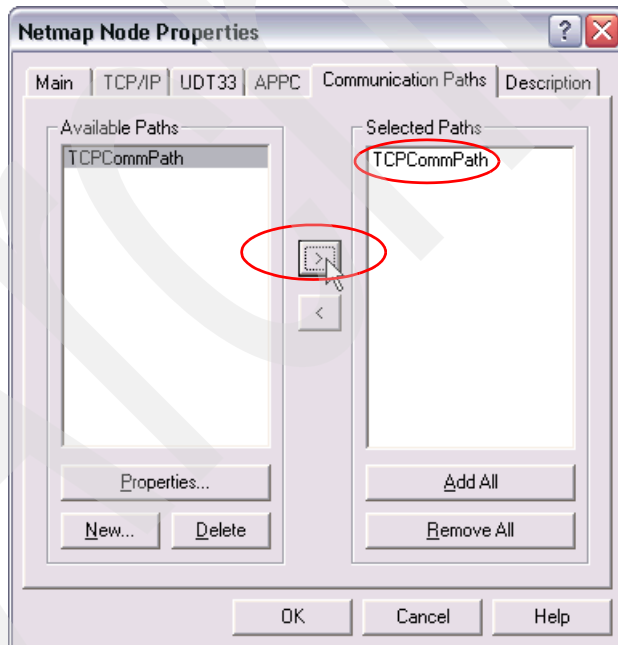


Figure 5-15 netmap Communications Paths tab

- d. Click **OK** to stage the new entry.

9. Back on the main Netmap window, select **Netmap** → **Apply** (Figure 5-16).

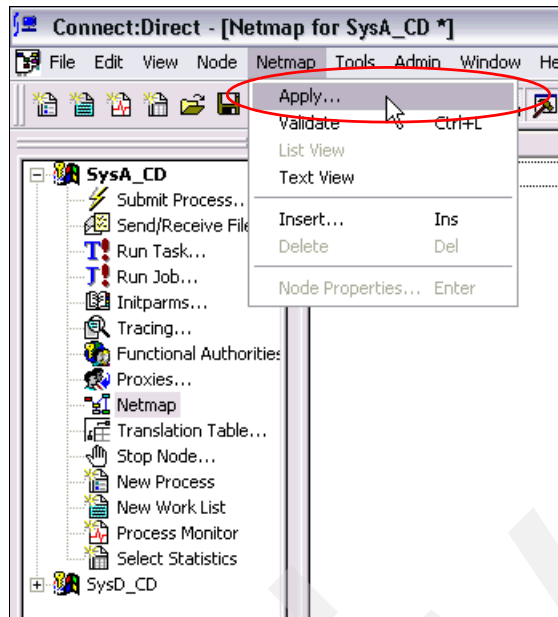


Figure 5-16 Applying a new netmap entry

10. If prompted to select a Connect:Direct node, select **SysA_CD** (Figure 5-17).

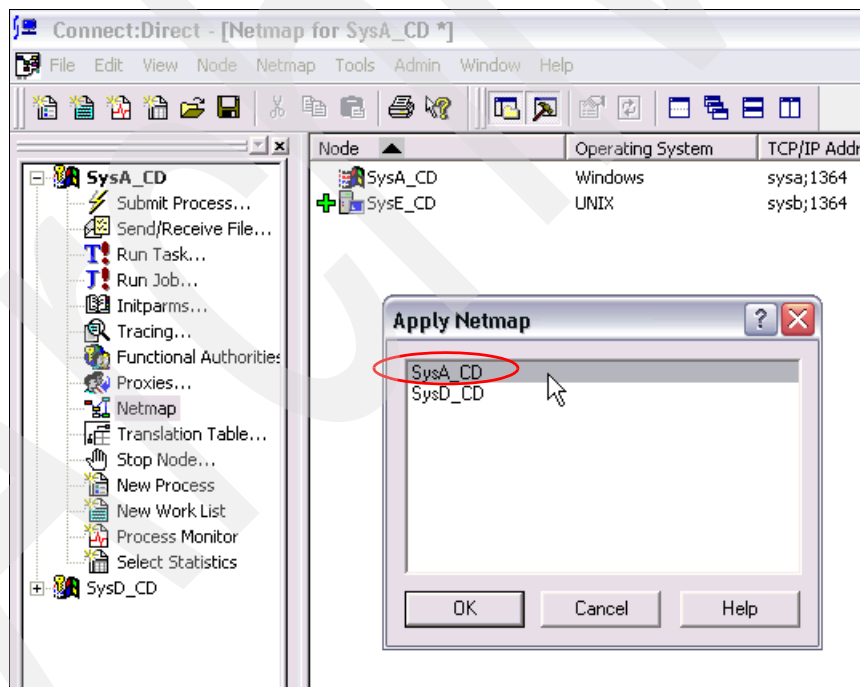


Figure 5-17 Applying a netmap entry to multiple nodes

11. Create the following directories on SysA_CD where Sterling Connect:Direct for Microsoft Windows is installed. Ensure that the cdadmin user ID has operating system permissions to read and write to these directories.

- C:\CDWindows_files\upload
- C:\CDWindows_files\download

12. Expand the entry for **SysA_CD**, and double-click **Functional Authorities** (Figure 5-18).

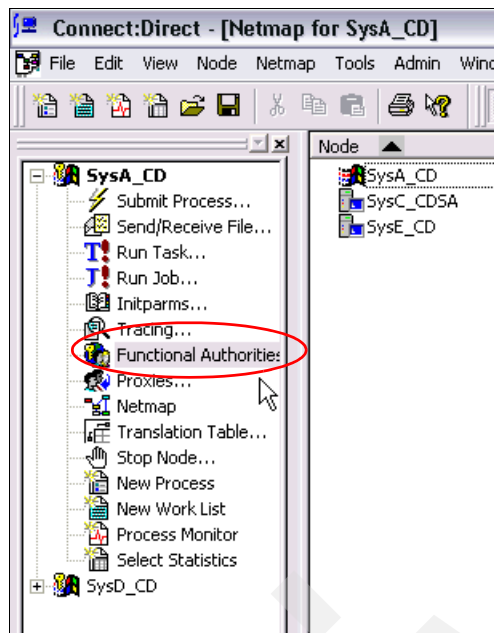


Figure 5-18 Launching the Connect:Direct requester Functional Authorities configuration

13. In the Functional Authorities dialog box, click **New Admin** (Figure 5-19).

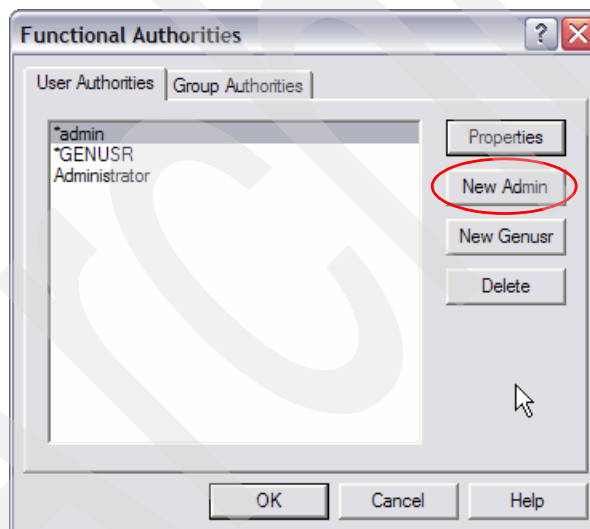


Figure 5-19 Functional Authorities User Authorities tab

14. In the Adding New User window:

- a. On the Main tab, enter `cdadmin` in the Name field (Figure 5-20).

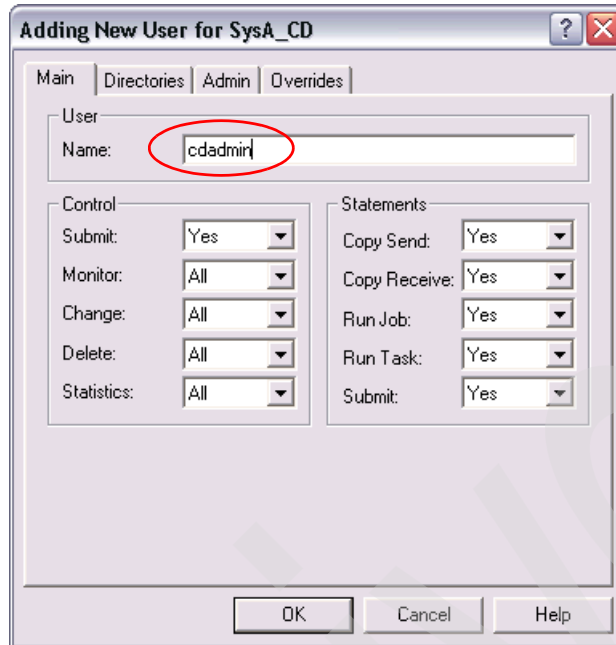


Figure 5-20 Functional Authorities adding a new user Main tab

- b. On the Directories tab, enter the upload and download directories that you created on SysA_CD in step 11 on page 110 (Figure 5-21).

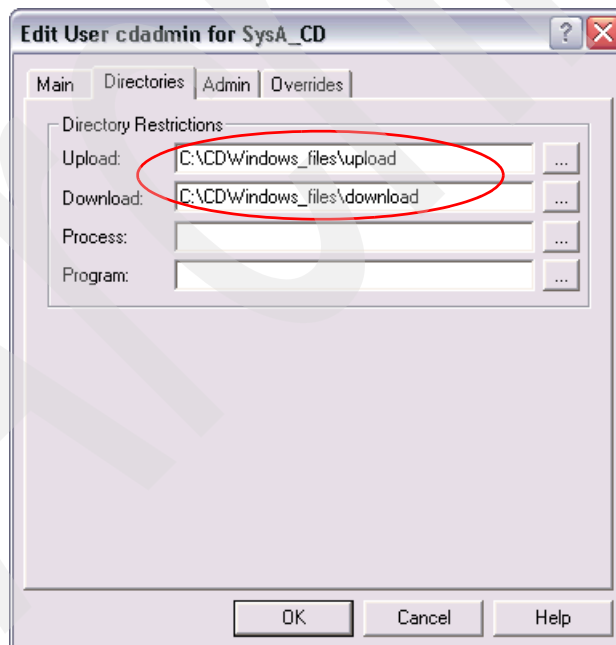


Figure 5-21 Functional Authorities adding a new user Directories tab

- c. Click **OK** to save your changes.

15. Expand the entry for **SysA_CD**, and double-click **Proxies** (Figure 5-22).

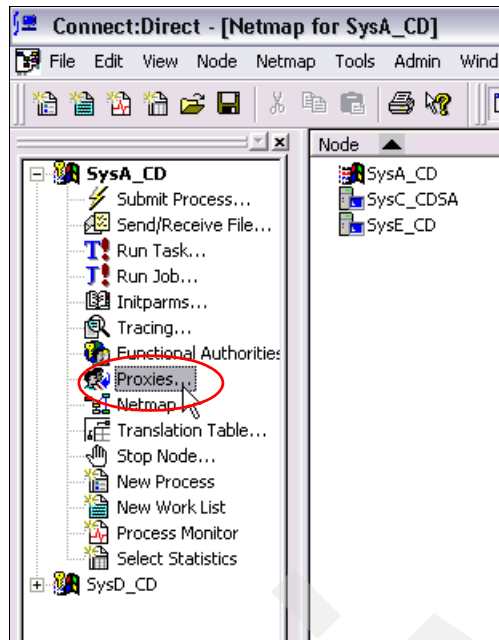


Figure 5-22 Launching the Connect:Direct requester Proxies configuration

16. Click **Insert** (Figure 5-23).

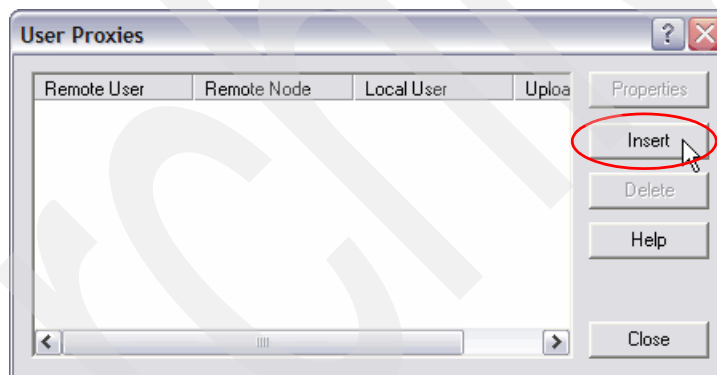


Figure 5-23 User Proxies dialog box

17. In the New User Proxy window:

- a. On the Main tab, enter the text as shown in Figure 5-24. All fields must be populated.

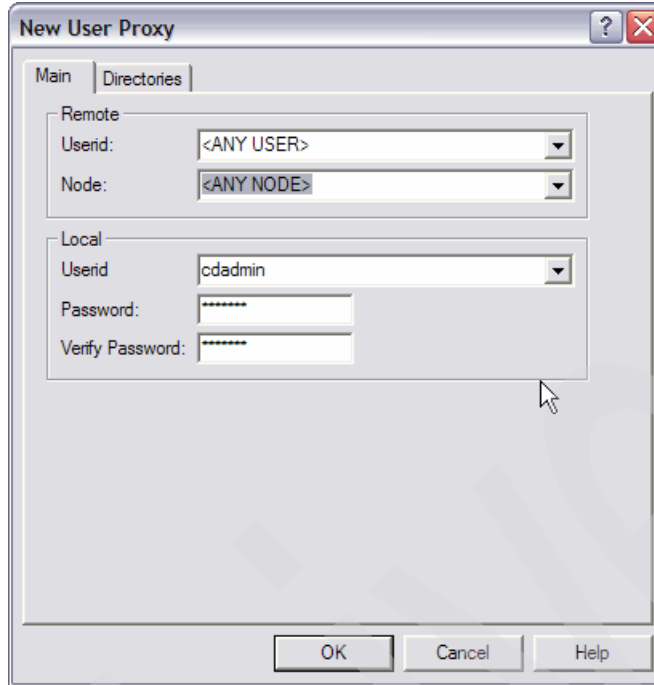


Figure 5-24 New User Proxy Main tab

- b. On the Directories tab, change the copy send and copy receive permissions to **Yes** (Figure 5-25). Enter the upload and download directories that you created on SysA_CD in step 11 on page 110.

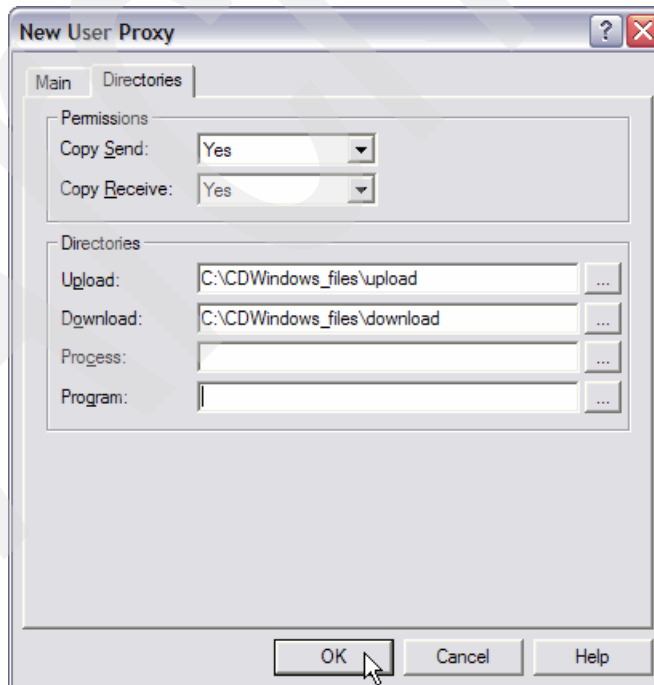


Figure 5-25 New User Proxy Directories tab

5.2.11 Configuring the Connect:Direct for Linux node SysE_CD

To configure the Sterling Connect:Direct for Linux node SysE_CD:

1. Append the netmap entries shown in Example 5-1 to the `/opt/cdunix/ndm/cfg/SysE_CD/netmap.cfg` file.

Example 5-1 Netmap entry on SysE_CD that points to SysC_CDSA

```
SysC_CDSA:\
:conn.retry.stwait=00.00.30:\
:conn.retry.stattempts=3:\
:conn.retry.ltwait=00.10.00:\
:conn.retry.ltattempts=6:\
:tcp.max.time.to.wait=180:\
:runstep.max.time.to.wait=0:\
:contact.name=:\
:contact.phone=:\
:descrip=:\
:sess.total=255:\
:sess.pnode.max=255:\
:sess.snode.max=255:\
:sess.default=1:\
:comm.info=sysc;1364:\
:comm.transport=tcp:\
:comm.bufsize=65536:\
:pacing.send.delay=0:\
:pacing.send.count=0:
```

2. Append the userfile entry shown in Example 5-2 to the `/opt/cdunix/ndm/cfg/SysE_CD/userfile.cfg` file. Notice that an upload and download directory is applied.

Example 5-2 Userfile entry on SysE_CD for inbound connections by the cdadmin user ID

```
*@*:\
:local.id=cdadmin:\
:pstmt.upload=y:\
:pstmt.upload_dir=:\
:pstmt.download=y:\
:pstmt.download_dir=:\
:pstmt.run_dir=:\
:pstmt.submit_dir=:\
:descrip=

cdadmin:\
:admin.auth=y:\
:pstmt.copy.ulimit=y:\
:pstmt.upload=y:\
:pstmt.upload_dir=/opt/cdunix/upload:\
:pstmt.download=y:\
:pstmt.download_dir=/opt/cdunix/download:\
:pstmt.run_dir=:\
:pstmt.submit_dir=:\
:name=:\
:phone=:\
:descrip=
```

5.2.12 Installing and configuring the Connect:Direct server adapter

For Sterling Connect:Direct file transfers through Sterling B2B Integrator and the Sterling File Gateway, you need to install and configure a Connect:Direct server adapter (CDSA) in Sterling B2B Integrator. This adapter enables Sterling B2B Integrator to behave like a Connect:Direct node, which can then initiate and receive file transfers as though it were a standard PNODE or SNODE.

Starting Sterling B2B Integrator and logging in to the console

To configure the Connect:Direct server adapter, follow these steps:

1. Start Sterling B2B Integrator and Sterling File Gateway if it is not already running by double-clicking the **Sterling_Integrator_at_8080** desktop icon.

2. Start Internet Explorer and go to:

`http://<servername>:<port>/filegateway/`

Where `<servername>` and `<port>` are the server and port that are defined for your configuration.

3. Log in using the Sterling File Gateway administrator user ID and password (Figure 5-26). The default administrator user ID is `fg_sysadmin`.



Figure 5-26 Sterling File Gateway login window

4. In Sterling File Gateway, go to **Tools** → **B2B Console** (Figure 5-27) to open the Sterling B2B Integrator console in a new browser window.



Figure 5-27 Open the Sterling B2B Integrator browser window

Creating the Sterling Connect:Direct netmap in Sterling B2B Integrator

To create the Sterling Connect:Direct netmap in Sterling B2B Integrator:

1. Define the Connect:Direct nodes that you connect within your network. Go to **Deployment** → **Adapter Utilities** → **C:D Netmaps** → **C:D Nodes**. Then create a new node by clicking **Go** to the right of New Node (Figure 5-28).

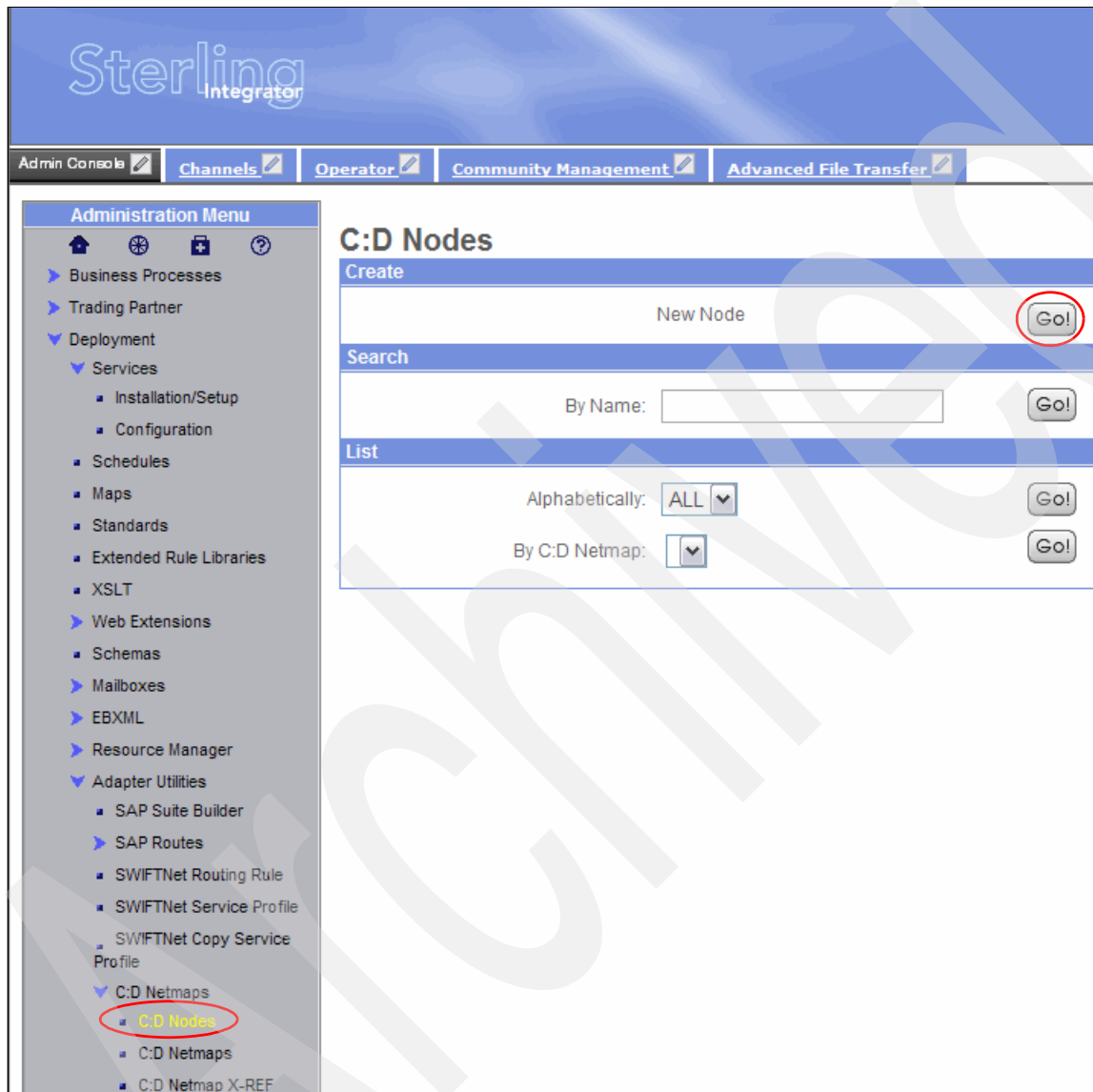


Figure 5-28 Create new Connect:Direct node

2. Enter values for the Connect:Direct node to which you will be connecting in your internal trusted network (SysE in our environment) (Figure 5-29). The Connect:Direct Server Node name on SysE is set up as SysE_CD and is set up to use the default port of 1364.

C:D Nodes

Create Node: Information

Connect:Direct Server Node Name:

Connect:Direct Server Host:

Connect:Direct Server Port:

Max locally initiated (pnode) sessions allowed:

Max remotely initiated (snode) sessions allowed:

Alternate Comm Info:

Secure+ Option

Disabled

Enabled

< Back Next > Cancel Save

Figure 5-29 New Connect:Direct node input panel

3. Click **Next**, and then click **Finish** on the confirmation panel.
4. Repeat the creation of a C:D node, but this time enter the details to connect to the Connect:Direct node in the external (unprotected) network using a proxy server. Enter the values shown in Table 5-2. Note that you do not enter the host name of the external Connect:Direct node itself (which is SysA in our environment), but instead enter the host name of the proxy server node (SysB). The proxy is configured to route the requests to the intended recipient.

Table 5-2 Values for Connect:Direct node SysA_CD on external network using the proxy

Parameter name	Value
Connect:Direct Server Node Name	SysA_CD
Connect:Direct Server Host	<hostname_of_proxy> In our scenario, this host name is SysB.
Connect:Direct Server Port	1364

5. Create the netmap of Connect:Direct nodes in your network. Go to **Deployment** → **Adapter Utilities** → **C:D Netmaps** → **C:D Netmaps**. Then create a new netmap by clicking **Go** to the right of New Netmap (Figure 5-30).

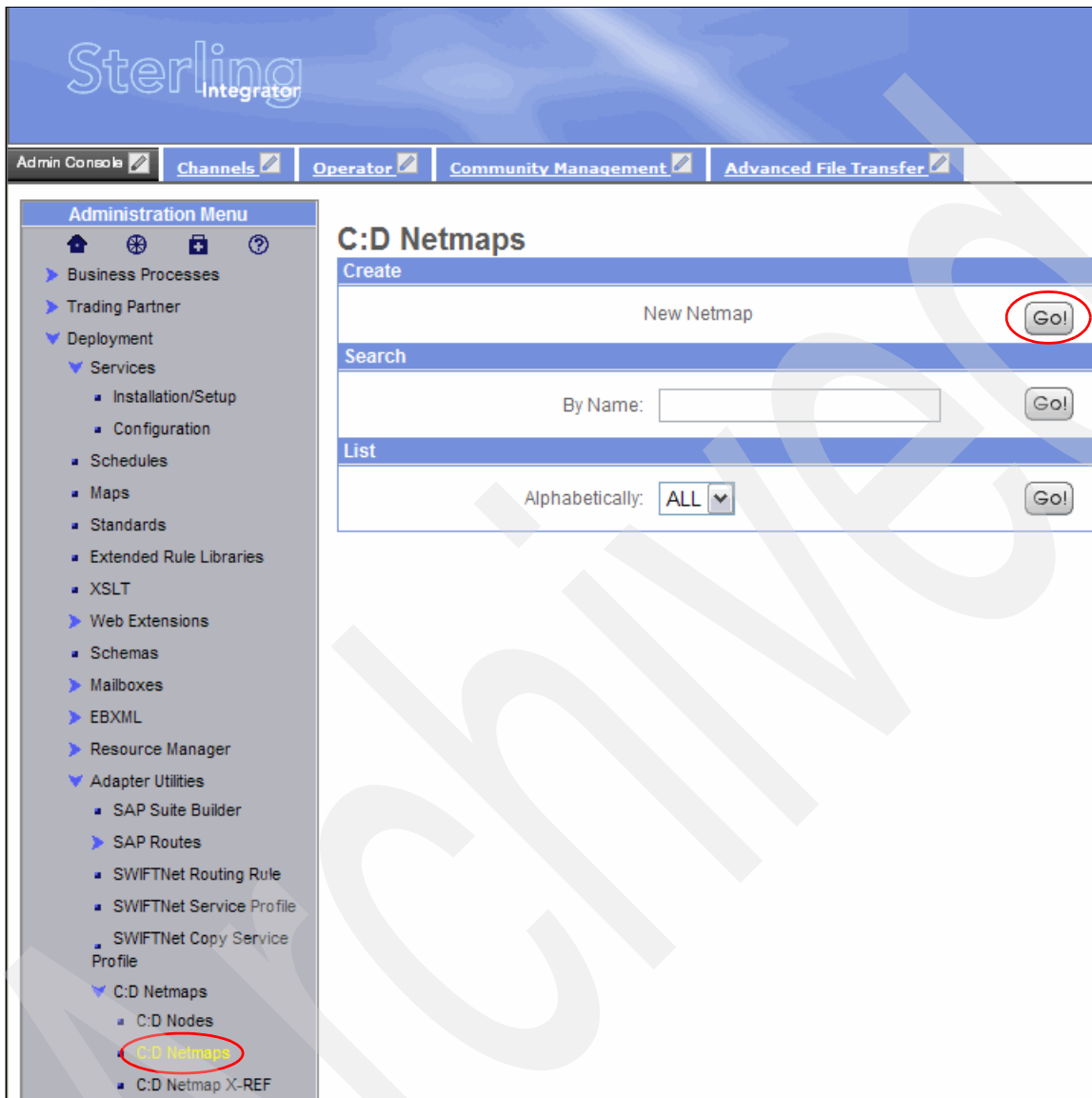
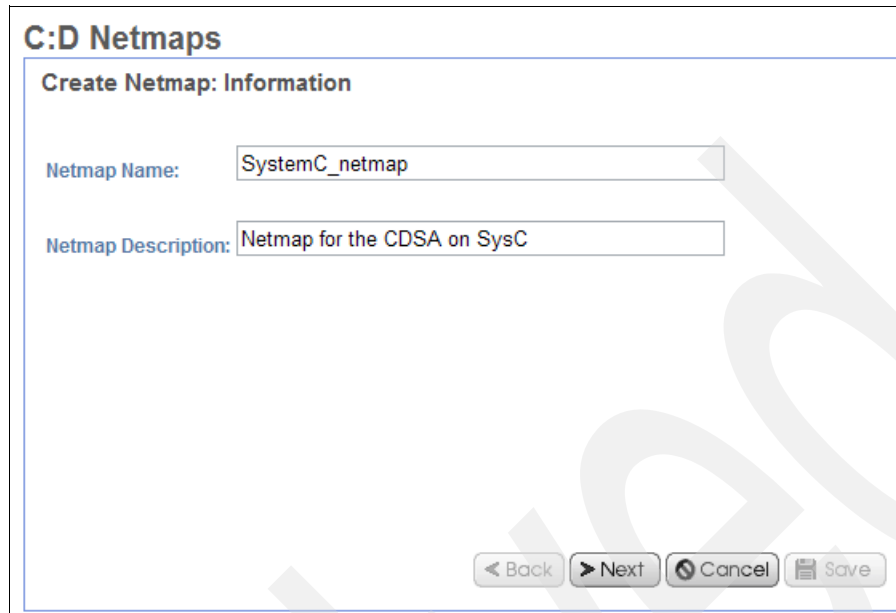


Figure 5-30 Create new netmap

6. Enter the values shown in Figure 5-31, and click **Next**.



C:D Netmaps

Create Netmap: Information

Netmap Name:

Netmap Description:

< Back > Next Cancel Save

Figure 5-31 New netmap input panel

7. Click **Next**, and then click **Finish** on the confirmation panel.

8. Populate the newly created SystemC_netmap with the Connect:Direct nodes in your network. Go to **Deployment** → **Adapter Utilities** → **C:D Netmaps** → **C:D Netmap X-REF**. Create the cross-references by clicking **Go** to the right of **Netmap Cross Reference** (Figure 5-32).

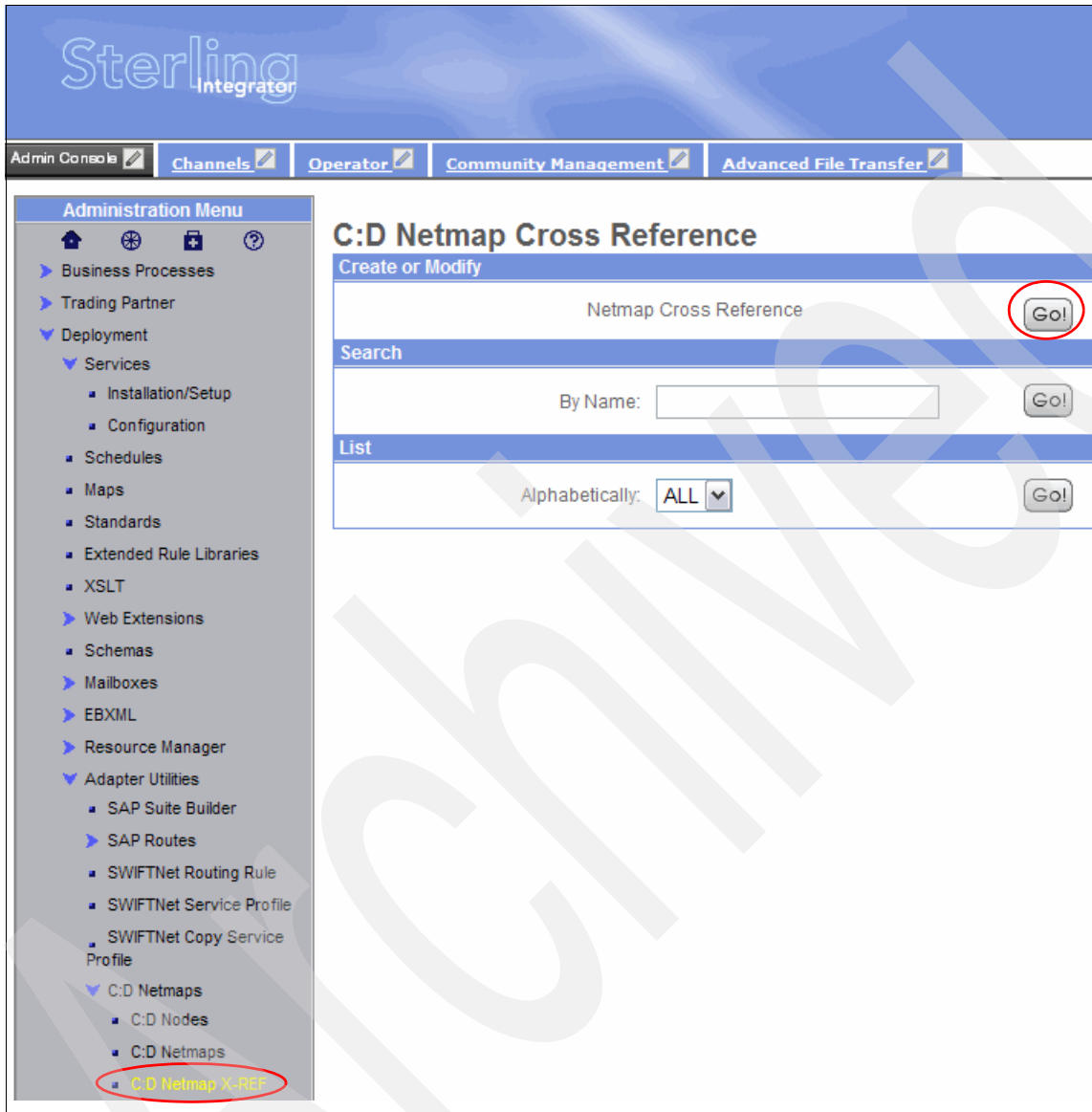


Figure 5-32 Populate the netmap

9. Select **SystemC_netmap** as the C:D Netmap Name, and click **Next** (Figure 5-32 on page 122).

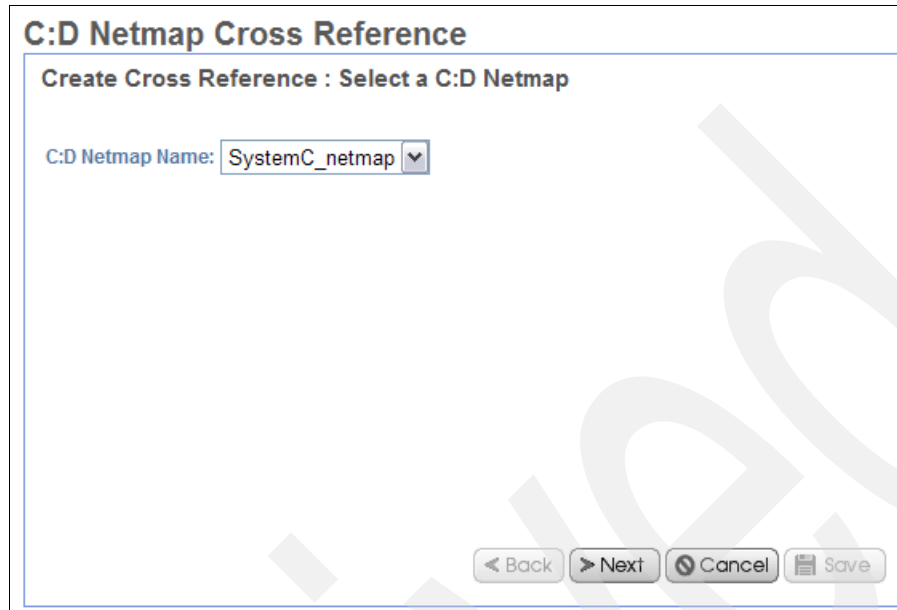


Figure 5-33 netmap name selection panel

10. Click **add** (Figure 5-34).

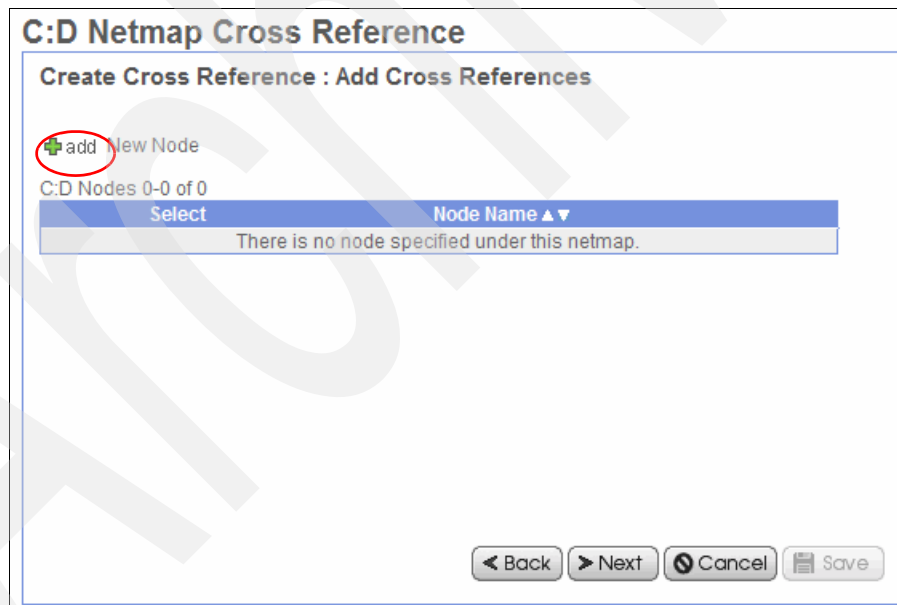


Figure 5-34 Adding a cross-reference

11. Select both Connect:Direct nodes, and click **Save** (Figure 5-35).

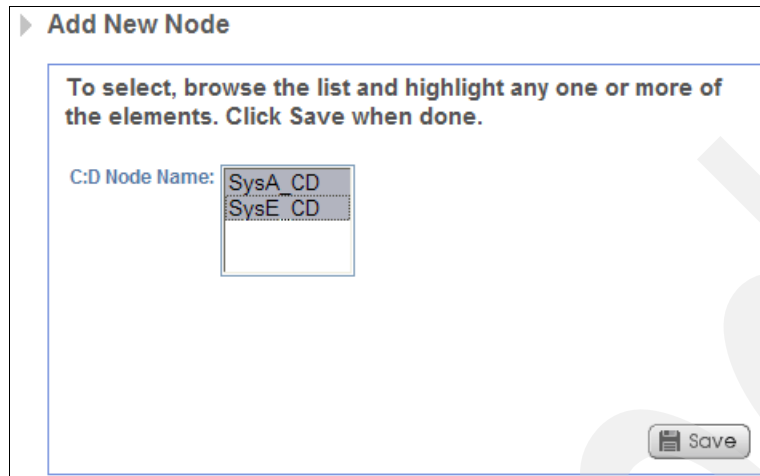


Figure 5-35 Select nodes to add to the netmap

12. Click **Next**, and then click **Finish** on the confirmation panel (Figure 5-36). Your netmap is now created, and you can create the Connect:Direct server adapter that uses it.

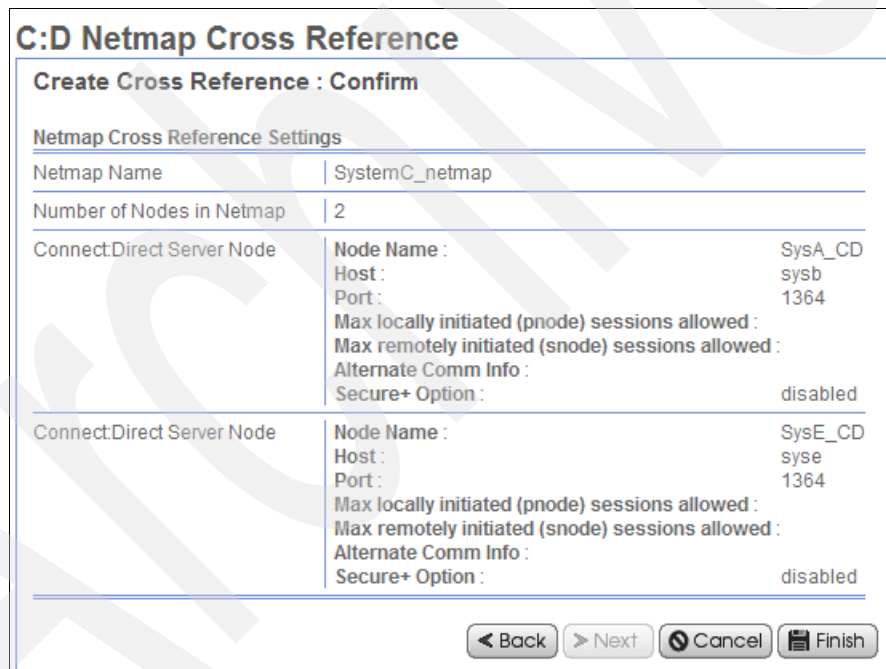


Figure 5-36 Netmap cross reference confirmation panel

The netmap is accessible by the Connect:Direct server adapter after it is installed and configured. The netmap contains the names and connection details of Connect:Direct nodes to which the Connect:Direct server adapter can receive and send messages. SysA_CD is the external partner's Connect:Direct node from which a file is received in this scenario. Note that the host is defined as SysB, which is the host name of the proxy system that forwards the messages. SysE_CD is the internal partner's Connect:Direct node to which the message is sent.

Installing the Connect:Direct server adapter

To install the Connect:Direct server adapter:

1. On the Sterling B2B Integrator dashboard, click **Deployment** → **Services** → **Configuration** (Figure 5-37). Click **Go** to the right of New Service.

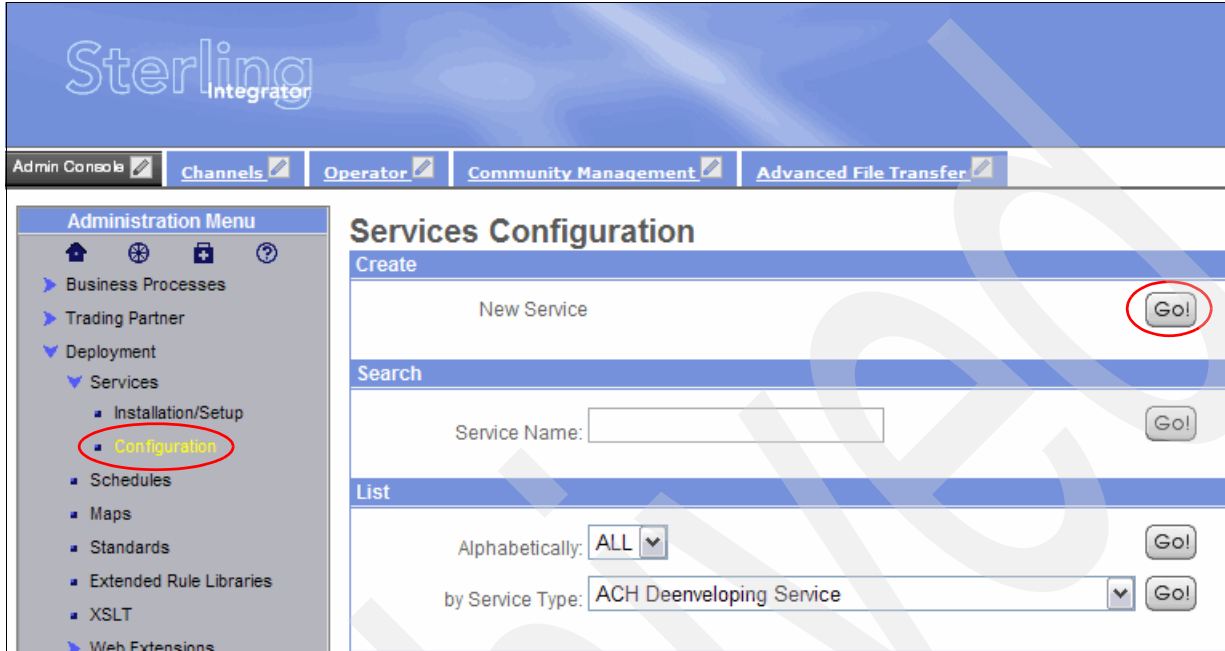


Figure 5-37 Create a new service configuration

2. Click the file-tree icon (Figure 5-38).

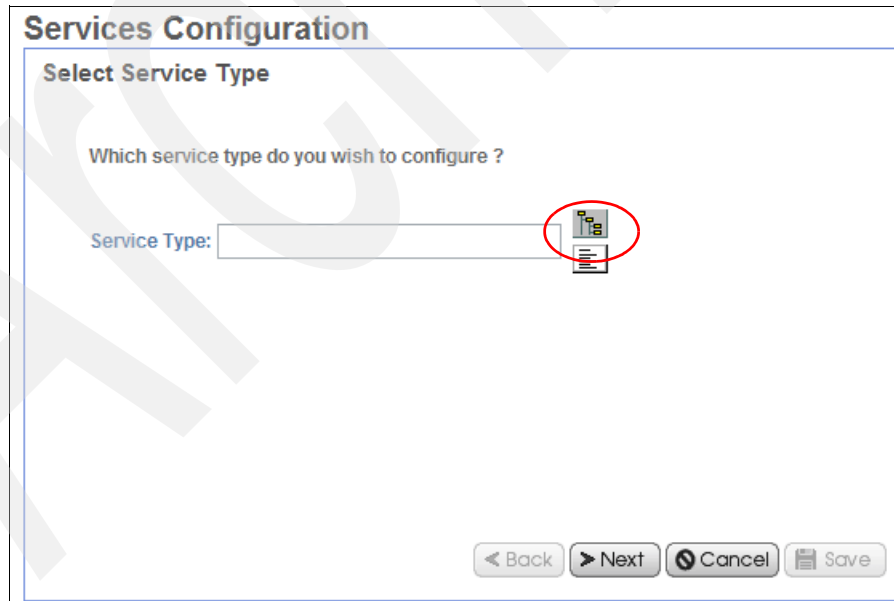


Figure 5-38 Select the service type to create

- Expand **Applications** → **Sterling Commerce** → **Connect:Direct**, and select **Connect:Direct Server Adapter** (Figure 5-39). Click **Save**, and then click **Next**.

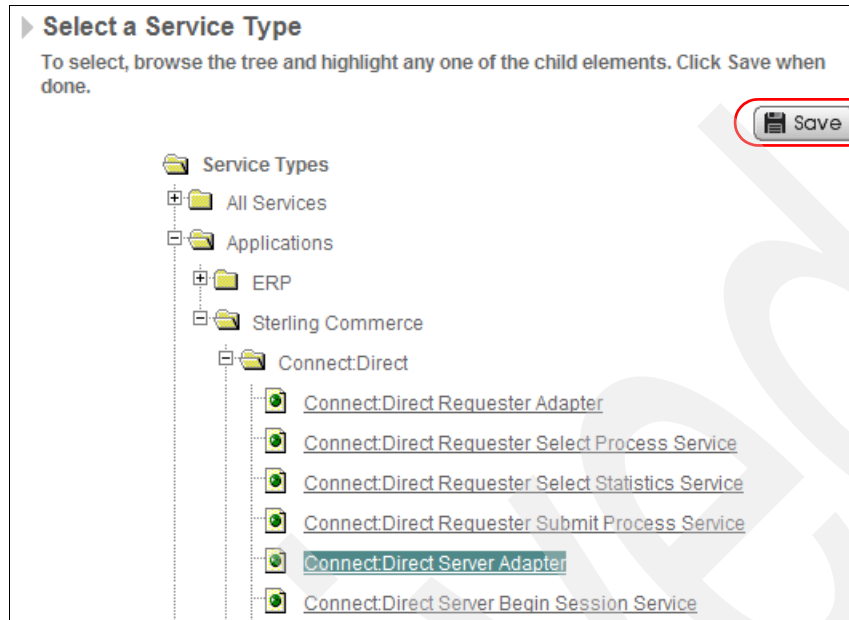


Figure 5-39 Select Connect:Direct server adapter

- Enter a name of SysC_CDSA and a description of C:D Server Adapter on SysC (Figure 5-40). Click **Next**.

The screenshot shows the 'Services Configuration' dialog box for 'Connect:Direct Server Adapter: Name'. The 'Name' field contains 'SysC_CDSA' and the 'Description' field contains 'C:D Server Adapter on SysC'. The 'Select a group' section has 'None' selected. The 'Create New Group' and 'Select Group' options are also visible. At the bottom, there are buttons for '< Back', '> Next', 'Cancel', and 'Save'.

Figure 5-40 Setting the name of the adapter in Sterling B2B Integrator

5. Enter the Connect:Direct Server Node Name as SysC_CDSA, and click **Next** (accepting all other default values) (Figure 5-41).

Services Configuration

SysC_CDSA: InitParms

Connect:Direct Server Node Name: SysC_CDSA

Connect:Direct Perimeter Services Option: node1 & local

Connect:Direct Server Port: 1364

Firewall Ports:

Max locally initiated (pnode) sessions allowed: 5

Max remotely initiated (snode) sessions allowed: 5

Document Storage

System Default

Database

File System

NetMap Check

Check both node name and IP address

Check node name only

No

Buffer-size for Copy: 32768

Number of short-term session retry attempts: 0

Interval between short-term session attempts (seconds): 5

Number of long-term session retry attempts: 0

Interval between long-term session attempts (minutes): 1

Retry Remote File Allocation Errors

Max Session Establishment Timeout value in Seconds: 600

Max Socket Read Timeout value in Seconds: 90

Server Start Option: Warm

< Back Next > Cancel Save

Figure 5-41 Values for the Connect:Direct server adapter

- Keep clicking **Next**, accepting all default values, until you reach the SysC_CDSA: Netmap panel. Select the netmap name **SystemC_netmap** (Figure 5-42) to define the netmap that the new Connect:Direct server adapter will use. Click **Next**.

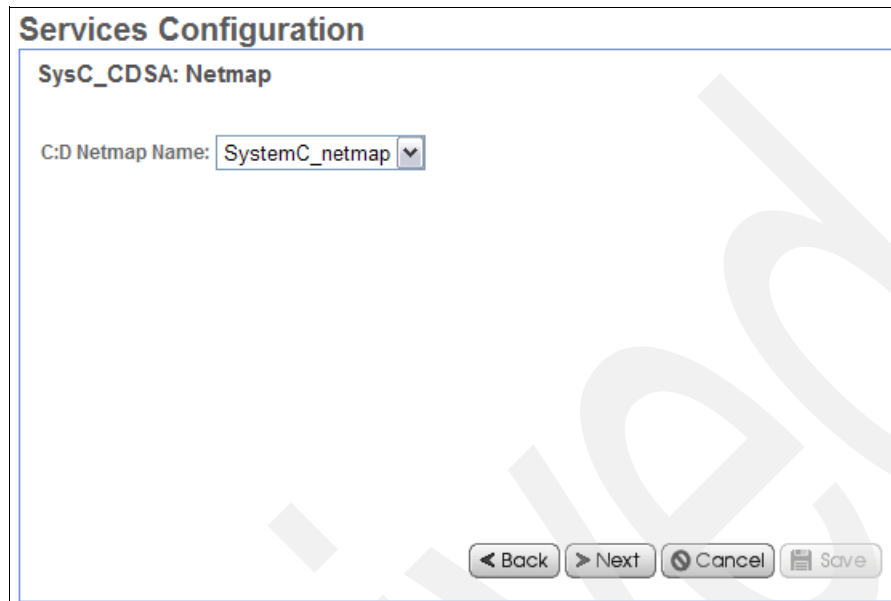


Figure 5-42 Select netmap for the Connect:Direct server adapter

- On the confirmation panel, scroll down to see that the netmap details are configured for the Connect:Direct server adapter (Figure 5-43). Click **Finish**.

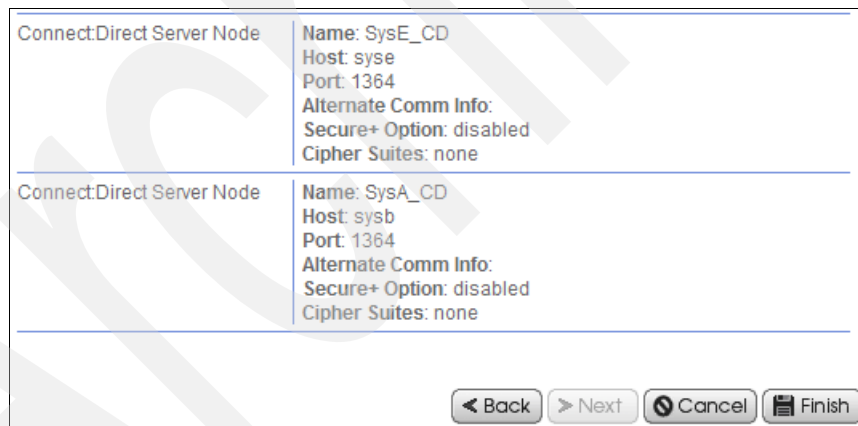


Figure 5-43 Node details defined in the configured netmap for the CDSA

5.2.13 Configuring the proxy

The netmap on Sterling B2B Integrator is already configured to point to the proxy node on SysB instead of going directly to SysA. The Sterling Connect:Direct netmap for SysA_CD on SysA is also configured to point to SysC_CDSA on the SysB node instead of going directly to SysC.

You need to configure the chosen proxy server that you use in your scenario to route requests from the Connect:Direct node SysA_CD on SysA to the Connect:Direct server adapter SysC_CDSA on SysC and vice-versa.

5.2.14 Configuring Sterling File Gateway

This section describes setting up Sterling File Gateway with a routing channel to automatically route requests from the external Connect:Direct node on SysA through to the internal Connect:Direct node on SysE and vice versa. Sterling File Gateway uses the concept of *partners*. Files are transferred between partners. This scenario includes two partners between that files are transferred:

- ▶ SysA_CD_Partner is the external partner that, in this scenario, communicates using Sterling Connect:Direct.
- ▶ SysE_Partner transfers files using Sterling Connect:Direct.

For simplicity, we named the partners based on the systems that are used for the file transfer and the transport protocol that is used. So, the external partner always communicates from SysA. In this scenario, because the transport protocol is Sterling Connect:Direct, the partner name used is SysA_CD_Partner.

SysE is a dedicated internal Connect:Direct node and only communicates with Sterling File Gateway using that protocol. The partner name used is SysE_Partner.

Launching Sterling File Gateway

To log in to the Sterling File Gateway user interface:

1. Sterling B2B Integrator, and thus Sterling File Gateway, should already be started, as described in “Starting Sterling B2B Integrator and logging in to the console” on page 116. If this is not the case and if your server is stopped, start Sterling B2B Integrator by double-clicking the desktop icon **Sterling_Integrator_at_8080**.
2. Start Internet Explorer and go to:

`http://<servername>:<port>/filegateway/`

Where `<servername>` and `<port>` are the server and port that are defined for your configuration.

3. Log in (Figure 5-44). The default admin user ID for Sterling File Gateway is fg_sysadmin.

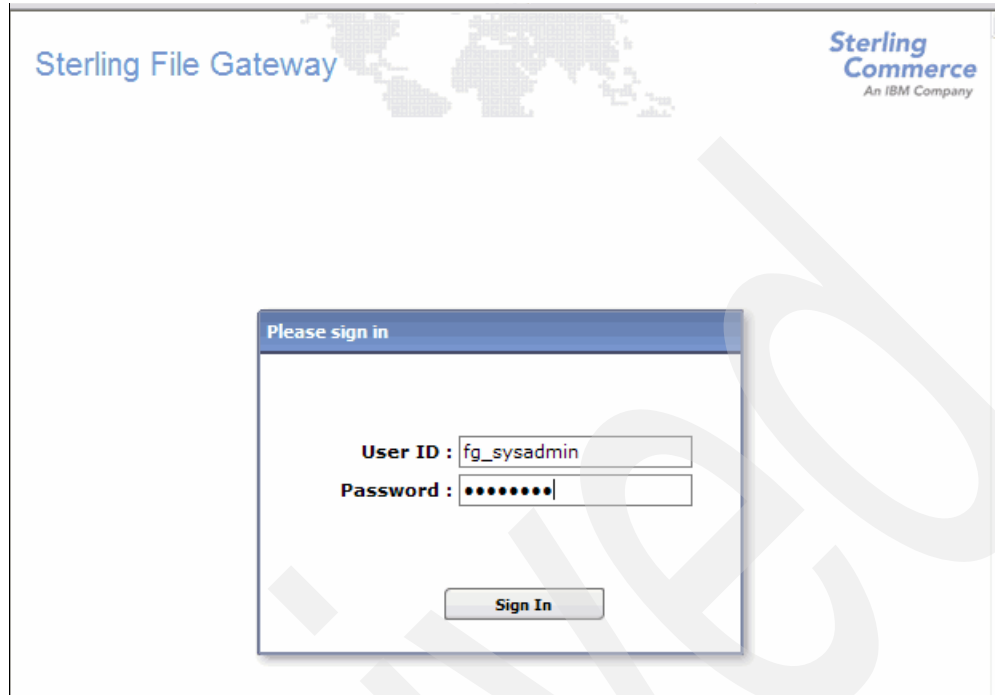


Figure 5-44 Login window for Sterling File Gateway

The main panel for Sterling File Gateway opens (Figure 5-45).

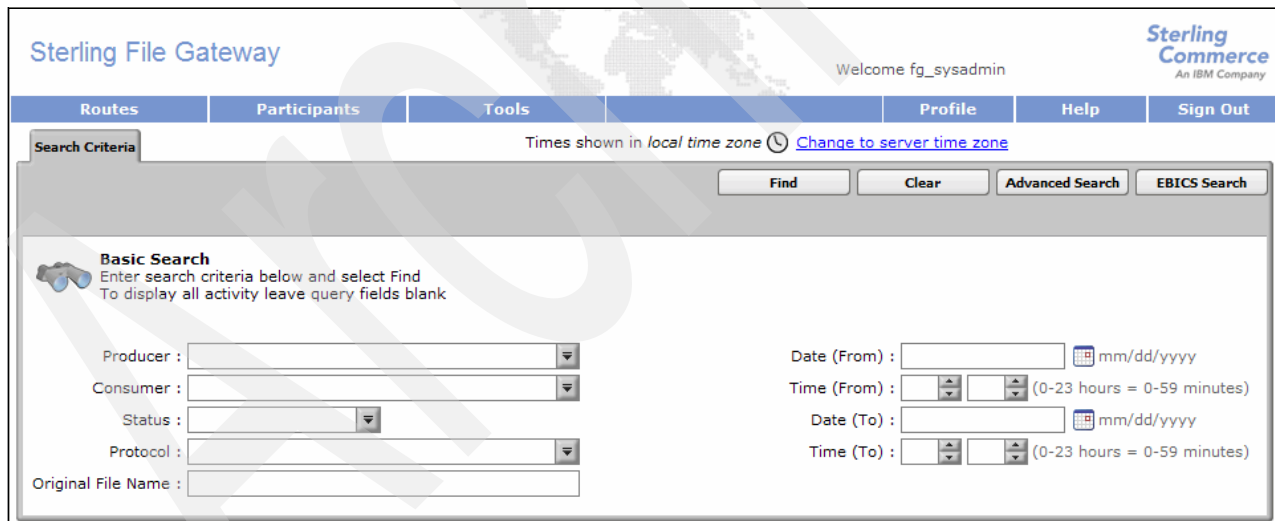


Figure 5-45 First panel when logged in to Sterling File Gateway

Creating a community

A *community* defines the protocols that partners within this community can use. To create a community:

1. Select **Participants** → **Communities** (Figure 5-46).

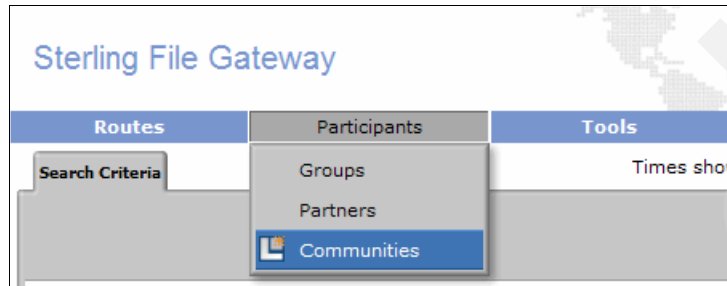


Figure 5-46 Create a new community

2. Select **add** (Figure 5-47).

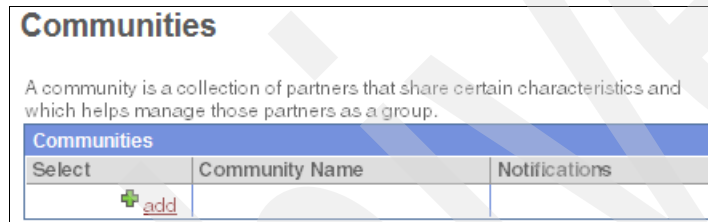


Figure 5-47 Adding a community

3. Enter a name for your community, such as FirstCommunity (Figure 5-48), and click **Next**. For simplicity, we create only one community for this book and enable all protocols within that community.

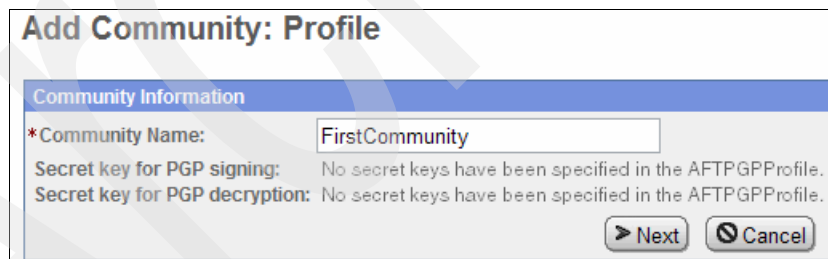


Figure 5-48 Name the community

4. Select all available check boxes to allow all partners in this community to communicate using all available protocols (FTP or FTPS, Sterling Connect:Direct, or SSH/SFTP) (Figure 5-49). Click **Next**.



Figure 5-49 Select all protocols

5. On the Notifications panel, accept the default setting of **No**, and click **Next** (Figure 5-50).

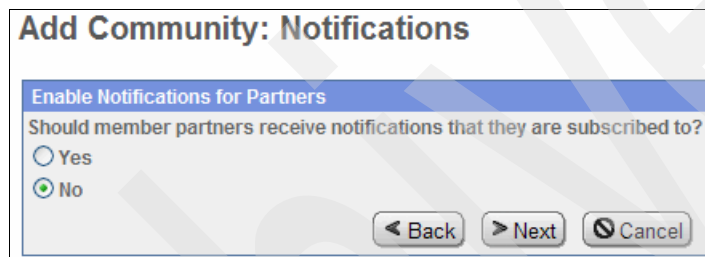


Figure 5-50 Accept the default setting for notifications

6. On the confirmation panel, click **Finish** (Figure 5-51). Then close the window and go back to the main Sterling File Gateway panel.



Figure 5-51 Add community confirmation panel

Creating partners

You now create partners to represent the organizations or departments that own the Connect:Direct nodes between which you will be sending files. One of the partners is the internal Connect:Direct node SysE_CD that is running on SysE, which is owned by

Company B. SysE is behind the firewall and belongs to the same internal network as Sterling B2B Integrator and Sterling File Gateway.

The other partner is an external partner (Company A) that owns the Connect:Direct node SysA_CD on external SysA.

To create partners:

1. Go to **Participants** → **Partners** (Figure 5-52).



Figure 5-52 Create partners

2. Click **Create** (Figure 5-53).

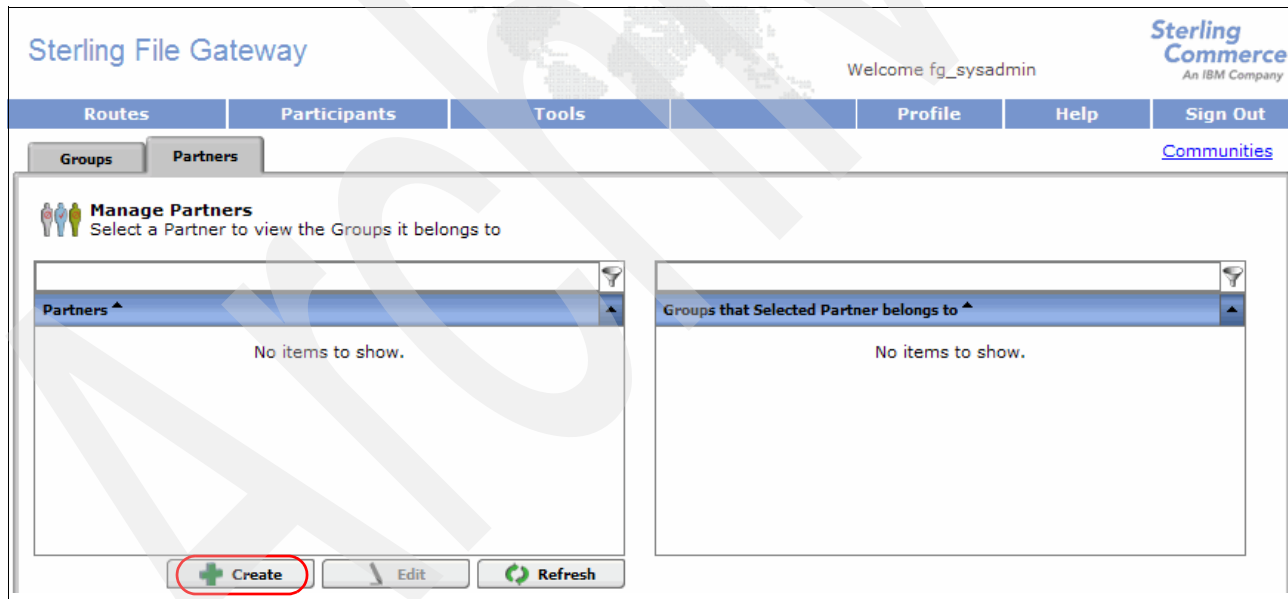


Figure 5-53 Create a new partner

3. Select **FirstCommunity**, and then click **Next** (Figure 5-54).



Figure 5-54 Select the community of which the new partner is a member

4. Create the SysA partner, who in this case communicates using Sterling Connect:Direct into Sterling File Gateway. Use the name SysA_CD_Partner. The Phone and Email Address fields are mandatory even though we have notifications disabled in this example. We used a false telephone number and email address for this example (Figure 5-55). Click **Next**.



Figure 5-55 Enter details for the SysA external partner

5. Enter a user name and password for the SysA external partner. These values are passed to Sterling File Gateway from Sterling Connect:Direct to place files into the SysA_CD_Partner mailbox. These same values are used if the SysA partner wants to use either the myFileGateway interface to access mailboxes or the FTP/SFTP to transfer files into mailboxes.

Use SysA_CD_Partner as the user name and its04you as the password. Because this is an example scenario, the values for the first and last name do not need to be real names (Figure 5-56). Click **Next**.

Password note: You cannot use the word *password* as the password with Connect:Direct nodes. The word *password* is considered a reserved keyword. File transfer scripts in Sterling Connect:Direct fail validation if you use the word *password* as the password.

Add Partner: User Account

User Account

Create a user account by filling the following information. This user account is to be used by SysA_CD_Partner when logging into myFileGateway.

*User Name: SysA_CD_Partner

*Password: ●●●●●●

*Confirm Password: ●●●●●●

*First Name: SysA_CD_Firstname

*Last Name: SysA_CD_Lastname

< Back > Next Cancel

Figure 5-56 Enter login details for the SysA external partner

6. On the Partner Role panel, select both the **Consumer of Data** and **Producer of Data** options because SysA_CD_Partner will put files into a mailbox on Sterling File Gateway (Producer of Data) and will also retrieve files that are placed into its mailbox (Consumer of Data). That is, this partner is used for both inbound and outbound file transfers.

Under the Consumer of Data section, select the **Partner will listen for a connection to consume data** option, which means that SysA_CD_Partner will listen on a designated mailbox for files to consume any files placed there (Figure 5-57). Click **Next**.

Add Partner: Partner Role

Select Partner Role

There are no consumers configured in the system right now. It is recommended that the first partner that is configured is one that is a consumer or both a consumer and a producer.

SysA_CD_Partner is a Consumer of Data

Partner will initiate the connection to consume data.

Partner will listen for a connection to consume data.

SysA_CD_Partner is a Producer of Data

< Back > Next Cancel

Figure 5-57 Partner role panel

- On the Initiate Connection Settings panel, accept the default answer of **No** (Figure 5-58). This partner communicates only using Sterling Connect:Direct and does not use SSH/SFTP or SSH/SCP protocols. Click **Next**.

Figure 5-58 Initiate Connections Settings panel

- On the Protocol panel, select **Listen for Connect:Direct Connections**, and then select **Next** (Figure 5-59).

Figure 5-59 Select the protocol

- On the Connect:Direct panel (Figure 5-60), enter the details for the local and remote Connect:Direct nodes, according to SysA_CD_Partner. In this case, the local Connect:Direct node name is the name of the Connect:Direct server adapter configured on Sterling B2B Integrator on SysC, which is SysC_CDSA. The remote Connect:Direct node name is SysA_CD. Enter the user ID and password for the remote Connect:Direct node on SysA, for example, cdadmin/cdadmin. Click **Next**.

Figure 5-60 Enter the node details for SysA_CD_Partner

10. On the PGP Settings panel, accept the defaults (both **No**) (Figure 5-61). Click **Next**.

PGP Settings

PGP Settings

When SysA_CD_Partner sends PGP packaged files
 ... the files are processed in accordance with the Routing Channels (and their templates) that SysA_CD_Partner is a producer for.

Note:
 *If SysA_CD_Partner sends data that is encrypted, it will be decrypted using the Router's secret PGP key.
 *If SysA_CD_Partner sends data that is signed, it will be verified using SysA_CD_Partner's public PGP key if that key is present in the public key ring.

Does SysA_CD_Partner require data to be signed by the Router ?
 Yes No

Does SysA_CD_Partner require data to be encrypted by the Router ?
 Yes No

< Back Next > Cancel

Figure 5-61 PGP Settings panel

11. The confirmation panel displays your choices (Figure 5-62). Click **Finish** to complete the creation of SysA_CD_Partner.

Add Trading Partner: Confirm

Profile	
Partner Name:	SysA_CD_Partner
Address:	,
Phone:	1234
Email Address:	sysacd@itso_redbooks.com
User Account	
User Name:	SysA_CD_Partner
Password:	*****
First Name:	SysA_CD_Firstname
Last Name:	SysA_CD_Lastname
Protocol	
Partner Role:	Producer of Data, Consumer of Data
Connection Direction:	Listen Connection
Transport Method:	Connect:Direct
• Local Node Name	SysC_CD_SA
• Remote Node Name	SysA_CD
• Remote User Id	cdadmin
• Remote Password	*****
• Remote File Name	
• Disposition to be applied to remote file	
• Sysopts string to be applied to remote file	
• CheckpointInterval to be used for checkpoint restart	
Does SysA_CD_Partner require data to be signed by the Router :	No
Does SysA_CD_Partner require data to be encrypted by the Router :	No
Community Membership	
Community Name:	FirstCommunity
Joined Date:	Today

< Back Cancel Finish

Figure 5-62 Confirmation panel

12. When the partner is added successfully, close the browser window and return to the main Sterling File Gateway browser window.
13. Repeat these steps from step 1 to create a second partner for SysE_Partner, who is an internal partner that sends and receives files using the Sterling Connect:Direct protocol. Use the values shown in Table 5-3.

Table 5-3 Parameters for SysE_Partner

Parameter name	Value
Community	FirstCommunity
Partner name	SysE_Partner
Phone	9876
E-mail address	syse@itso_redbooks.com
User name	SysE_Partner
Password	itso4you
First name	SysE_Firstname
Last name	SysE_Lastname
SysE_Partner is a Consumer of Data	Checked
Partner will listen for a connection to consume data.	Checked
SysE_Partner is a Producer of Data	Checked
Will SysE_Partner use either SSH/SFTP or SSH/SCP protocol to initiate connections?	No
SysE_Partner will	Listen for Sterling Connect:Direct connections
Local Node Name	SysC_CDSA
Remote Node Name	SysE_CD
Remote User Id	cdadmin
Remote password	cdadmin
Does SysE_Partner require data to be signed by the Router?	No
Does SysE_Partner require data to be encrypted by the Router?	No

The confirmation page looks as shown in Figure 5-63.

Add Trading Partner: Confirm	
Profile	
Partner Name:	SysE_Partner
Address:	
Phone:	9876
Email Address:	sysE@itso_redbooks.com
User Account	
User Name:	SysE_Partner
Password:	*****
First Name:	SysE_Firstname
Last Name:	SysE_Lastname
Protocol	
Partner Role:	Producer of Data, Consumer of Data
Connection Direction:	Listen Connection
Transport Method:	Connect:Direct
<ul style="list-style-type: none"> • Local Node Name • Remote Node Name • Remote User Id • Remote Password • Remote File Name • Disposition to be applied to remote file • Sysopts string to be applied to remote file • CheckpointInterval to be used for checkpoint restart 	SysC_CD_SA SysE_CD ccladmin *****
Does SysE_Partner require data to be signed by the Router :	No
Does SysE_Partner require data to be encrypted by the Router :	No
Community Membership	
Community Name:	FirstCommunity
Joined Date:	Today
<input type="button" value="Back"/> <input type="button" value="Cancel"/> <input type="button" value="Finish"/>	

Figure 5-63 Details for SysE_Partner

Creating the PassThrough_RouteByMailbox routing channel template

A routing channel template defines the structure through which the routing occurs. It specifies producer and consumer mailbox structures and file structures and basically establishes the producer-consumer relationship for file transfers.

In this example, a routing channel template called PassThrough_RouteByMailbox is created. This template takes files and places them in the following Producer Mailbox path:

```
<producer_partner_name>/To_<consumer_partner_name>
```

In this scenario, files are sent from SysA_CD_Partner to SysE_Partner. Files are placed initially in the SysA_CD_Partner/To_SysE_Partner mailbox. When the Routing Channel is created, it is configured to route files that are placed in this mailbox to SysE_Partner, where it ultimately is routed using the Sterling Connect:Direct protocol that we configured when creating SysE_Partner.

This routing channel template is also used in other scenarios in this book where a producer partner needs to transfer a file to different consumer partners (each with different transport protocols) and does so by placing the file into an appropriate mailbox path that is configured to route to a specific partner.

To create the routing channel template:

1. In the main Sterling File Gateway panel, go to **Routes** → **Templates**, and then click **Create** (Figure 5-64).

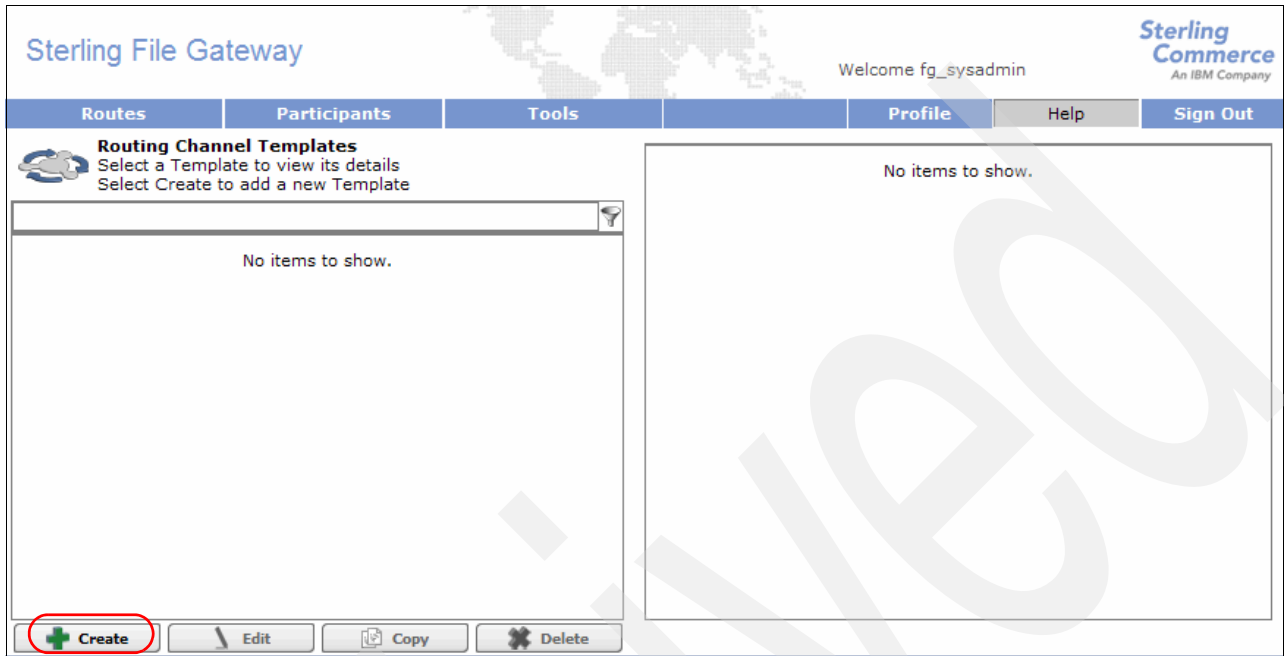


Figure 5-64 Create a new routing channel template

2. Enter a template name of `PassThrough_RouteByMailbox`, and leave the default option of `Static` (Figure 5-65).

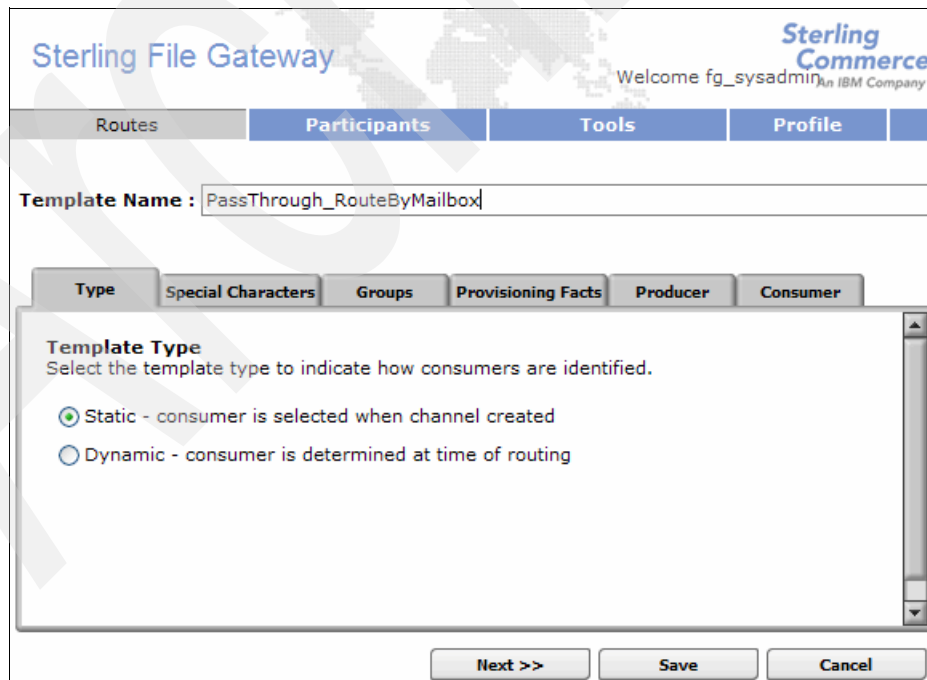


Figure 5-65 Creating a static routing channel template

- On the Groups tab, click **Add** under the box for Producer Groups. Select **All Partners** from the drop-down menu. Click **Add** under the box for Consumer Groups, and again select **All Partners** from the drop-down menu (Figure 5-66).



Figure 5-66 Select groups for this template

- On the Producer tab, enter the following string in the Pattern for Producer Mailbox Path field:
`/${ProducerName}/To_${ConsumerName}`
 Click **Add** to add the file structure definition.
- Enter the following file name pattern as a regular expression (Figure 5-67):
`.+`
 Click **Save**.

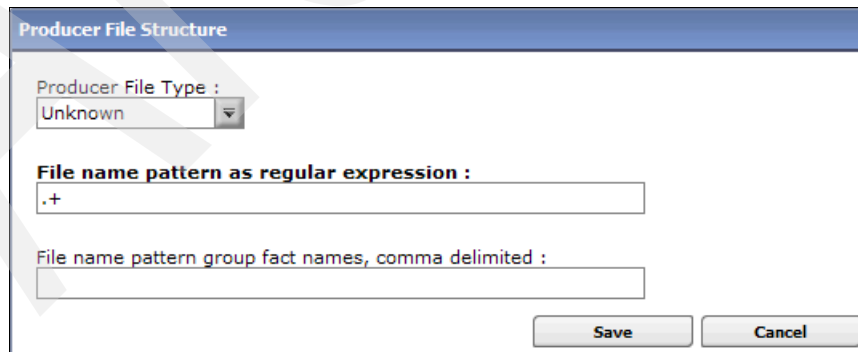


Figure 5-67 Create producer file structure to accept all files

Your Producer tab should now look as shown in Figure 5-68.

Template Name :

Type Special Characters Groups Provisioning Facts **Producer** Consumer

Producer Mailbox and Files
Describe the pattern for producer mailbox and the structure of files placed in the producer mailbox

Pattern for Producer Mailbox Path :

Producer File Structure Description
Unknown

Figure 5-68 Completed Producer tab

6. On the Consumer tab, click **Add** to open the New Delivery Channel window (Figure 5-69). Click **Add** under the Consumer File Structures heading.

New Delivery Channel

Pattern for Consumer Mailbox Path :

If checked mailboxes matching this pattern may be created on demand

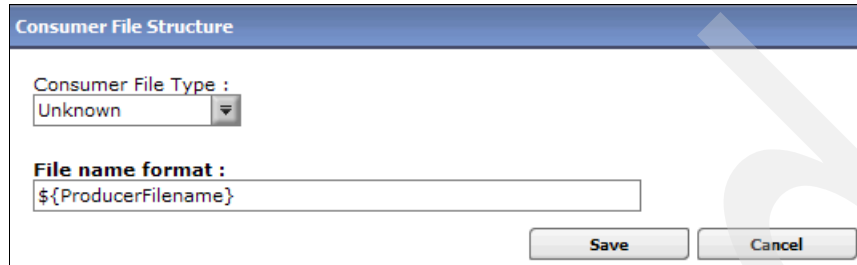
Consumer File Structures
Create or edit your file structures here

Figure 5-69 New Delivery Channel window

7. Select **Unknown**, and then enter the following file name format (Figure 5-70):

`${ProducerFilename}`

This entry maintains the file name as specified when it is placed in the mailbox of the producer (SysA_CD_Partner). Click **Save**.



Consumer File Structure

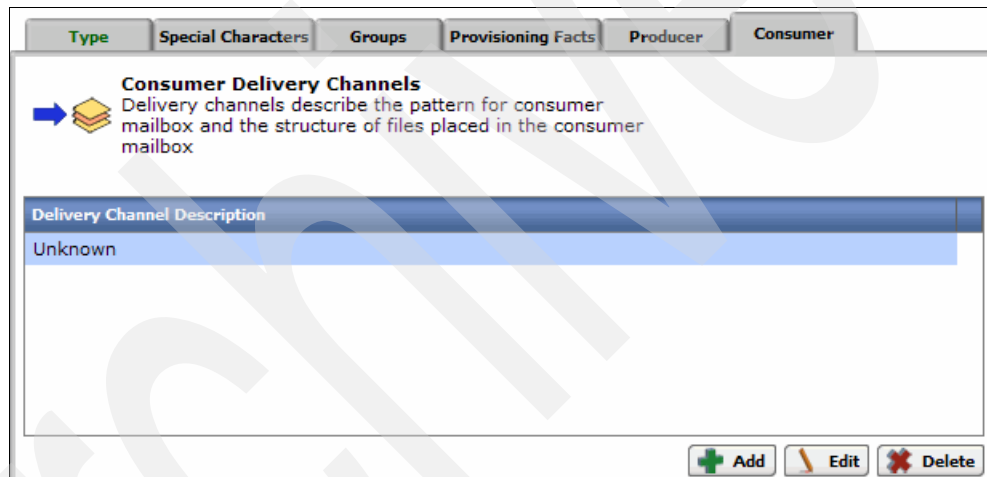
Consumer File Type :
Unknown

File name format :
`${ProducerFilename}`

Save Cancel

Figure 5-70 Define the Consumer File Structure

8. Back on the New Delivery Channel panel, click **Save** to create the new routing channel template. Your Consumer tab should now look as shown in Figure 5-71.



Type Special Characters Groups Provisioning Facts Producer **Consumer**

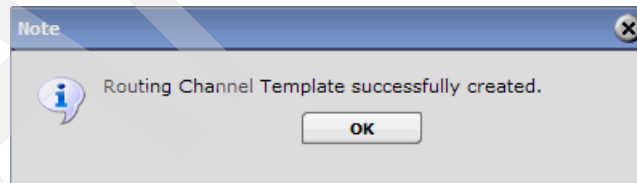
Consumer Delivery Channels
Delivery channels describe the pattern for consumer mailbox and the structure of files placed in the consumer mailbox

Delivery Channel Description
Unknown

+ Add Edit Delete

Figure 5-71 Completed Consumer tab

9. A success message displays (Figure 5-72). Click **OK**.



Note

Routing Channel Template successfully created.

OK

Figure 5-72 Success message

Creating the routing channel

The final step is to create the routing channel to define the route from SysA_CD_Partner to SysE_Partner. In the main Sterling File Gateway browser, select **Routes** → **Channels**. Click **Create**. Select the values listed in Table 5-4, and click **Save**.

Table 5-4 Values for routing channel

Parameter	Value
Template	PassThrough_RouteByMailbox
Producer	SysA_CD_Partner
Consumer	SysE_Partner

Figure 5-73 shows the successfully created routing channel.

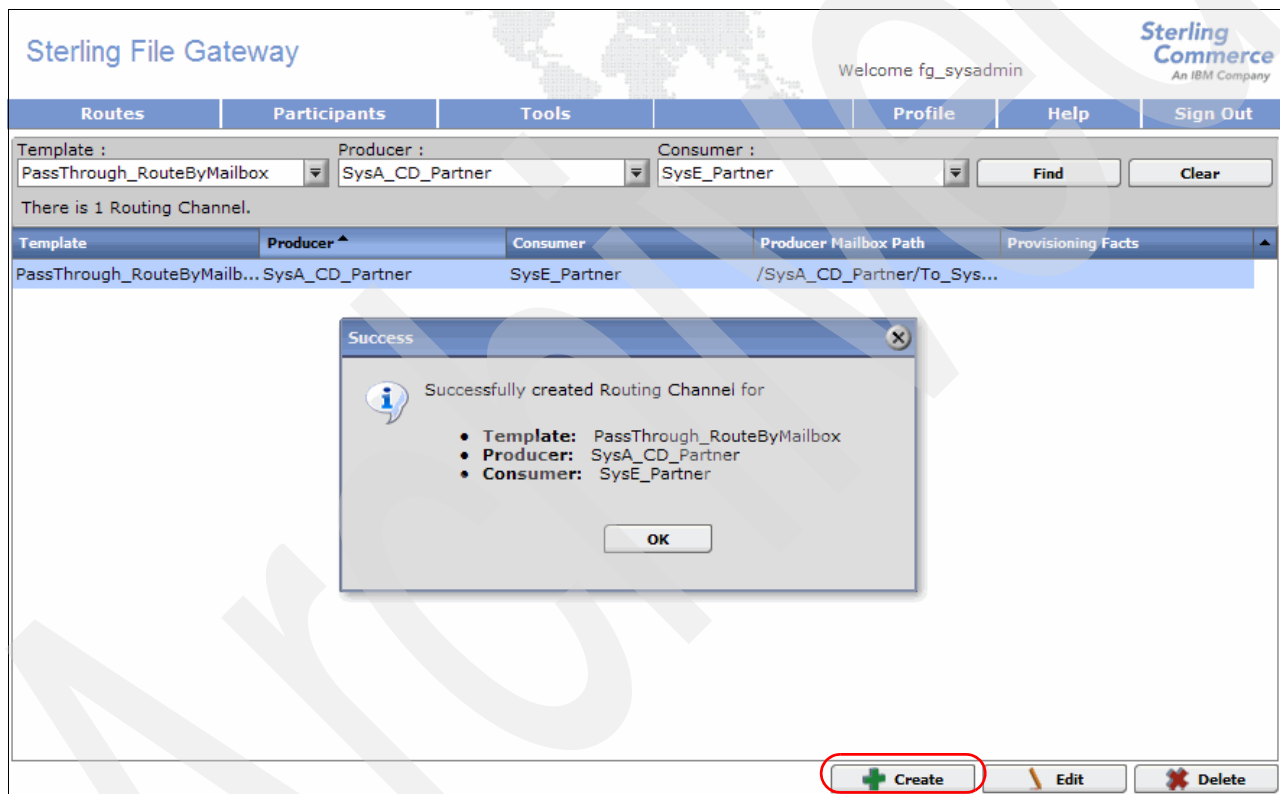


Figure 5-73 Create routing channel

5.3 Testing the flows

This section documents the movement of files as they are transferred from Company A to Company B. We discuss the procedures to initiate a file transfer from an external and internal organization to an internal organization.

5.3.1 Sterling Connect:Direct push file to Sterling Connect:Direct

The following scenario demonstrates transferring a file between Sterling Connect:Direct for Microsoft Windows and Sterling Connect:Direct for Linux using Sterling Secure Proxy (Figure 5-74).

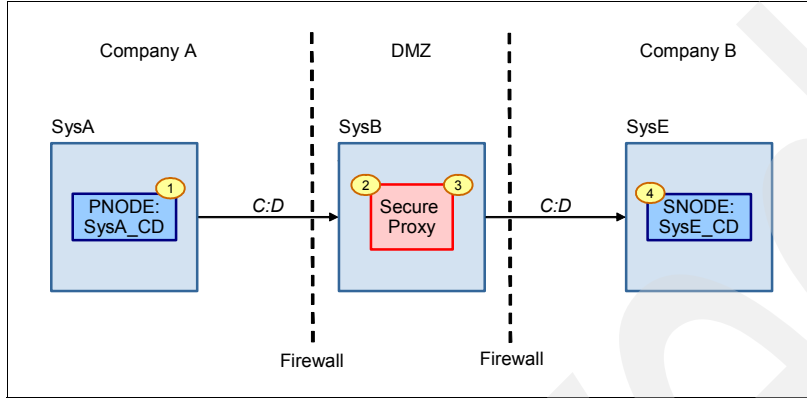


Figure 5-74 External Sterling Connect:Direct to internal Sterling Connect:Direct using Sterling Secure Proxy

To start the file transfer:

1. Log on to machine SysA and start the CD Requester tool by going to **Start** → **All Programs** → **Sterling Commerce Connect Direct 4.5.01** → **CD Requester** (Figure 5-75).

Note: This scenario uses the CD Requester tool and the GUI Send/Receive File option. You can also achieve the same file transfer using a process file.

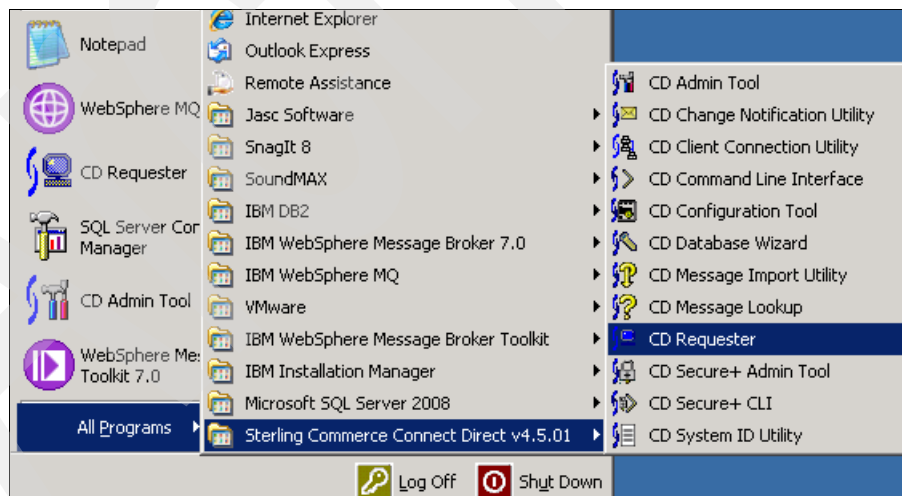


Figure 5-75 Start CD Requester

The CD Requester starts (Figure 5-76).

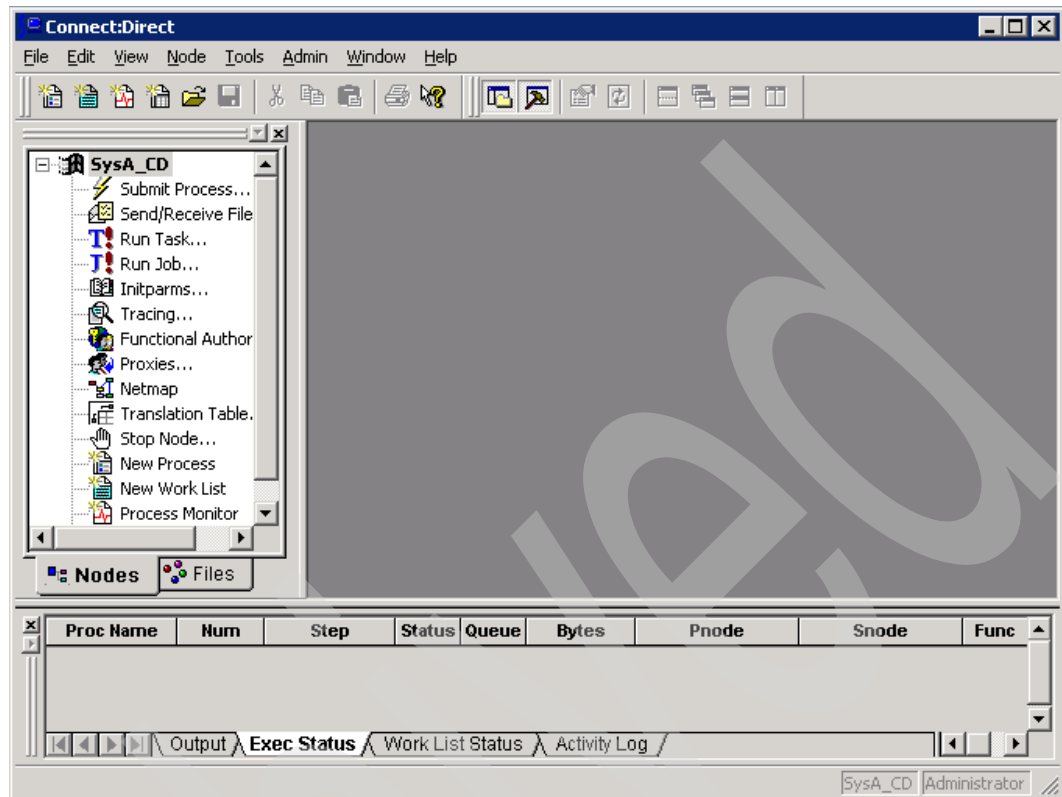


Figure 5-76 CD Requester

2. Right-click **SysA_CD**, and click **Attach** to connect to the Connect:Direct node SysA_CD (Figure 5-77).

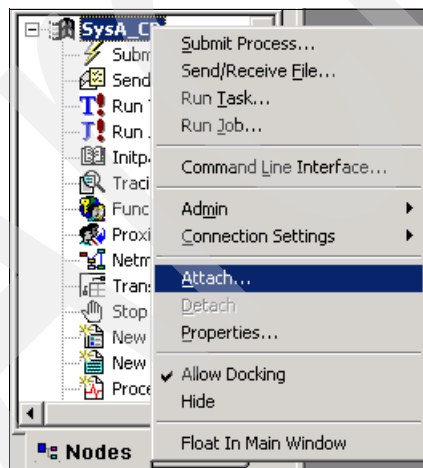


Figure 5-77 Connect to the SysA_CD Connect:Direct node

3. Enter the user ID and password for the Connect:Direct node, and then click **OK** (Figure 5-78).

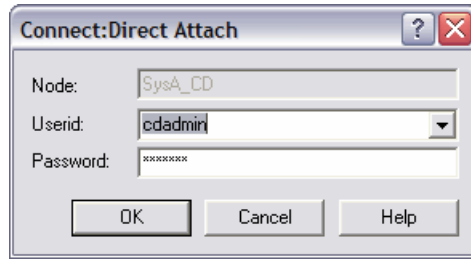


Figure 5-78 Userid and Password for the SysA_CD node

The icon next to SysA_CD changes from black-and-white to color to indicate that it has successfully connected to the Connect:Direct node.

4. Double-click **Send/Receive File** to open a window where you can enter details about the file to be transferred to SysE (Figure 5-79). Select the SNODE from the drop-down menu of SysE_CD. Select a file to send from the C:\CDWindows_files\upload directory, and enter the file name without the directory path. If no suitable file exists, create a simple file in that directory first. For our scenario, we created a text file called SysA_to_SysE.txt and populated it with sample text.

Enter the destination as:

SysA_to_SysE.txt

Select a disposition of **RPL - Replace or create a new file**.

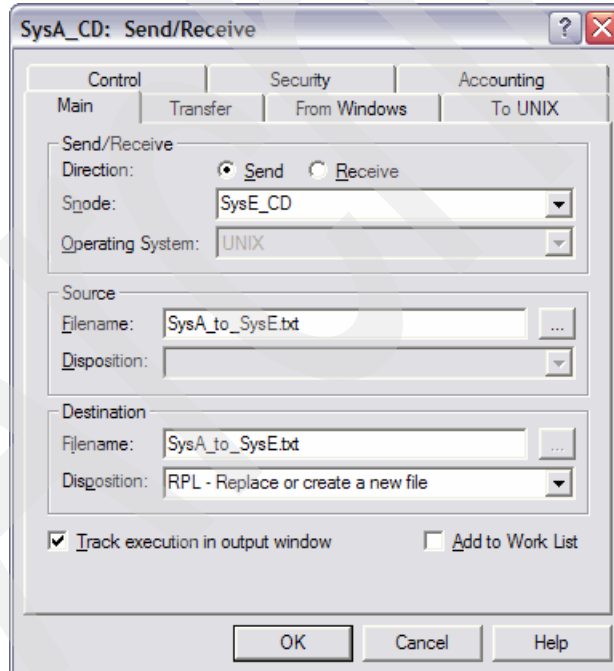


Figure 5-79 Values for initiating the transfer

5. Select the **Security** tab and enter the user ID and password for the SNODE SysA_CD_Partner partner as it is defined in Sterling File Gateway (Figure 5-80). The user ID defines the destination mailbox path, because it gives access only to the SysA_CD_Partner mailboxes. The password must match the one that is defined in Sterling File Gateway to allow access to that partner's mailbox.

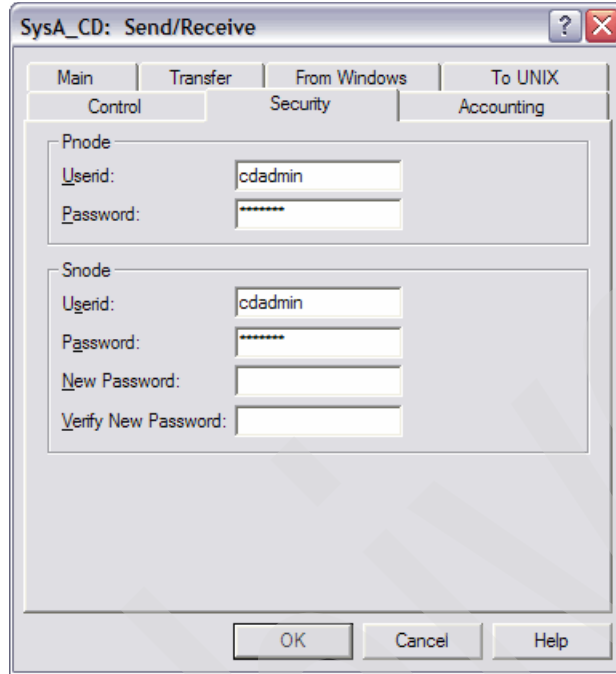


Figure 5-80 Enter Sterling File Gateway partner credentials

6. Click **OK** to start the transfer.

When the transfer starts, it displays in the Exec Status window in the CD Requester tool. Click the grey box to the left of that window to show the status of the Sterling Connect:Direct file transfer between SysA_CD and SysC_CD. Note that Sterling Connect:Direct has no knowledge of the ongoing routing to the consumer partner. As far as Company A is concerned, it has now sent its file to Company B and it has no need to know how it is routed or processed inside the internal company.

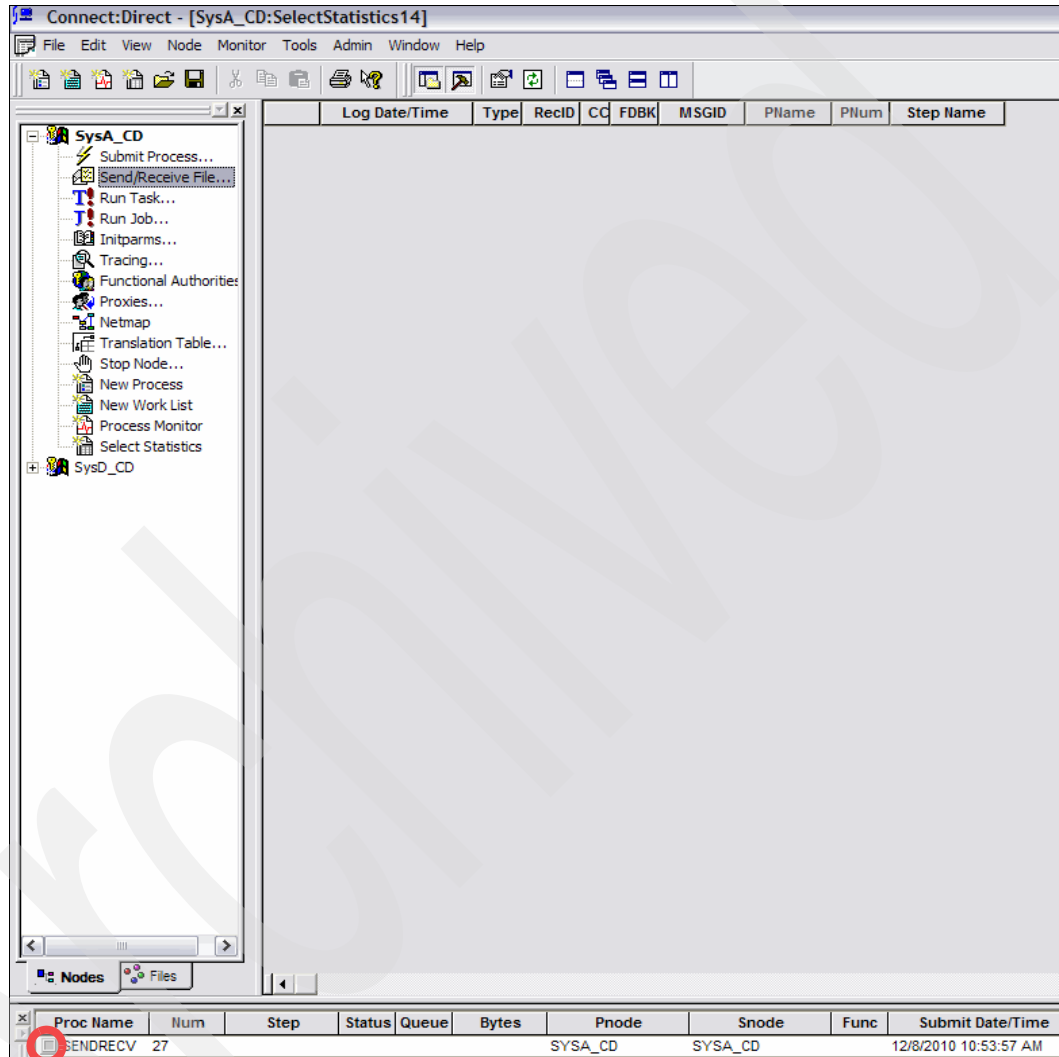


Figure 5-81 Click the grey box to open the transfer status

- The transfer process status displays in the Process Execution Statistics window. Scrolling to the bottom of the window reveals a Copy Operation Successful message (Figure 5-82).

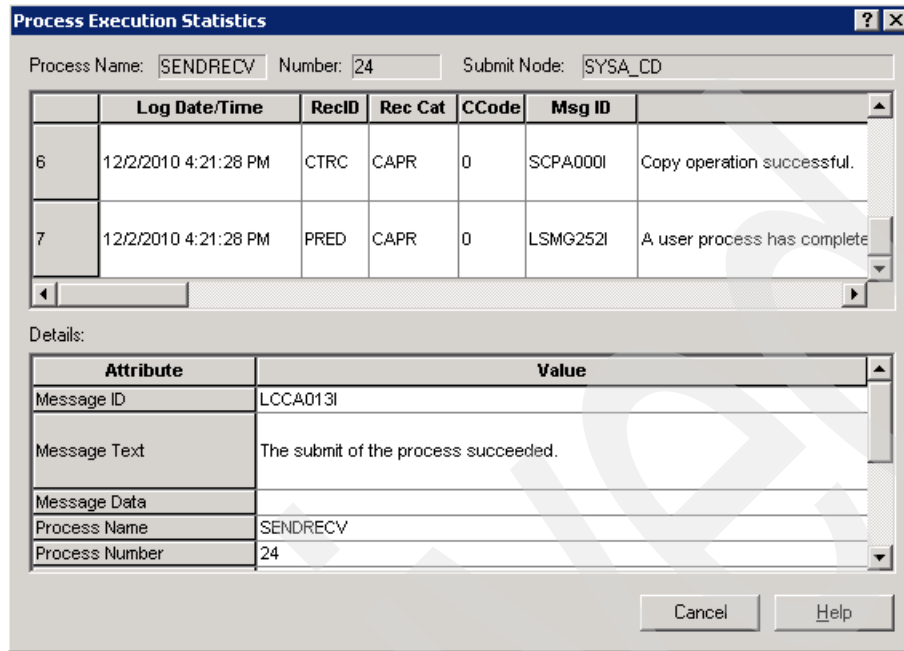


Figure 5-82 File transfer status

You can also determine end-to-end success by looking in the `/opt/cdunix/downloads/` directory on SysE to make sure that the file arrived successfully.

5.3.2 External Sterling Connect:Direct push to Sterling File Gateway to internal Sterling Connect:Direct

This scenario demonstrates transferring a file from an external trading partner using Sterling Connect:Direct to an internal Sterling File Gateway instance (Figure 5-83). Sterling Secure Proxy is used in the DMZ. Sterling File Gateway processes a routing channel, which triggers the file to proceed to an internal Sterling Connect:Direct.

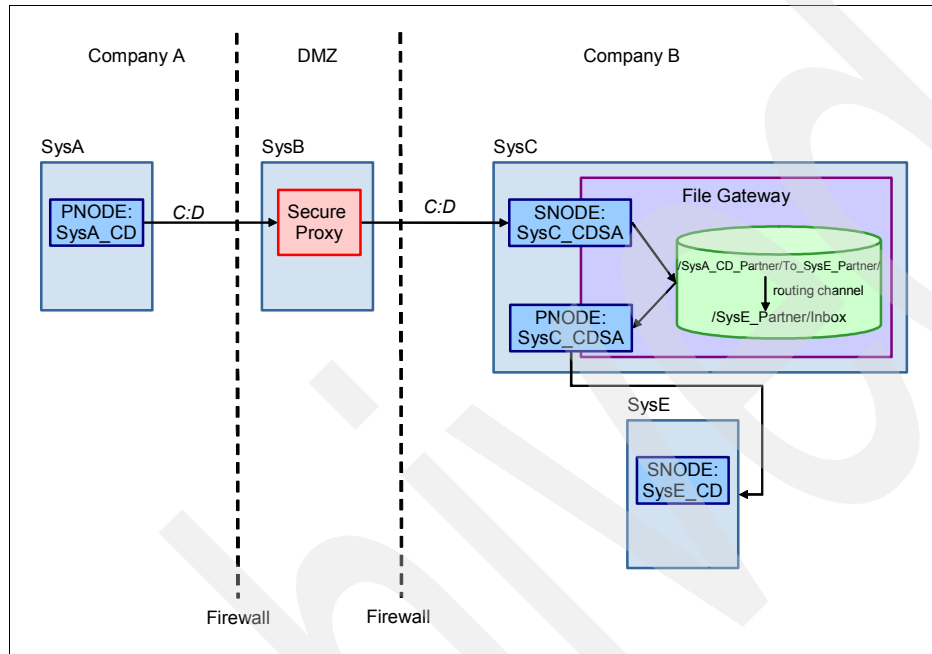


Figure 5-83 Scenario diagram

Company A uses Sterling Connect:Direct to send a file through a proxy server in a DMZ to Company B. Company A uses a Connect:Direct node called SysA_CD on SysA.

Sterling File Gateway has this node configuration stored as external partner SysA_CD_Partner. A script running on SysA initiates a file transfer from the PNODE SysA_CD to the SNODE of SysC_CDSA (using the secure proxy server). SysC_CDSA is actually a Connect:Direct server adapter running in Sterling B2B Integrator on SysC.

SysC_CDSA receives the file from SysA_CD and, based on the user ID and password sent in the file transfer, determines that the file has come from partner SysA_CD_Partner.

A routing channel is configured such that a file received from SysA_CD_Partner is routed from the /SysA_CD_Partner/To_SysE_Partner/ mailbox to the SysE_Partner, which is a dedicated Connect:Direct node internal to Company B's network.

When the file is routed from the Sterling File Gateway to SysE_Partner, it initiates a second file transfer from the SNODE SysC_CDSA to PNODE SysE_CD.

To run the outbound scenario:

1. Log on to SysA and start the CD Requester tool by going to **Start → All Programs → Sterling Commerce Connect Direct 4.5.01 → CD Requester**.

Scenario note: This scenario uses the CD Requester tool and the GUI Send/Receive File option. You can achieve the same file transfer using a process file.

2. Right-click **SysA_CD**, and then click **Attach** to connect to the SysA_CD node (Figure 5-84).

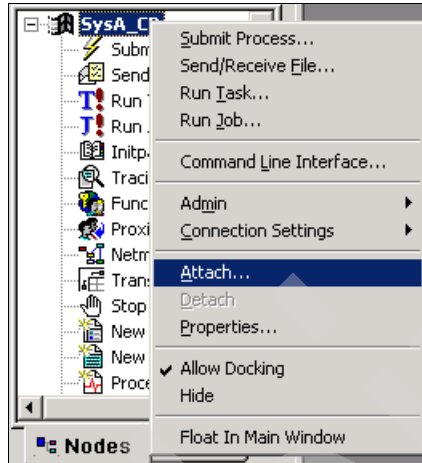


Figure 5-84 Connect to the SysA_CD Connect:Direct node

3. Enter the user ID and password for the node, and then click **OK** (Figure 5-85).

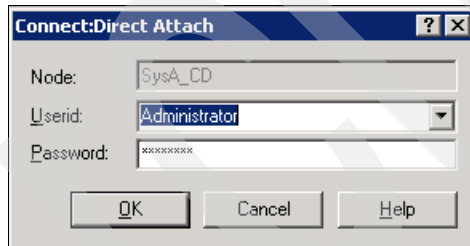


Figure 5-85 Userid and Password for the SysA_CD node

The icon next to SysA_CD changes from black-and-white to color to indicate that it is successfully connected to the node.

4. Double-click **Send/Receive File** to open a window where you can enter details about the file to be transferred to SysE (Figure 5-86):
 - a. In the Send/Receive section, select **Send** and select **SysC_CDSA** as the SNODE.
 - b. In the Source section, select a file to send from the C:\CDWindows_files\upload directory, and enter the file name without the directory path. If no suitable file exists, create a simple file in that directory first. For our scenario, we created a text file called SysA_to_SysE.txt and populated it with sample text.
 - c. In the Destination section, enter the destination as:
 /mailbox/To_SysE_Partner/SysA_to_SysE.txt
 This mailbox path, along with the user ID and password that is supplied in the transfer, is used to put the file in the correct mailbox and causes the routing channel to forward the file to SysE_Partner.
 - d. Select a disposition of **RPL - Replace or create a new file**.

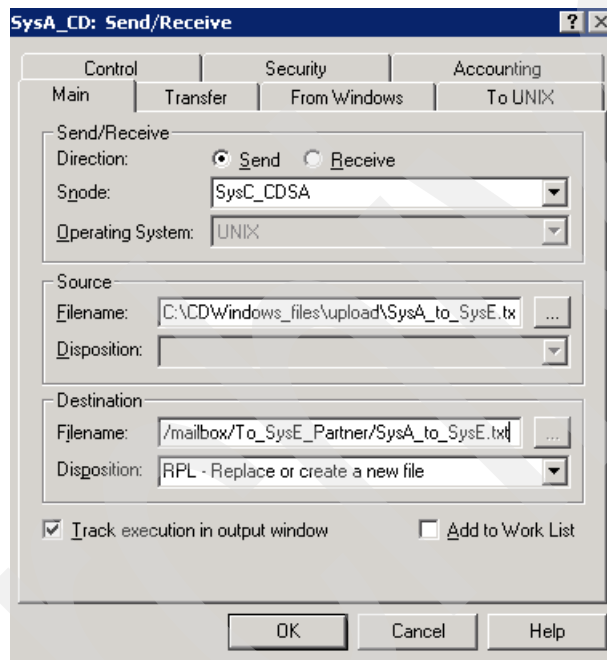


Figure 5-86 Values for initiating the transfer

5. Select the **Security** tab, and enter the user ID and password for the SNODE SysA_CD_Partner partner because it is defined in Sterling File Gateway (Figure 5-87). The user ID defines the destination mailbox path, because it gives access only to SysA_CD_Partner's mailboxes. The password must match the one defined in Sterling File Gateway to allow access to that partner's mailbox.

The screenshot shows a dialog box titled "SysA_CD: Send/Receive" with a standard Windows window title bar (minimize, maximize, close buttons). The dialog has a tabbed interface with four tabs: "Main", "Transfer", "From Windows", and "To Windows". The "Security" tab is selected. Below the tabs, there are three sub-sections: "Control", "Security", and "Accounting". The "Security" sub-section is active and contains two groups of fields. The first group, labeled "Pnode", has "Userid:" and "Password:" fields. The second group, labeled "Snode", has "Userid:" (containing "SysA_CD_Partner"), "Password:" (with masked characters "xxxxxxxx"), "New Password:", and "Verify New Password:" fields. At the bottom of the dialog are "OK", "Cancel", and "Help" buttons.

Figure 5-87 Enter Sterling File Gateway partner credentials

- Click **OK** to start the transfer. When the transfer is started, it displays in the Exec Status window in the CD Requester tool. Click the grey box to the left of that window to show the status of the Sterling Connect:Direct file transfer between SysA_CD and SysC_CDSA Connect:Direct nodes (Figure 5-88). Note that Sterling Connect:Direct has no knowledge of the ongoing routing to the consumer partner. As far as Company A is concerned, they have now sent their file to Company B and have no need to know how it is routed or processed inside the internal company.

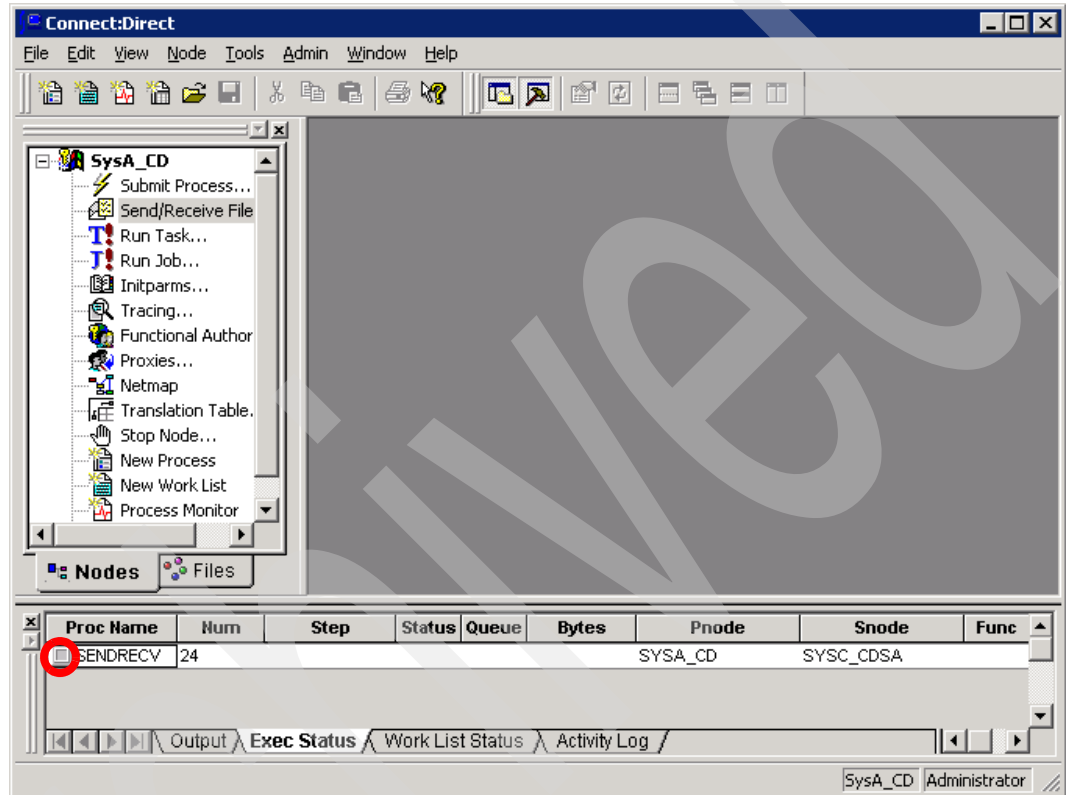


Figure 5-88 Click the grey box to open the transfer status

- The transfer process status displays in the Process Execution Statistics window. Scrolling to the bottom of the window reveals a Copy Operation Successful message (Figure 5-89).

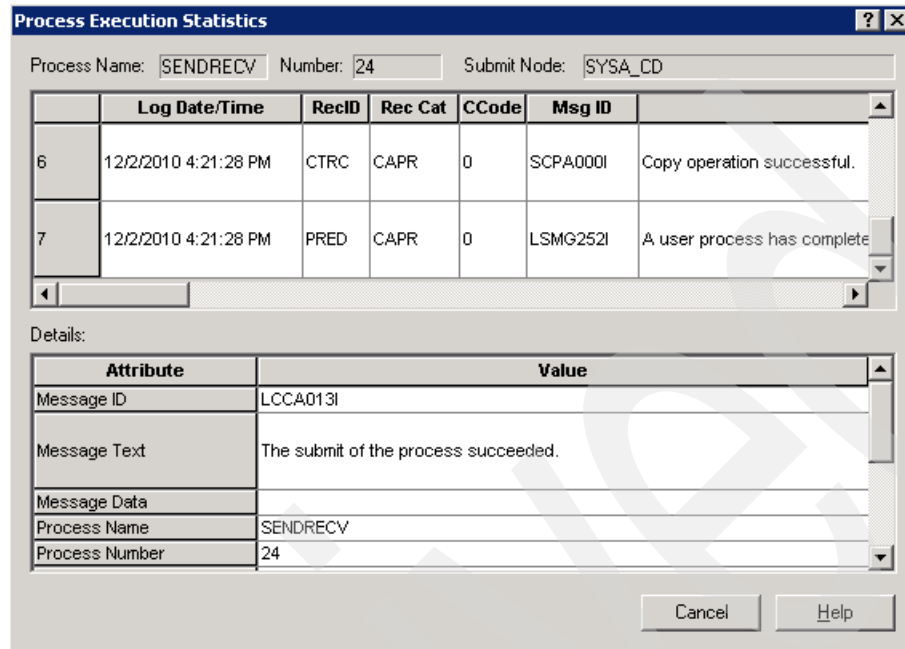


Figure 5-89 File transfer status

- The most basic way to determine end-to-end success is to look in the C:\downloads\ directory on SysE to make sure that the file arrived successfully.

5.3.3 Internal Sterling Connect:Direct push to Sterling File Gateway, to external Sterling Connect:Direct using Sterling Secure Proxy

A Connect:Direct process on Company B's SysE initiates a file transfer from the PNODE SysE_CD to the SNODE of SysC_CDSA. SysC_CDSA is actually a Connect:Direct server adapter running inside Sterling B2B Integrator on machine SysC (Figure 5-90).

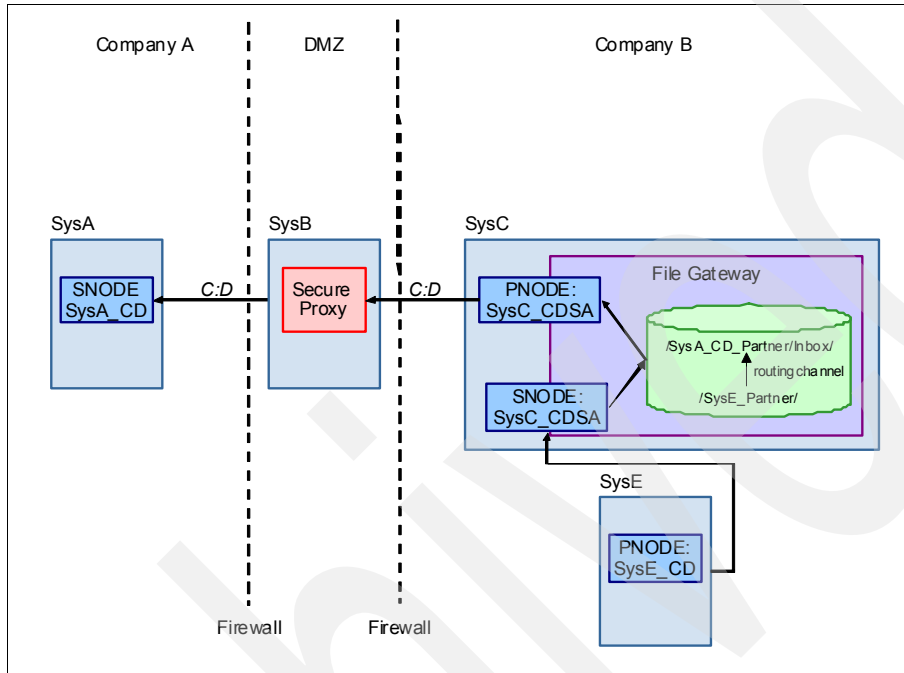


Figure 5-90 Scenario diagram

SysC_CDSA receives the file from SysE_CD and, based on the user ID and password sent in the file transfer, determines that the file has come from partner SysE_Partner.

A routing channel is configured such that a file received from SysE_Partner is routed from a mailbox /SysE_Partner to the external partner SysA_CD_Partner through a proxy server in a DMZ to Company A. Company A uses a node called SysA_CD on SysA.

When the file is routed from the Sterling File Gateway to SysA_CD_Partner, it initiates a second file transfer from the SNODE SysC_CDSA to PNODE SysA_CD using the proxy server.

5.3.4 Creating the route in Sterling File Gateway

You already created a routing channel template and routing channel in Sterling File Gateway to send files from SysA_CD_Partner to SysE_Partner. Now you need to create the necessary artifacts to run this transfer in reverse, that is, from SysE_Partner to SysA_CD_Partner.

Launching Sterling File Gateway

If Sterling File Gateway is not already open, log in to the Sterling File Gateway user interface:

1. Sterling B2B Integrator, and thus Sterling File Gateway, should already be started from the steps in “Starting Sterling B2B Integrator and logging in to the console” on page 116. If this is not the case and your server is stopped, start Sterling B2B Integrator by double-clicking the desktop icon **Sterling_Integrator_at_8080**.
2. Start Internet Explorer and go to:
`http://<servername>:<port>/filegateway/`
3. Log in (Figure 5-91). The default user ID for Sterling File Gateway is fg_sysadmin.

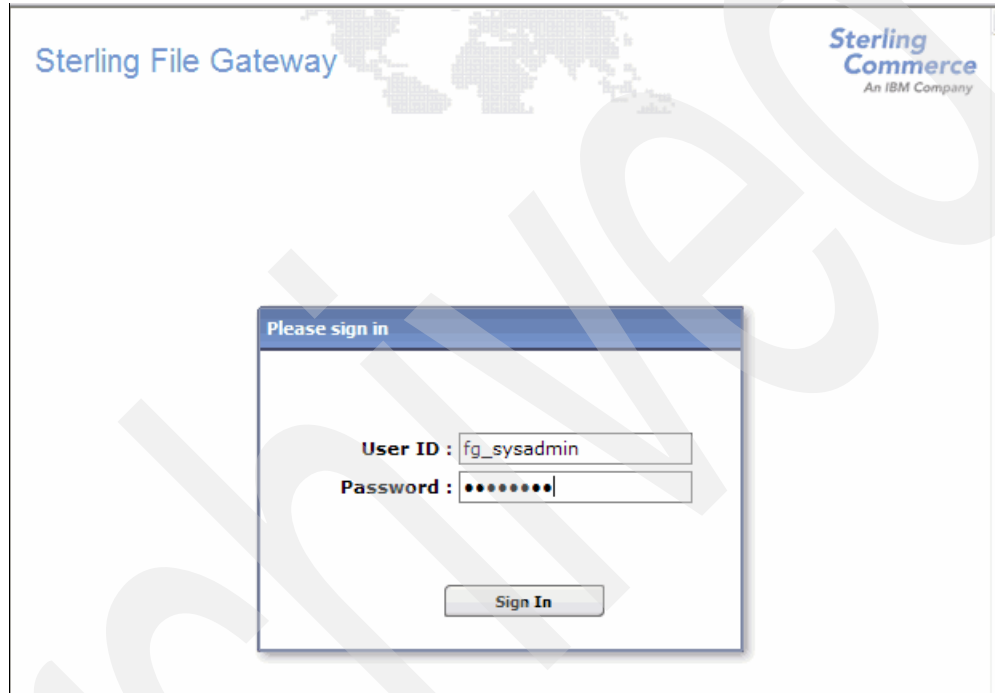


Figure 5-91 Login window for Sterling File Gateway

The main window for Sterling File Gateway opens (Figure 5-92).

The screenshot shows the Sterling File Gateway web interface. At the top left, it says "Sterling File Gateway". At the top right, it says "Welcome fg_sysadmin" and "Sterling Commerce An IBM Company". Below the header is a navigation bar with tabs: "Routes", "Participants", "Tools", "Profile", "Help", and "Sign Out". Below the navigation bar is a "Search Criteria" section. It includes a "Search Criteria" tab, a clock icon, and a link "Change to server time zone". Below this are buttons for "Find", "Clear", "Advanced Search", and "EBICS Search". The main content area is titled "Basic Search" and contains the following fields: "Producer :", "Consumer :", "Status :", "Protocol :", "Original File Name :", "Date (From) :", "Time (From) :", "Date (To) :", and "Time (To) :".

Figure 5-92 First window when logged in to Sterling File Gateway

Creating the PassThrough routing channel template

The routing channel template that you created earlier (see “Creating the PassThrough_RouteByMailbox routing channel template” on page 139) is set up to be slightly more complex than the one that is required in this case. The PassThrough_RouteByMailbox template is configured to place a file in a specific producer mailbox based on the intended consumer name.

In the previous scenario, any files coming into SysA_CD_Partner intended to be sent onwards to SysE_Partner go into the /To_SysE_Partner mailbox. It is configured this way so that in future scenarios, when SysA_CD_Partner needs to send a file to a different internal system over a different protocol, they can put those files into a different mailbox path (/To_<new_partner>) using this template. Therefore, the routing channels can be configured to route to different partners using different protocols, depending on which mailbox path a file arrives on.

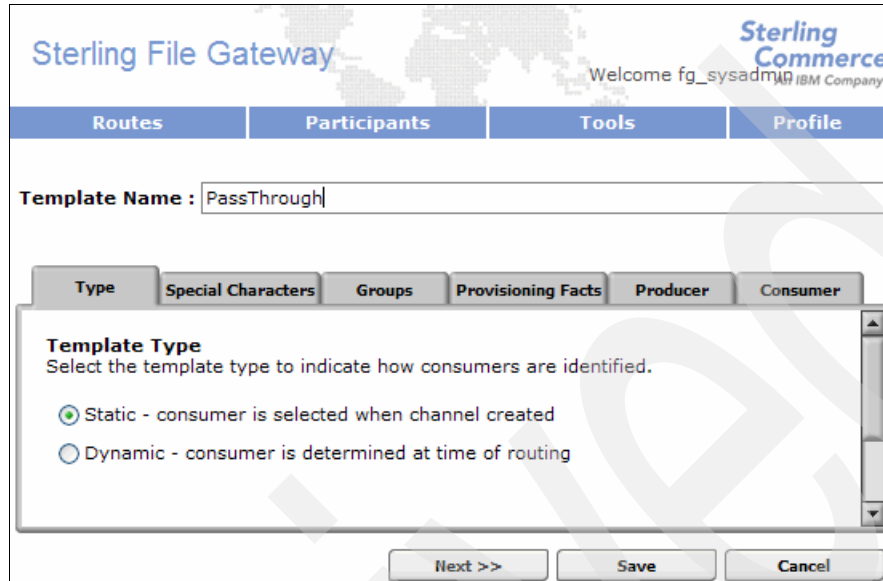
In this reverse scenario, this configuration is not necessary. There is a trading partner already configured in Sterling File Gateway, SysE_Partner, which owns an internal system, SysE. This system needs to send files out to the external trading partner, SysA_CD_Partner. This is the only trading partner with which SysE_Partner needs to communicate using the Sterling File Gateway.

A routing channel is set up such that any files that are placed in the root mailbox of SysE_Partner are transported to SysA_CD_Partner.

To create that routing channel, you need to create a routing channel template that is configured so that any files that are placed in the root mailbox of the producer partner are placed in the /Inbox mailbox of the consumer partner.

To create the routing channel template:

1. In the console, go to **Routes** → **Templates**, and click **Create**.
2. Enter a template name of PassThrough. Leave the template type as **Static** (Figure 5-93).



Sterling File Gateway

Welcome fg_sysadmin IBM Company

Routes Participants Tools Profile

Template Name : PassThrough

Type Special Characters Groups Provisioning Facts Producer Consumer

Template Type
Select the template type to indicate how consumers are identified.

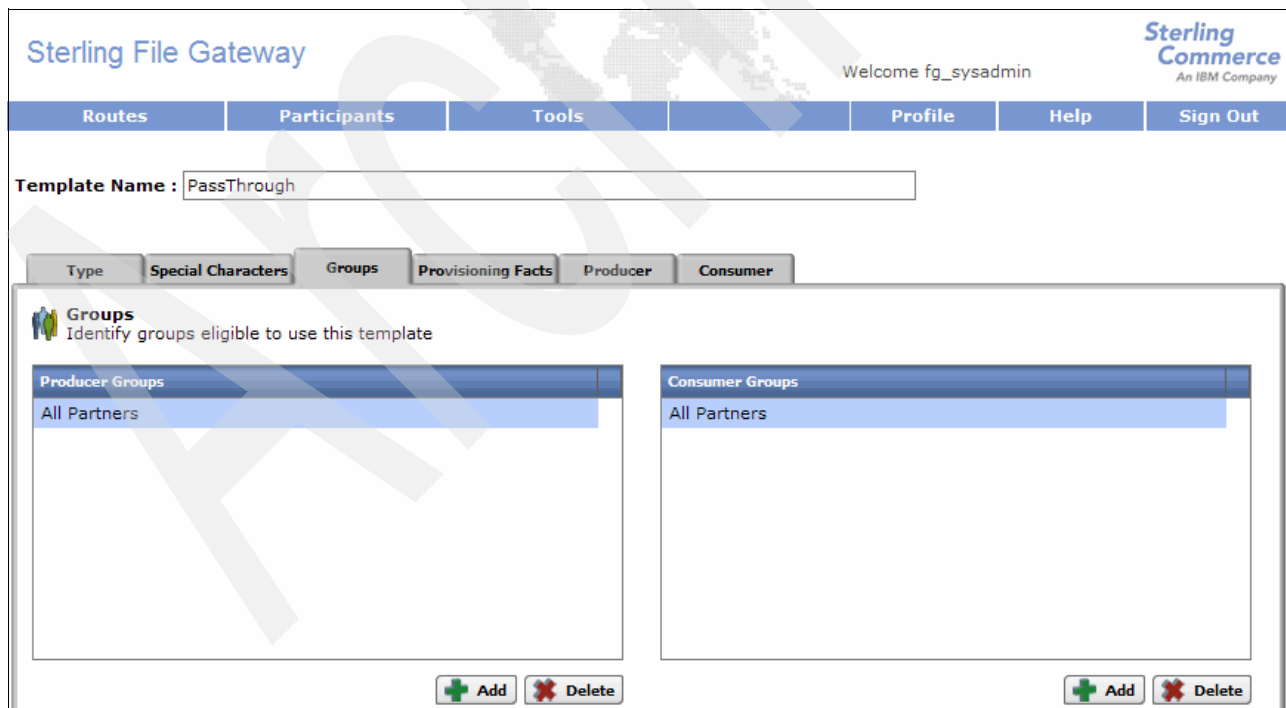
Static - consumer is selected when channel created

Dynamic - consumer is determined at time of routing

Next >> Save Cancel

Figure 5-93 Create simple PassThrough route template

3. On the Groups tab, click **Add** under the Producer Groups box. Select **All Partners**. Click **Add** under the Consumer Groups box, and again select **All Partners** (Figure 5-94).



Sterling File Gateway

Welcome fg_sysadmin

Routes Participants Tools Profile Help Sign Out

Template Name : PassThrough

Type Special Characters Groups Provisioning Facts Producer Consumer

Groups
Identify groups eligible to use this template

Producer Groups	Consumer Groups
All Partners	All Partners

+ Add - Delete + Add - Delete

Figure 5-94 Select groups for this template

4. Select the **Producer** tab. This template is set up such that any file placed in the root mailbox of the producer is routed, so leave the following default value:

/\${ProducerName}

Click **Add** to add the file structure definition. For the “File name pattern as a regular expression” option, enter:

.+

Click **Save** (Figure 5-95).



Producer File Structure

Producer File Type :
Unknown

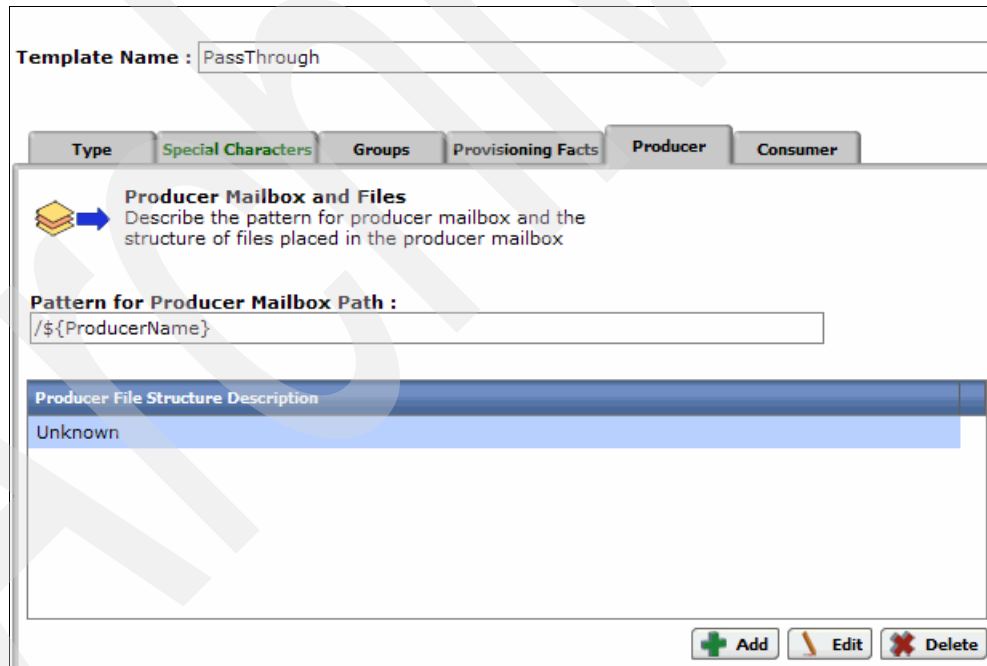
File name pattern as regular expression :
.+

File name pattern group fact names, comma delimited :

Save Cancel

Figure 5-95 Create producer file structure to accept all files

Your Producer tab should now look as shown in Figure 5-96.



Template Name : PassThrough

Type Special Characters Groups Provisioning Facts **Producer** Consumer

Producer Mailbox and Files
Describe the pattern for producer mailbox and the structure of files placed in the producer mailbox

Pattern for Producer Mailbox Path :
/\${ProducerName}

Producer File Structure Description
Unknown

+ Add Edit Delete

Figure 5-96 Completed Producer tab

5. On the Consumer tab, click **Add** to open the New Delivery Channel window (Figure 5-97). Click **Add** under the Consumer File Structures heading.

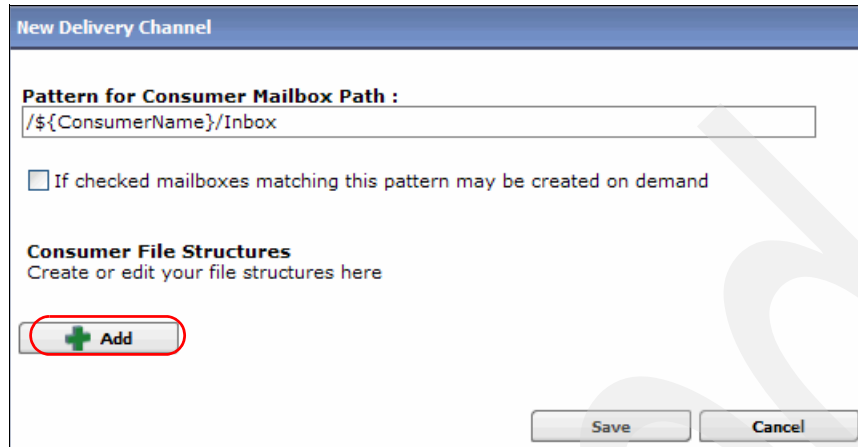


Figure 5-97 New Delivery Channel window

6. Select **Unknown** as the consumer file type, and use the following file name format:
\${ProducerFilename}
This maintains the file name as specified when it is placed in the mailbox of the producer (SysE_Partner in this scenario). Click **Save** (Figure 5-98).

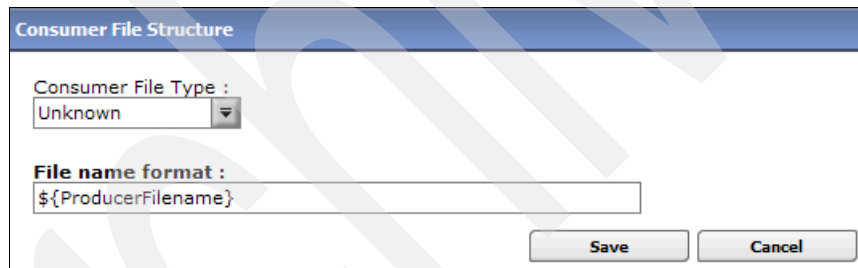


Figure 5-98 Define the Consumer File Structure

- Back on the New Delivery Channel panel, click **Save**. Your Consumer tab should now look as shown in Figure 5-99.

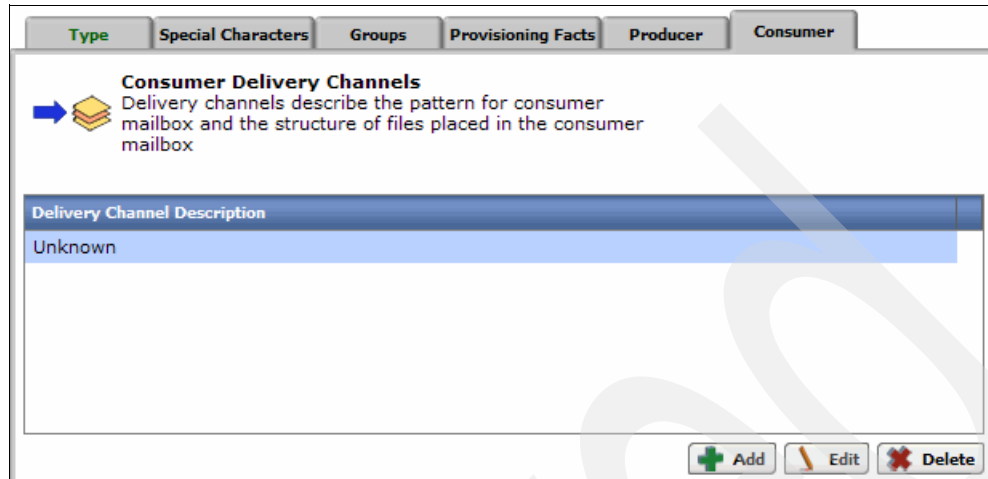


Figure 5-99 Completed Consumer tab

- Click **Save** to create the new routing channel template. You should see a success message (Figure 5-100).

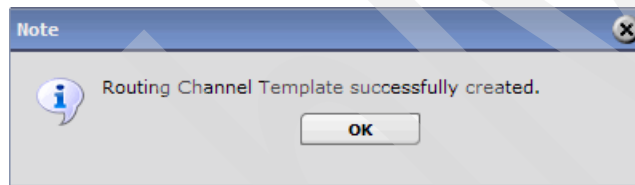


Figure 5-100 Success message

Creating the routing channel

Create the routing channel from SysE_Partner to SysA_CD_Partner using the newly created PassThrough template. Back in the main Sterling File Gateway browser, select **Routes** → **Channels**. Click **Create**. Select the values as shown in Table 5-5, and click **Save**.

Table 5-5 Values for routing channel

Parameter	Value
Template	PassThrough
Producer	SysE_Partner
Consumer	SysA_CD_Partner

Figure 5-101 shows this configuration.

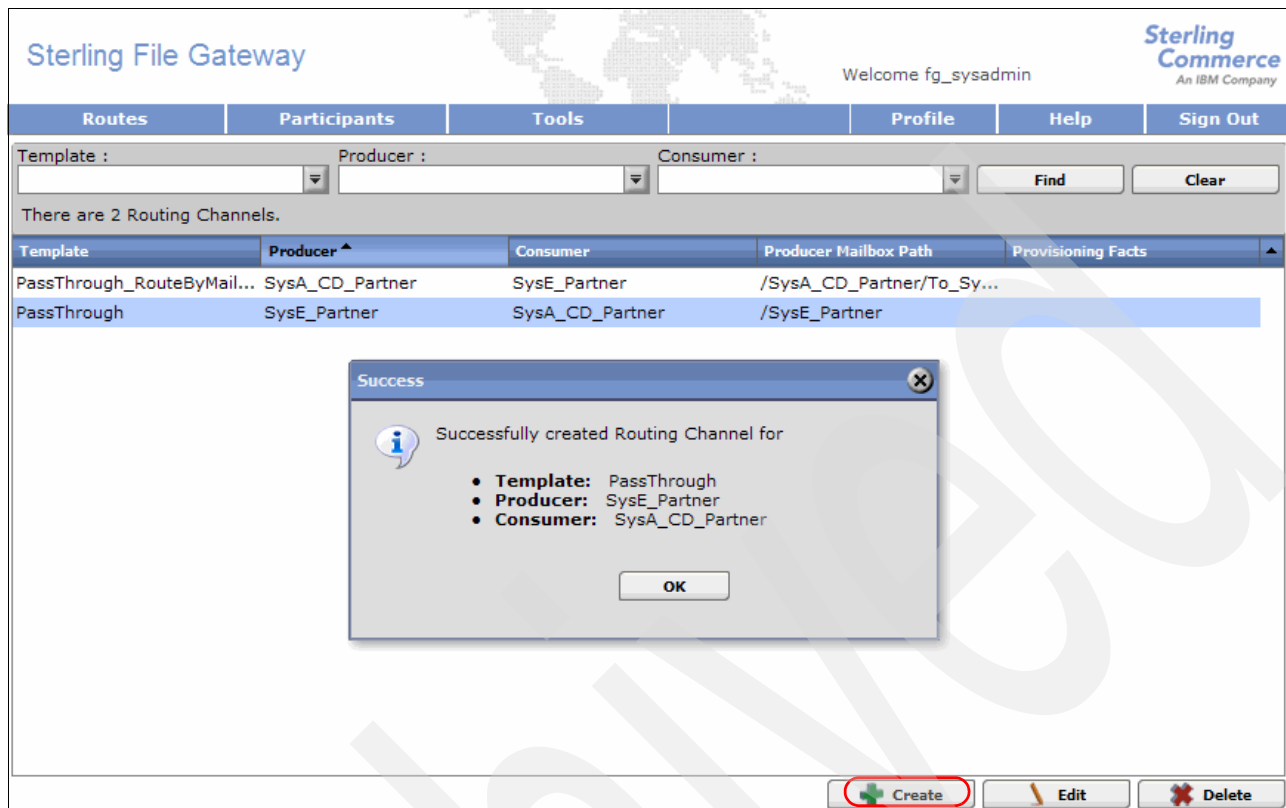


Figure 5-101 Create routing channel

Creating the Connect:Direct process text file on SysE

Create a plain text file on SysE similar to the one shown in Example 5-3. This process is used to instruct the Connect:Direct node SysE_CD on how to perform the file transfer. Save the file to the `ndm/bin` directory of the Sterling Connect:Direct installation. For this example, the directory is `/opt/cdunix/ndm/bin/`.

Example 5-3 Plain text file Connect:Direct process named "sample_push_to_A-SFG.cd"

```
/*
 * This sample process copies a text file "SysE_to_SysA.txt"
 * from "pnode" SysE_CD to SysC_CDSA to SysA_CD.
 */
ToCDSA process snode=SysC_CDSA snodeid=(SysE_Partner, itso4you)
step01 copy
    from( file = SysE_to_SysA.txt
         pnode
        )
    ckpt = 2M
    compress extended
    to( file = /mailbox/SysE_to_SysA.txt
       snode
```

```
    disp = rpl
  )
```

```
pend;
```

Submitting the Connect:Direct process text file on SysE_CD

Example 5-4 shows all the panel output for launching the Sterling Connect:Direct CLI. The Sterling Connect:Direct CLI can be started by executing the `direct` binary file located in the `/opt/cdunix/ndm/bin/` directory.

Example 5-4 Output from the Connect:Direct CLI

```
/opt/cdunix/ndm/bin> direct
```

```
*****
*
*          Connect:Direct for UNIX          *
*
*-----*
* Copyright (c) 1992, 2008 Sterling Commerce, Inc. *
*              Version 4.0.00                 *
*              Fix date: 30SEP2010            *
*****
```

Enter a ';' at the end of a command to submit it. Type 'quit;' to exit CLI.

```
Direct> submit file=sample_push_to_A-SFG.cd;
Process Submitted, Process Number = 29
Direct> sel stat pnum=29;
```

```
=====
                          SELECT STATISTICS
=====
P RECID LOG TIME          PNAME          PNUMBER  STEPNAME  CCOD  FDBK  MSGID
E RECID LOG TIME          MESSAGE TEXT
=====
P PSTR  12/08/2010 11:32:50 SENDRECV          29          0      XSMG200I
P RSST  12/08/2010 11:32:50 SENDRECV          29          0      XSMG201I
P CTRC  12/08/2010 11:32:50 SENDRECV          29          0      SCPA000I
P PRED  12/08/2010 11:32:50 SENDRECV          29          0      LSMG252I
E QCEX  12/08/2010 11:42:45 TCQ queue change from WAIT to EXEC, status PE.
E SPCA  12/08/2010 11:42:46 Secure+ bypassed for remote node SysC_CDSA.
E SSTR  12/08/2010 11:42:46 Session started, SNODE:SysC_CDSA, Protocol:tcp
                                LCLP          9.42.170.168, PORT=50581
                                RMTD          9.42.171.153, PORT=1364
P PSTR  12/08/2010 11:42:46 ToCDSA          29          0      XSMG200I
P LSST  12/08/2010 11:42:46 ToCDSA          29  step01  0      XSMG201I
P CTRC  12/08/2010 11:42:46 ToCDSA          29  step01  0      SCPA000I
P PRED  12/08/2010 11:42:46 ToCDSA          29          0      XSMG252I
E SEND  12/08/2010 11:42:46 Session ended, Session Manager shutting down SNODE:
                                SysC_CDSA
=====
```

```
Select Statistics Completed Successfully.
Direct>
```

The following commands are entered at the UNIX command prompt:

direct	Starts the Sterling Connect:Direct CLI
submit file=sample_push_to_A-SFG.cd;	Submits the job to Sterling Connect:Direct
sel stat pnum=29;	Views the outcome of the process

You can view the success or failure of a Connect:Direct process job in the statistics logs for a process job. Example 5-4 on page 165 shows that Sterling Connect:Direct assigned the submitted process a process number of 29. We used the **Select Statistics** command to discover the outcome. We entered the **Select Statistics** command by applying the process number. This avoids getting all statistics data returned. The success of the process is indicated in the Completion Code (CC) column. A return code of 0 indicates success. Any other value indicates failure.

5.4 Troubleshooting

If you experience difficulty running the scenario, see “Sterling File Gateway and Sterling B2B Integrator” on page 380.



External Transfers with Protocol Switching between IBM Sterling Connect:Direct and WebSphere MQ File Transfer Edition via Sterling File Gateway

This chapter shows how integration can be achieved between IBM Sterling Connect:Direct and WebSphere MQ File Transfer Edition by configuring Sterling File Gateway to handle protocol conversion. This chapter highlights the capabilities of Sterling File Gateway to enable an organization using one protocol to perform managed file transfer securely with an external organization using a different protocol.

To demonstrate the ability to integrate these two technologies, we created a scenario to do managed file transfer between an external partner using IBM Sterling Connect:Direct with a partner within an internal network using WebSphere MQ File Transfer Edition, via a secure proxy server.

6.1 Solution overview

This scenario shows how a file can come into an organization using one protocol (Sterling Connect:Direct) over the public internet through a secure, hardened interface in the demilitarized zone (DMZ) into an organization's secure internal network that uses a different protocol (WebSphere MQ File Transfer Edition) to send files within their enterprise.

An external Sterling Connect:Direct partner sends a file to Sterling File Gateway in the internal network through a proxy server. A proxy server is used in the DMZ to provide security between the external partner and the internal customer network. Sterling File Gateway is used to perform the routing of the file and protocol conversion from Sterling Connect:Direct to WebSphere MQ File Transfer Edition.

This scenario provides a detailed example of the customization to Sterling File Gateway and Sterling B2B Integrator to enable WebSphere MQ File Transfer Edition to be enabled as an available transfer protocol.

6.1.1 Appropriate use

This scenario demonstrates Sterling Connect:Direct communication between an external trading partner and Sterling File Gateway to the internal network using WebSphere MQ File Transfer Edition. However, this scenario can be varied to use any of the protocols supported on Sterling B2B Integrator and Sterling File Gateway.

Other possible protocols include:

- ▶ HTTP
- ▶ FTP
- ▶ SFTP
- ▶ FTP/S
- ▶ SSH/SCP
- ▶ AS2
- ▶ AS3
- ▶ Odette FTP

Note: Any data flowing between internal and external trading partners should always be encrypted. Certain scenarios shown here represent file transfers to and from an external entity using Sterling Connect:Direct. The Sterling Connect:Direct protocol can be encrypted using the Sterling Connect:Direct Secure Plus. Alternatively, the channel between two trading partners can be encrypted with a separate solution.

Chapter 7, "External transfers using IBM WebSphere Message Broker and IBM Sterling File Gateway" on page 245, describes a scenario using HTTP and SFTP to send and receive files from within the enterprise to external partners who communicate with those protocols.

To modify the scenario to use one of the other protocols, verify that the appropriate adapter is installed and configured in Sterling B2B Integrator and configure partners and routing channels in Sterling File Gateway that might be needed.

There are many ways to vary the WebSphere MQ File Transfer Edition deployment to meet specific needs by using more sophisticated topologies. These topologies are described in detail in *Getting Started with WebSphere MQ File Transfer Edition V7*, SG24-7760.

6.1.2 Business value

There is significant business value in integrating the Sterling Connect:Direct and WebSphere MQ File Transfer Edition to enable reliable and auditable external file transfers between organizations. One of the benefits to an organization using Sterling File Gateway is that it is not necessary for external partners to install the same file transfer protocol to perform business with that organization, so there are no additional software costs to connect to external partners.

Integrating Sterling File Gateway, Sterling Connect:Direct, and WebSphere MQ File Transfer Edition gives organizations a robust multi-enterprise managed file transfer solution. A file coming into an organization should pass through a secure mechanism in the demilitarized zone to enable security and terminate the external connection before passing the incoming data to secured, internal systems. This secure mechanism resides in the demilitarized zone to protect against unauthorized access. Once the external partner has been granted access through the demilitarized zone, Sterling File Gateway extends Sterling B2B Integrator's ability to switch between various protocols by allowing for movement of large and high-volume transfers, with end-to-end visibility of file movement in a process-oriented manner. Sterling File Gateway then leverages the WebSphere MQ File Transfer Edition managed file transfer backbone to move files within the enterprise to back-end applications.

This seamless integration from outside an organization, through a demilitarized zone and into an organization's managed file transfer infrastructure, creates a universal architecture that meets many departments' business needs. The flexibility and versatility of Sterling B2B Integrator teamed with the partner interfaces and administration featured in Sterling File Gateway give organizations a way to remain flexible to meet changing business demands. Sterling B2B Integrator features protocol switching, which allows for almost any protocol to be accepted from a trading partner, while still allowing for organizations to maintain the opportunity to use a single managed file transfer protocol internally. Sterling B2B Integrator partnered with Sterling File Gateway enables organizations to accept a wide variety of protocols and file types to meet external trading partners needs. Additionally, Sterling B2B Integrator and Sterling File Gateway make use of templates, which promotes re-use to reduce administration and development time.

Other benefits that are derived from this scenario include:

- ▶ Security
 - Real-time monitoring through a portal in Sterling File Gateway allows for IT and trading partners to gain visibility to in-flight file transfers.
 - Connections in the protected network can be configured with SSL using Sterling File Gateway and WebSphere MQ File Transfer Edition.
 - Encrypt data *in-flight* and *at-rest* in Sterling File Gateway.
 - Data transport security and data encryption support in Sterling B2B Integrator.
 - Sterling B2B Integrator features a secured mailbox repository to hold files.
 - Identity management, including authorization and authentication for trading partners defined in Sterling B2B Integrator.

- ▶ Administration, operation, and logging:
 - Having the ability to trace the file transfers end-to-end with Sterling File Gateway and WebSphere MQ File Transfer Edition reduces the resources required to troubleshoot file transfer failures and retries.
 - WebSphere MQ File Transfer Edition allows you to set up file transfers to occur at specified times or dates, or to be repeated at specific intervals. File transfers can also be triggered by a range of system events, such as new files or updated files.
 - Sterling File Gateway provides the ability for external partners to view the state of their own transfers and the ability to initiate upload and download requests.
 - Sterling File Gateway leverages auditing and reporting to provide metrics that verify regulatory compliance and adherence to service level agreements.
 - Monitor file transfer activity in Sterling File Gateway on an exception basis built on event management notifications from the event logging that provides a complete audit trail of file transfer activities.
 - Automate the replay, reprocess, and resend with failed file transfers in Sterling File Gateway.
 - WebSphere MQ File Transfer Edition provides full logging of transfers at both the source and destination systems for internal transfers.
 - Reusable templates in Sterling File Gateway and WebSphere MQ File Transfer Edition reduce staff time to build and maintain file transfer processes.

Sterling File Gateway can intelligently route files based on sender, file name, file type, and file contents.

Sterling Connect:Direct

Sterling Connect:Direct is a solution for secure, point-to-point file transfers. It has been optimized for high-volume, assured data delivery of files within and between enterprises, and provides script-based automation, scheduling, and alert notifications for 24x7 unattended operations. Sterling Connect:Direct allows organizations to automate the data exchange between mission-critical applications regardless of platform. The event-based architecture enables high volumes and large files, with no product-defined limits on file sizes. Sterling Connect:Direct, which also supports various clustering technologies and IBM Sysplex on the mainframe, provides built-in automation and checkpoint restart to ensure lights-out operations.

Automation and management

Automation and management provide these capabilities:

- ▶ Supports 24x7 unattended operations
- ▶ Schedules jobs on a one-time, recurring, or continuous basis
- ▶ Assigns and manages file transfer workload
- ▶ Event-driven alert notification
- ▶ Process language builds scripts to provide integration with back-end systems
- ▶ API and SDK for programmatic access by other applications
- ▶ Supports checkpoint restart
- ▶ Automatic recovery from network interruptions
- ▶ Automated alert notifications for success/failure

Security and compliance

Security and compliance are covered by these products:

- ▶ Standard Sterling Connect:Direct
 - Interfaces with operating system security for user authentication
 - Provides a complete audit trail of data movement through extensive statistics logs
- ▶ Sterling Connect:Direct Secure Plus
 - User authentication
 - X.509 certificates for authentication
 - Data encryption (SSL/TLS)
 - Certificate and Certificate Revocation List (CRL) checking
 - FIPS 140-2 and Common Criteria certification

Multiple platform support

The operating systems supported:

- ▶ z/OS and z/VSE
- ▶ OpenVMS
- ▶ i5/OS (OS/400)
- ▶ UNIX and Linux
- ▶ Windows
- ▶ HP NonStop
- ▶ Sterling Connect:Direct Select (Java version that can run on multiple platforms)

6.2 Scenario details

In this scenario we show that multiple enterprises can exchange files with each other between a Sterling Connect:Direct network and a WebSphere MQ File Transfer Edition network. Data is transferred through a proxy server in the demilitarized zone to Sterling File Gateway. Sterling File Gateway is used to perform the routing of the file and protocol conversion between Sterling Connect:Direct and WebSphere MQ File Transfer Edition.

We show how to configure and set up Sterling B2B Integrator and Sterling File Gateway to enable WebSphere MQ File Transfer Edition as an available transfer protocol that can be selected when creating a partner configuration in Sterling File Gateway. Specifically, configuration of Sterling B2B Integrator to use the WebSphere MQ Adapter and FTP Server adapter are needed to for our scenario. We also show how to set up a bridge agent that acts as the bridge between Sterling File Gateway and the WebSphere MQ File Transfer Edition backbone network within the enterprise.

In a inbound scenario, with Sterling Connect:Direct, the external partner uploads a file to a Connect:Direct server adapter through a proxy server and puts the file in their mailbox in Sterling File Gateway. The file is then directed through Sterling File Gateway's routing channels to the WebSphere MQ File Transfer Edition bridge agent. Sterling File Gateway creates a WebSphere MQ message to be placed in the command queue of a WebSphere MQ File Transfer Edition protocol bridge agent by utilizing the Sterling B2B Integrator WebSphere MQ adapter. The protocol bridge agent then sends the file to the agent running on the destination machine.

In a outbound scenario, an internal partner uploads a file to their Sterling File Gateway inbox with WebSphere MQ File Transfer Edition. The file is then directed via the Connect:Direct server adapter in the same way as the inbound scenario, which sends the file to the destination partner via proxy server with Sterling Connect:Direct.

6.2.1 Solution components

This section describes the components associated with each product in this solution (Figure 6-1). Certain components require specific configuration for the solution to work. We discuss the configuration steps required when necessary.

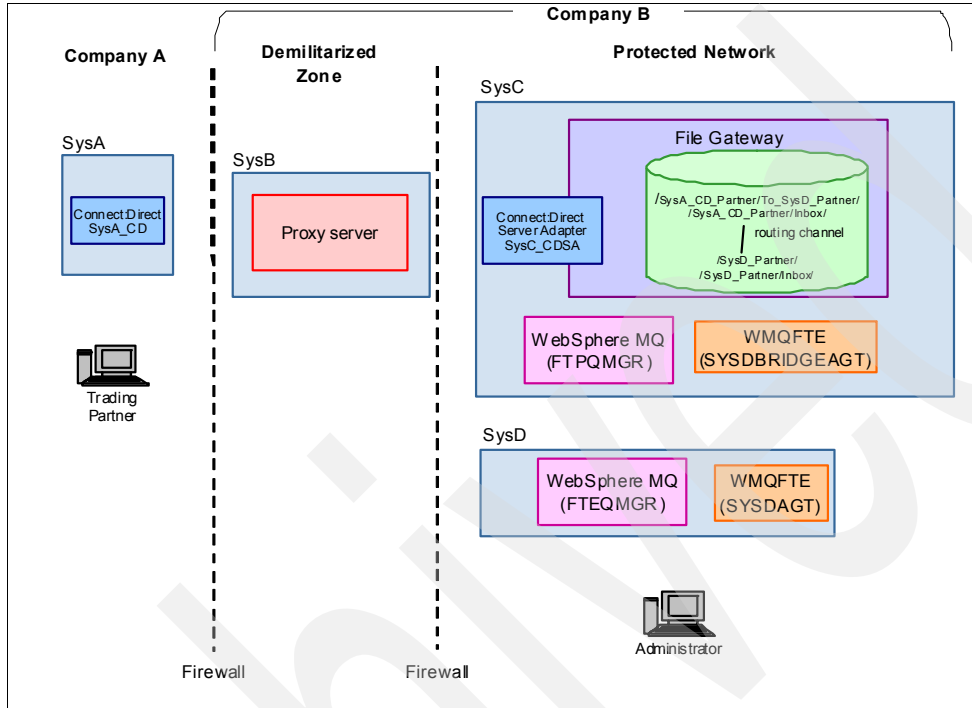


Figure 6-1 Solution components

Sterling Connect:Direct

Sterling Connect:Direct moves files point-to-point (peer-to-peer) using the Sterling Connect:Direct protocol. Sterling Connect:Direct offers the following benefits:

- ▶ Predictability

Assures delivery via automated scheduling, checkpoint restart, and automatic recovery/retry. If a data transmission is interrupted, the transmission tries to restart at a predefined interval for a configured amount of time. All activity and statistics are logged so that there are verifiable audit trails of all actions.

- ▶ Security

Ensures that customer information stays private through a proprietary protocol and offers basic security through authentication and user proxies. Supports a comprehensive cryptographic solution (Sterling Connect:Direct Secure Plus) that provides strong mutual authentication using X.509 certificates, SSL, and TLS data encryption, and data integrity checking. For more information about Sterling Connect:Direct Secure Plus Option and other products that enhance Sterling Connect:Direct's security model, see 2.1.1, "Sterling Connect:Direct additional features" on page 14.

- ▶ Performance

Handles the most demanding loads, from high volumes of small files to terabyte files.

Sterling Connect:Direct offers the following features:

- ▶ Provides automation through easy-to-use process definition and scripting. Multi-step processes manage data movement as well as pre-processing and post-processing.
- ▶ Provides automation through scripting, scheduling, and watch directories.
- ▶ Automatically establishes connection to remote server when data is ready for transfer. Automatic session retry re-establishes an interrupted connection. Work resumes at the point of failure.
- ▶ Offers flexible security options to control access to data, network, or system resources. Interfaces to operating system and vendor-supplied access control and security software.
- ▶ Supports a comprehensive cryptographic solution (Sterling Connect:Direct Secure Plus Option).
- ▶ Provides checkpoint/restart and automatic session retry.
- ▶ Supports local and remote administration, configuration, and process management through a browser user interface.
- ▶ Supports non-intrusive integration to existing applications through the command-line interface (CLI), which can be used in batch files or scripts. Also supports direct use by applications through APIs.
- ▶ Provides a complete audit trail of data movement through extensive statistics logs.
- ▶ Supports extensive configuration options for flexibility of deployment, management of network resources, and optimization of data transfer performance.
- ▶ Provides optional data compression that is configurable for maximum compression or optimal use of system resources.
- ▶ Supports all major file types, media, and record formats across multiple platforms. Data exchange is independent of content.

A Sterling Connect:Direct client is used to communicate with a Connect:Direct server regarding the work that will be performed. The Connect:Direct requester graphical user interface (GUI) and CLI are used to communicate with the Connect:Direct server.

Connect:Direct server adapter

The Connect:Direct server adapter is a component of Sterling B2B Integrator that allows Sterling B2B Integrator to act like a Connect:Direct server, sending and receiving data using the Sterling Connect:Direct protocol. This allows a remote Connect:Direct server to communicate directly with Sterling B2B Integrator and the Sterling File Gateway.

Proxy server

This can be any proxy server or other demilitarized zone hardened security mechanism. This security piece should validate that the external connection is coming from an approved domain over the port specified for the protocol used, terminate the external session, begin a secure session to continue back to the protected network, and authenticate users.

Sterling File Gateway

Sterling File Gateway routes files, incoming and outgoing, based on defined partners and their mailboxes. Sterling File Gateway utilizes Sterling B2B Integrator to switch protocols in this scenario. The mailboxes are created based on partner definitions. Sterling File Gateway can be administered through:

- ▶ Sterling File Gateway Administration Console (a web-based GUI) allows administrators to create partners, routs, and channel templates.
- ▶ Sterling B2B Integrator Administration Console (a web-based GUI) allows administrators to perform actions like creating business processes, defining trading partners, creating server adapters, and configuring protocols.
- ▶ Sterling B2B Integrator Import/Export Configuration allows administrators to configure one Sterling File Gateway instance that can be exported and then imported into another Sterling File Gateway instance.

In our scenario, we created additional mailboxes for each unique routing. Table 6-1 explains what the mailbox is for and to which mailbox it sends files.

Table 6-1 Sterling File Gateway mailboxes

External trading partner mailbox	Usage	Routing direction	Internal mailbox	Usage
/SysA_CD_Partner/ To_SysD_Partner	Receives a file from SysA_CD_Partner sent from Company A's SysA_CD Connect:Direct node. The routing channel is configured to route the file to SysD_Partner.	Inbound	/SysD_Partner/In box	The file is routed from the SysA_CD_Partner/To_SysD_Partner mailbox to SysD_Partner. The file is never placed in the SysD_Partner/Inbox mailbox though, as SysD_Partner has been configured to route the file using WebSphere MQ File Transfer Edition to machine SysD.
/SysD_Partner/	Receives a file from the Company B internal WebSphere MQ File Transfer Edition agent SYSDAGT running on machine SysD. This file is then routed to the external partner SysA_CD_Partner.	Outbound	/SysA_CD_Partn er/Inbox	The file is routed from SysD_Partner's root mailbox to partner SysA_CD_Partner. The file is never placed into the SysA_CD_Partner/Inbox mailbox, as SysA_CD_Partner has been configured to route the file using the Connect:Direct server adapter SysC_CDSA to send the file using Connect direct to the remote SysA_CD Connect:Direct node on Company A's SysA node.

Sterling B2B Integrator

Sterling B2B Integrator is a transaction engine and set of components designed to run processes that you define and manage according to your business needs. The suite supports high-volume electronic message exchange, complex routing, translation, and flexible interaction with multiple internal systems and external business partners.

Sterling File Gateway is a separate product that runs within Sterling B2B Integrator. The file routing and protocol switching functionality that is configured in Sterling File Gateway is

actually performed under the covers by Sterling B2B Integrator. Sterling B2B Integrator provides the following features and more:

- ▶ The ability to manage and grow trading partner communities
- ▶ Adapters for backend applications
- ▶ Role-based data access and system operation
- ▶ Data transport security and data encryption support
- ▶ Digital signature support
- ▶ Identity management, including authorization and authentication

For our scenario, we utilize Sterling B2B Integrator's ability to switch between the Sterling Connect:Direct protocol and WebSphere MQ File Transfer Edition. When switching to WebSphere MQ File Transfer Edition, Sterling B2B Integrator uses a business process that writes a WebSphere MQ command message, using the WebSphere MQ adapter in Sterling B2B Integrator, on a WebSphere MQ File Transfer Edition agent's command queue to initiate a file transfer. Once the transfer is initiated, Sterling B2B Integrator listens on a reply queue for status updates from WebSphere MQ File Transfer Edition.

WebSphere MQ Queue Manager (FTEQMGR)

FTEQMGR is the WebSphere MQ File Transfer Edition coordination queue manager. The coordination queue manager publishes status messages received from the agents showing the state of transfers and the agents' status. FTEQMGR also acts as the command and agent queue manager for the WebSphere MQ File Transfer Edition agent SYSDAGT.

WebSphere MQ Queue Manager (FTPQMGR)

FTPQMGR acts as the command and agent queue manager for the WebSphere MQ File Transfer Edition agent SYSDBRIDGEAGT. This queue manager must be local to the protocol bridge agent.

WebSphere MQ File Transfer Edition Bridge Agent (SYSDBRIDGEAGT)

SYSDBRIDGEAGT is a protocol bridge agent. This is a special agent that comes with WebSphere MQ File Transfer Edition Version 7.0.1 or later and requires a local queue manager. The protocol bridge agent cannot read or write a file to a file system without sending the file to a WebSphere MQ File Transfer Edition client or server agent.

SYSDBRIDGEAGT connects to a local queue manager in bindings mode.

SYSDBRIDGEAGT is configured to communicate with Sterling File Gateway using FTP. When Sterling File Gateway is ready to send a file over the MQ FTE backbone, it uses its WebSphere MQ adapter to put a message on SYSDBRIDGEAGT's command queue residing on FTPQMGR. This message instructs SYSDBRIDGEAGT to send the file received from Sterling File Gateway to SYSDAGT.

WebSphere MQ File Transfer Edition Server Agent (SYSDAGT)

SYSDAGT is the WebSphere MQ File Transfer Edition agent that connects to the local queue manager FTEQMGR in bindings mode. This type of agent is referred to as a server agent. SYSDAGT reads files from and writes files to the local file system. This agent is the target agent for the bridge agent for inbound scenarios and the source agent for outbound scenarios.

WebSphere MQ Explorer

The WebSphere MQ Explorer is used to view and administer the WebSphere MQ queue managers and queue manager objects such as queues, topics, and channels. WebSphere MQ Explorer is built on an Eclipse integrated development environment. The Eclipse-based platform allows plug-ins to be added to the base platform.

WebSphere MQ File Transfer Edition Explorer

The WebSphere MQ File Transfer Edition Explorer is a plug-in to the WebSphere MQ Explorer. It is used to schedule file transfer requests and view the status of current requests. The tool includes a Transfer Log view that subscribes to the coordination queue manager to obtain audit information. The audit information is displayed in the Transfer Log view for every transfer that occurs in the given topology. Beginning in WebSphere MQ File Transfer Edition Version 7.0.3, WebSphere MQ File Transfer Edition Explorer also includes the ability to view the status of an agent.

6.2.2 Inbound: Sterling Connect:Direct to WebSphere MQ File Transfer Edition

Figure 6-2 shows the flow for this scenario when files are inbound to the organization from an external partner. Note that we refer to our external partner as Company A. Company B is the other trading partner and is our lab environment.

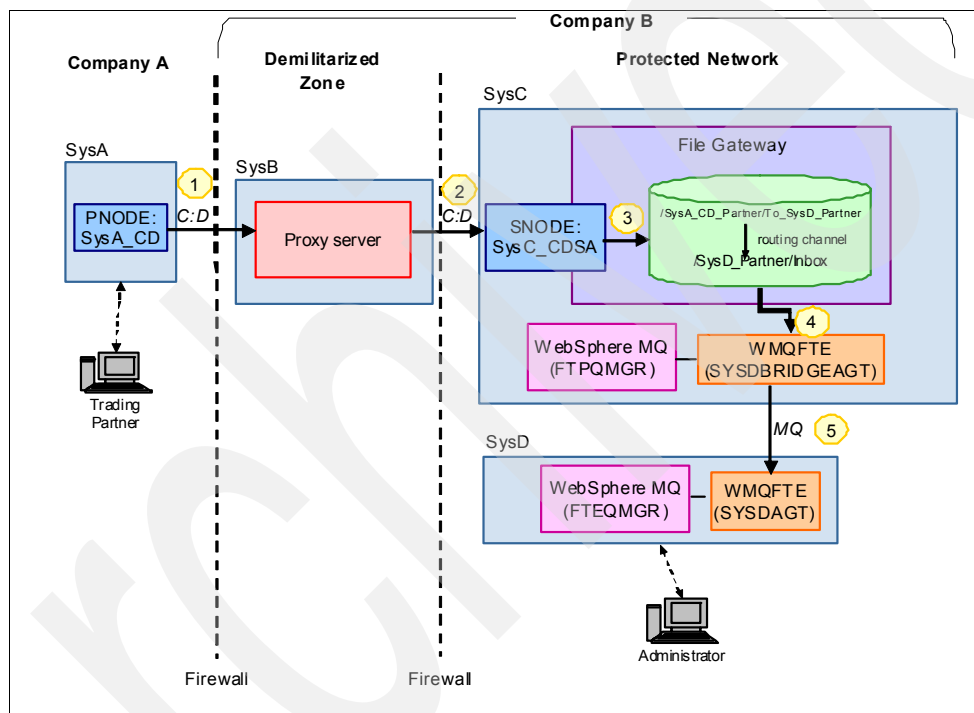


Figure 6-2 Inbound file transfer from Sterling Connect:Direct to WebSphere MQ file Transfer Edition

The following steps are performed in the inbound scenario:

1. The external partner, Company A, runs a script on SysA to initiate a Sterling Connect:Direct file transfer from the SysA_CD Connect:Direct node to the Connect:Direct server adapter node SysC_CDSA via the proxy server on SysB. The proxy server performs security checks and then passes the file to SysC.
2. The Connect:Direct server adapter node SysC_CDSA will receive the file from SysA_CD.
3. The file arrives in Sterling File Gateway in the mailbox for SysC_CDSA, /SysA_CD_Partner/To_SysD_Partner. Based on routing channels defined for the mailbox in Sterling File Gateway, the file is routed to partner SysD_Partner.
4. SysD_Partner is a listening consumer and is configured to automatically pass the file to the WebSphere MQ File Transfer Edition bridge agent, SYSDBRIDGEAGT, by placing a message on its command queue. SYSDBRIDGEAGT uses FTP to retrieve the file from Sterling File Gateway's proprietary file system.

5. SYSDBRIDGEAGT follows the instructions in the message placed on its command queue by Sterling File Gateway's WebSphere MQ adapter. The message tells SYSDBRIDGEAGT to transmit the file to SYSDAGT and place the file in a pre-configured location specified in Sterling File Gateway (c:\downloads).

6.2.3 Outbound: WebSphere MQ File Transfer Edition to Sterling Connect:Direct

Figure 6-3 shows the flow for this scenario when files are transferred outbound from the internal network to an external partner.

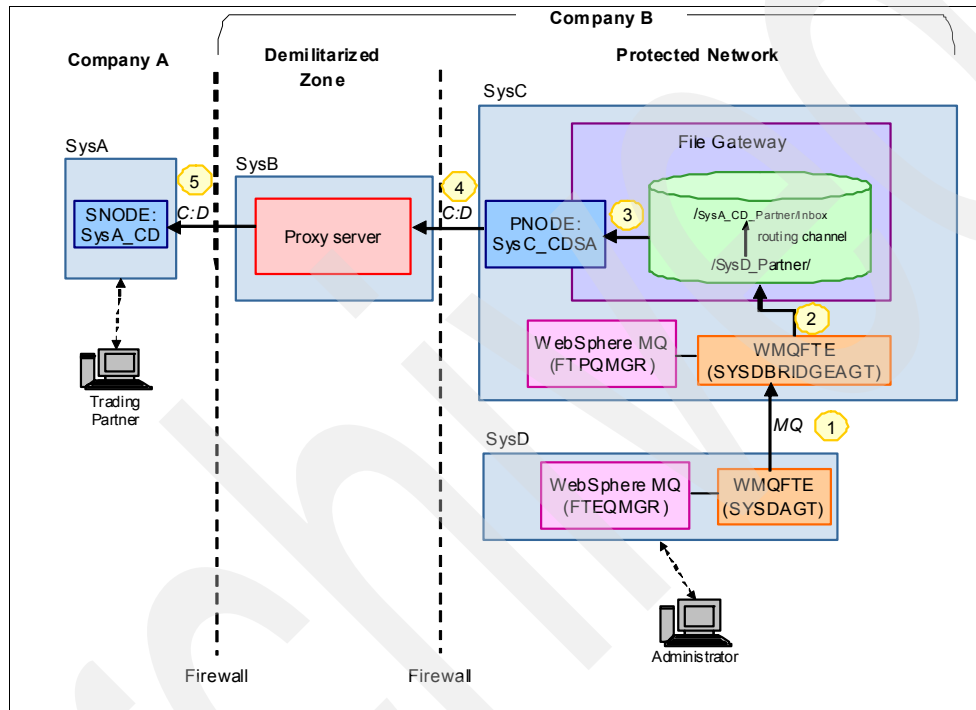


Figure 6-3 Outbound file transfer from WebSphere MQ file Transfer Edition to Sterling Connect:Direct

The following steps are performed in the outbound scenario:

1. An internal partner of Company B, SysD_Partner, initiates a file transfer using WebSphere MQ File Transfer Edition to send a file from the WebSphere MQ File Transfer Edition agent SYSDAGT to the WebSphere MQ File Transfer Edition bridge agent, SYSDBRIDGEAGT.
2. The file arrives in Sterling File Gateway in the mailbox /SysD_Partner/ (the root mailbox folder of SysD_Partner). Based on routing channels defined for the mailbox in Sterling File Gateway, the file is routed to the partner SysA_CD_Partner.
3. SysA_CD_Partner is configured as a listening consumer in Sterling File Gateway and is configured to route files to external partner SysA_CD_Partner using the Connect:Direct server adapter configured in Sterling B2B Integrator.
4. A file transfer is initiated from the Connect:Direct server adapter SysC_CDSA (the PNODE) to the remote Connect:Direct node SysA_CD (the SNODE) belonging to Company A. The Sterling Connect:Direct netmap in Sterling B2B Integrator, which is used by the Connect:Direct server adapter, routes Sterling Connect:Direct transfers via the secure proxy server on SysB. The proxy server performs security checks and then passes the file to SysA_CD.

5. SysA_CD receives the file from SysC_CDSA.

6.2.4 Protocols

This scenario uses the Sterling Connect:Direct protocol to transfer files between Company A, the external business partner, and Company B. These protocols are used to communicate with Sterling File Gateway in the Company B protected zone. The protocol used for integration between Sterling File Gateway and WebSphere MQ File Transfer Edition is FTP and WebSphere MQ for message-level integration. The WebSphere MQ File Transfer Edition backbone utilizes WebSphere MQ to move files from one location to the next.

6.2.5 Security

In this scenario, we are concerned with securing data during transmission over Sterling Connect:Direct and WebSphere MQ File Transfer Edition backbone.

Authorization of outside protocols, domains, ports, and key exchanges is handled by a proxy server in the Company B demilitarized zone. This could be any demilitarized zone hardened security mechanism, such as Sterling Secure Proxy, IBM WebSphere DataPower® B2B Appliance XB60, IBM HTTP Server, or another publicly available reverse proxy security server. In the Company B protected zone, Sterling B2B Integrator authenticates the partner user IDs.

Sterling Connect:Direct Security

To perform work in your enterprise, Sterling Connect:Direct relies on building blocks of information that define the local and remote nodes, users who can access those nodes, and the functions that they can perform.

Local node definition

During installation, you define a local node for Sterling Connect:Direct. The local node definition specifies information, such as the operating system, default user ID, TCP/IP address, and port number. After installation, you can change the local node's settings and define remote nodes. In addition to the default user ID that you specify for a local node, you can add other users who will access that node.

Local user authorities

After you define a user ID for each user who has access to the local node, you can restrict the ability of each user to perform certain tasks by defining user authorities for each user ID. For example, you can permit a user to submit a process but not to monitor or delete processes.

Sterling Connect:Direct has two types of users, administrators and general users, and each type has a set of default privileges. You can use these user templates to assign user authorities and restrict user privileges. Local user authorities provide one type of authentication in Sterling Connect:Direct. An alternative method of authentication is available using remote user proxies. For a listing of the default authorities for each type, see the product documentation for your Sterling Connect:Direct platform.

Remote user proxies

User proxy definitions (referred to as secure point of entry on the mainframe) contain remote user information for operations initiated from remote Connect:Direct nodes. These definitions identify a proxy relationship between a user ID at a remote Connect:Direct node and a local user ID. This mapping of remote and local user IDs enables users at remote Connect:Direct nodes to submit work to the local Connect:Direct node without explicitly

defining user IDs and passwords in the processes, eliminating the need to share passwords with your trading partners. User proxies also define what each user ID can do on the local Connect:Direct node.

Configuration settings for the local node

Initialization parameters determine various Sterling Connect:Direct settings that control system operation. The initialization parameters file is created when you install Sterling Connect:Direct and can be updated as needed. Some of these settings can be overwritten in the netmap, user authorities, user proxies, and processes.

Remote node definitions

The network map, or netmap, is a file created during the Sterling Connect:Direct installation that identifies the remote nodes that each local node can communicate with and the communication information needed to establish a connection. You create a remote node entry in the network map for each remote node that the local node communicates with. Each network map entry contains information about the remote node, such as the remote node name, operating system, session characteristics for a protocol, and transfer and protocol information about the available communications paths and their attributes.

Netmap checking

In addition to defining the remote nodes that communicate with the Connect:Direct node, the network map can be used to perform a security function. Netmap checking verifies that inbound sessions are from a node defined in the network map. If the node is not in the network map, the connection fails.

WebSphere MQ File Transfer Edition security

For any file transfer request, the agent processes require a certain level of access to its local file systems. In addition, both the user identifier associated with the agent process and the user identifiers associated with users performing file transfer operations must have the authority to use certain WebSphere MQ objects.

Commands are issued by users, who might be in operational roles in which they typically start file transfers. Alternatively, they might be in administrative roles, in which they can additionally control when agents are created, started, deleted, or cleaned (that is, when messages from all agent system queues are removed). Messages containing command requests are placed on an agent's SYSTEM.FTE.COMMAND queue when a user issues a command. The agent process retrieves messages containing command requests from the SYSTEM.FTE.COMMAND queue. The agent process also uses four other system queues, which are:

- ▶ SYSTEM.FTE.DATA.*agent_name*
- ▶ SYSTEM.FTE.EVENT.*agent_name*
- ▶ SYSTEM.FTE.REPLY.*agent_name*
- ▶ SYSTEM.FTE.STATE.*agent_name*

WebSphere MQ File Transfer Edition supports finer-grained checking of users' authorities, which permits access to be granted (or denied) to specific product functions for each user. For example, you can choose which users have the authority to schedule transfer operations to happen at a future time. Because users issuing commands use the above queues in different ways from the agent process, assign different WebSphere MQ authorities to the user identifiers or user groups associated with each. For more information, see *Using groups to manage authorities for resources specific to WebSphere File Transfer Edition* at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/group_resource_access.htm

The agent process has additional queues that can be used to grant users the authority to perform certain actions. The agent does not put or get messages on these queues. However, you must ensure that the queues are assigned the correct WebSphere MQ authorities, both for the user identifier used to run the agent process and for the user identifiers associated with users who are being authorized to perform certain actions. The authority queues are:

- ▶ SYSTEM.FTE.AUTHADM1.*agent_name*
- ▶ SYSTEM.FTE.AUTHAGT1.*agent_name*
- ▶ SYSTEM.FTE.AUTHMON1.*agent_name*
- ▶ SYSTEM.FTE.AUTHOPS1.*agent_name*
- ▶ SYSTEM.FTE.AUTHSCH1.*agent_name*
- ▶ SYSTEM.FTE.AUTHTRN1.*agent_name*

The agent process also publishes messages to the SYSTEM.FTE topic on the coordination queue manager using the SYSTEM.FTE queue. Depending on whether the agent process is in the role of the source agent or the destination agent, the agent process might require authority to read, write, update, and delete files.

You can create and modify authority records for WebSphere MQ objects using the WebSphere MQ Explorer. Right-click the object and then click **Object Authorities** → **Manage Authority Records**. You can also create authority records using the `setmqaut` command.

Instead of granting authority to individual users for all of the various objects that might be involved, configure two security groups for the purposes of administering WebSphere MQ File Transfer Edition access control:

- ▶ FTEUSER
- ▶ FTEAGENT

Additionally, WebSphere MQ File Transfer Edition has security features to protect the files and file systems from unauthorized access. These features allow organizations to control who can read and write files being transferred and how to protect the integrity of files. This security can be achieved by:

- ▶ Managing authorities to access file systems

The user ID running the agent process must have access to the local file system to read or write files during file transfer. This is controlled through the local operating system.

- ▶ Using sandboxes

The access of an agent to the file system can be restricted by defining the `sandboxRoot` property in an agent's properties file. This property restricts the agent's access to a certain directory or a certain area of the file system, the so-called sandbox. For more information about sandboxing, see:

<http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/sandboxes.htm>.

- ▶ Using the agent's `commandPath` property

The `commandPath` property in an agent's property file restricts the locations that an agent can run commands from. By default, the `commandPath` is empty, so an agent cannot call any commands. Take extreme care when this property is set because any command in one of the specified `commandPath` settings can be called from a remote client system that is able to send commands to the agent. For more information about this property, see:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/command_path.htm

- ▶ **Configuring SSL encryption for WebSphere MQ File Transfer Edition**
The file data, when transferred between WebSphere MQ File Transfer Edition agents, can be protected by establishing Secure Sockets Layer (SSL) on the WebSphere MQ channel connections.
- ▶ **Authority to publish log and status messages**
Agents issue various log, progress, and status messages to the coordination queue manager for publication. The publication of these messages can be secured using WebSphere MQ security.
- ▶ **Using the MD5 checksum**
This is the default setting on file transfers in WebSphere MQ File Transfer Edition. This is set to keep files from being manipulated after the transfer has been initiated or while the transfer is in progress.
- ▶ **Authentication**
Client FTE agents must be authenticated. The authentication can be performed by WebSphere MQ using SSL or exits.
- ▶ **Bridge agent credentials**
The bridge agent user must be authenticated when the bridge agent connects to a FTP server, in our scenario Sterling File Gateway FTP server adapter. The authentication of the bridge agent user at the FTP server can be done based on user ID and password credentials or by using a public/private key pair.

Firewall security

Firewall configuration plays an important role in securing your connections to and from external partners and in protecting the internal network.

The external firewall must allow incoming requests from all of the trading partner's source IP addresses or range of IP addresses. This can be configured in the firewall rules configuration. The method used to configure firewall rules depends on the model and type of firewall being used.

The demilitarized zone (DMZ) is a termination point at the edge of the protected network and typically is used to house internet-facing systems.

Setting up tight rules on the inner firewall is important for protecting your internal systems. Typically, inner firewall rules are set up to allow only traffic from a mediation server in the DMZ that terminates the connection from the internet. The mediation server then re-establishes the connection through the inner firewall to a system that the files are destined for or to a system that moves the files to a back-end application.

For this scenario, we need to open a Sterling Connect:Direct port, 1354, both inside and outside. These ports vary based on your mediation server configuration in your demilitarized zone and your organization's security policies and practices.

6.3 Configuring the solution components

This section describes the detailed technical steps involved in implementing the above scenarios into an environment with Sterling File Gateway, Sterling B2B Integrator, and WebSphere MQ File Transfer Edition.

6.3.1 Software prerequisites

This scenario uses the following software:

- ▶ WebSphere MQ V7.0.1
- ▶ WebSphere MQ File Transfer Edition V7.0.3
- ▶ Sterling File Gateway 2.1
- ▶ Sterling B2B Integrator 5.1.01
- ▶ DB2® Version 9.5
- ▶ Web browser

6.3.2 Configuration prerequisites

The following assumptions have been made for this chapter:

- ▶ You have already installed Sterling B2B Integrator and Sterling File Gateway.
- ▶ You have installed a suitable proxy server in your DMZ to validate external partners and authenticate users. This should also be set up to end the external session and begin an internal session, directing the data back to the protected network.
- ▶ Ports opened in the firewall (internal and external): 1364 for Sterling Connect:Direct.
- ▶ You have a Connect:Direct node installed external to your network (in the untrusted zone).
- ▶ You have WebSphere MQ installed on the same machine as Sterling B2B Integrator and Sterling File Gateway.
- ▶ You have WebSphere MQ File Transfer Edition installed on the same machine as Sterling B2B Integrator and Sterling File Gateway.
- ▶ WebSphere MQ queue manager FTPQMGR. This is shown in “Creating the queue managers” on page 349.

Security prerequisites: Review your local security policy and practices to determine what is appropriate for your production environment. While the scenarios in this book do not implement security, it is important that you take security into consideration when implementing these scenarios in your own environment.

6.3.3 Sterling B2B Integrator and Sterling File Gateway customization

To enable Sterling File Gateway to delivery files via the WebSphere MQ File Transfer Edition network, a custom protocol implementation must be added to Sterling File Gateway. This section describes the files used in the customization and how they work.

Note: For general information about adding custom protocols to Sterling File Gateway see:

- ▶ **System Administration** → **Extend the Capabilities** → **Add Custom Protocols** within the Sterling File Gateway online documentation at:

<http://help.sterlingcommerce.com/SFG21/index.jsp>

- ▶ The contents of the `samples/filegateway/protocol_extensions` directory within a Sterling File Gateway installation

The following components are required to implement the WebSphere MQ File Transfer Edition custom protocol within Sterling File Gateway:

- ▶ `AFTExtensionsCustomer.xml`

This file contains one or more AFTEExtension elements, each defining a custom protocol. Each AFTEExtension element defines the name of the custom protocol, the business process that implements the custom protocol, and the parameters that will be exposed in the Sterling File Gateway user interface when configuring a partner using the custom protocol.

▶ AFTEExtensionsCustomer.properties

This file contains the text strings that will be displayed in the Sterling File Gateway user interface when configuring a partner using a custom protocol. Each label attribute in the AFTEExtensionsCustomer.xml file must have a matching entry in this property file, mapping the label name with the text string to be displayed in the user interface.

▶ CustomFileGatewayDeliveryFTE.bmp1

This file contains the implementation of the business process that creates a transfer request on the WebSphere MQ File Transfer Edition. The name of this business process must match the bp attribute of the corresponding AFTEExtension element in the AFTEExtensionsCustomer.xml file. When invoked to deliver files for a given partner, the partner's values for each parameter defined in AFTEExtensionsCustomer.xml will be passed to the business process. The business process utilizes a number of existing Sterling B2B Integrator adapters and services, particularly the XSLT translation service and the WebSphere MQ Suite adapters. The business process implements the following logic:

- It creates a new mailbox message to be accessed by the WebSphere MQ File Transfer Edition source agent. This message can be extracted multiple times to allow for transfer recovery as necessary and will be deleted when the transfer has completed.
- It creates the XML document required to request a transfer on the WebSphere MQ File Transfer Edition. It uses an XSLT stylesheet to populate the document with parameters specific to the current transfer request.
- It creates a message containing the XML transfer request and puts it on the WebSphere MQ File Transfer Edition source agent's command queue.
- Optional: It waits to receive a message on a reply queue to determine the final disposition of the transfer on the WebSphere MQ File Transfer Edition network.

▶ SFGFTECreateTransfer.xslt

This XSLT stylesheet is used by the CustomFileGatewayDeliveryFTE business process to build the XML document requesting a transfer, incorporating parameters specific to the current file and partner. The schema for the managedTransfer element that it creates can be found in the samples/schema directory of a WebSphere MQ File Transfer Edition installation directory.

Table 6-2 lists the parameters collected when configuring a Sterling File Gateway partner to use the WebSphere MQ File Transfer Edition protocol.

Table 6-2 Parameters used by a WebSphere MQ File Transfer Edition partner

Parameter name	Values/use
Source agent name (-sa)	Source agent name (name of local FTP Bridge Agent).
Source agent queue manager (-sm)	Source agent's queue manager.
Source agent queue manager host name	Host or IP of source agent queue manager (required only if connecting to WebSphere MQ in client mode).
Source agent queue manager port	TCP port of source agent queue manager (required only if connecting to WebSphere MQ in client mode).

Parameter name	Values/use
Source agent queue manager user ID	User ID used when connecting to source agent's queue manager.
Source agent queue manager password	Password used when connecting to source agent's queue manager.
Destination agent name (-da)	Destination agent name.
Destination agent queue manager (-dm)	Destination agent queue manager.
Destination agent's directory (-dd)	Destination directory.
Destination file already exists (-de)	Disposition when file exists on destination agent (error or overwrite).
Queue for transfer status reply messages	Name of queue on source agent's queue manager used to contain transfer status reply messages. If blank, Sterling File Gateway will consider a delivery successful if request message is successfully added to source agent's command queue. If populated, Sterling File Gateway will look for reply messages on this queue to determine whether the transfer completed or failed.
Priority (-pr)	Priority of transfer (0 - 9).
Conversion (-t)	File conversion performed during transfer (binary or text).
Checksum method (-cs)	Checksum method computed by agents (MD5 or none).
Transfer timeout (seconds)	Number of seconds Sterling File Gateway will wait for a reply from source agent to determine whether the transfer completed. If no reply is received before timeout occurs, Sterling File Gateway marks delivery as failed <i>even though the transfer might still be pending in WebSphere MQ File Transfer Edition and will eventually complete.</i>

These files can be downloaded from the IBM Redbooks web server. For information about downloading the web material for this book, see Appendix E, "Additional material" on page 425. The steps for installing and configuring the customization are covered in this chapter.

6.3.4 Configuring Sterling Connect:Direct on SysA

If you have not already configured the external Connect:Direct node SysA_CD (as described in 5.2.10, "Configuring Sterling Connect:Direct on SysA" on page 104), then follow the steps below.

The following sections describe how to configure the external Connect:Direct node and the Connect:Direct server adapter on the Sterling File Gateway/Sterling B2B Integrator machine so that they are aware of each other. To simplify the examples, both nodes use the same operating system user ID. The external Connect:Direct node SysA_CD uses the Sterling Connect:Direct preset upload and download directories for transferred files. Preset upload and download directories enable a process to be submitted without using full path and file names. Given only a file name, the node will push or pull any file from the preset directory. The procedure to configure preset upload and download directories is described below.

Before the Connect:Direct nodes are configured, add a new user named cdadmin on SysA_CD. The cdadmin user ID needs only to be given basic user rights. The cdadmin user ID will be used for all the Sterling Connect:Direct transfers in the scenarios below.

1. On SysA select **Start** → **All Programs** → **Sterling Commerce Sterling Connect:Direct v4.5.01** → **CD Requester**.
2. Expand the entry for **SysA_CD** and double-click **Netmap** (Figure 6-4).

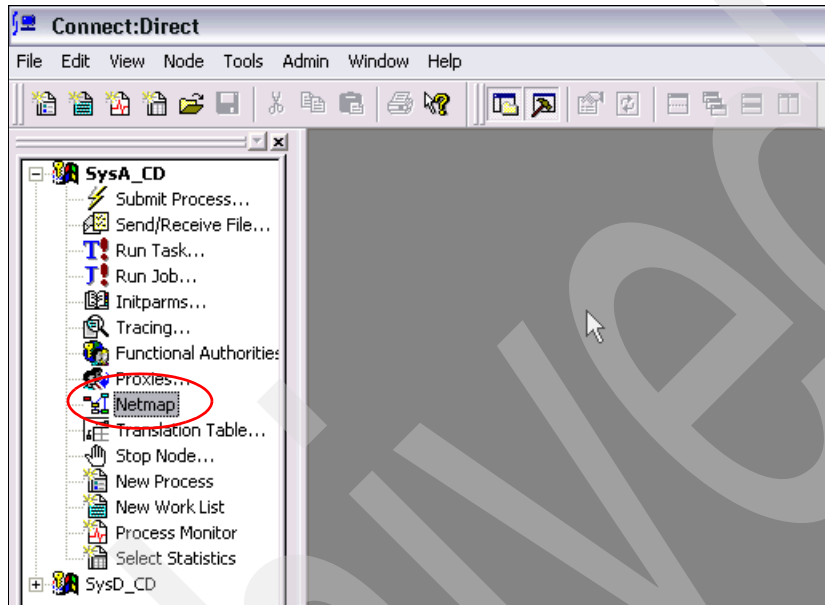


Figure 6-4 Launching the Connect:Direct Requester netmap configuration

3. From the menu, select **Netmap** → **Insert** (Figure 6-5).

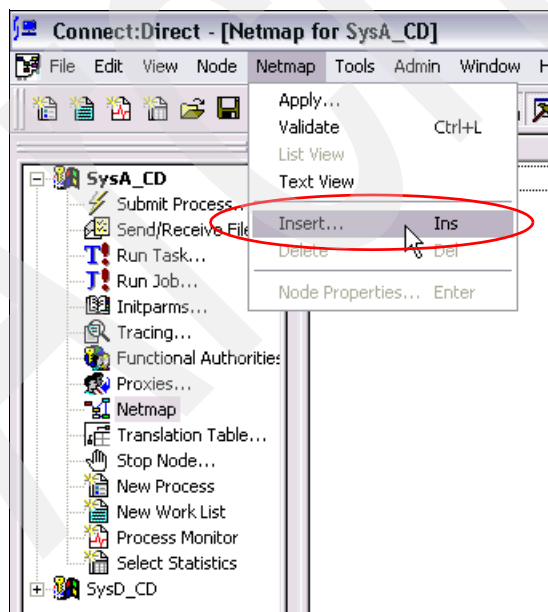


Figure 6-5 Inserting a new netmap entry

- Under the Main tab ensure that the values shown in Figure 6-6 are filled in.

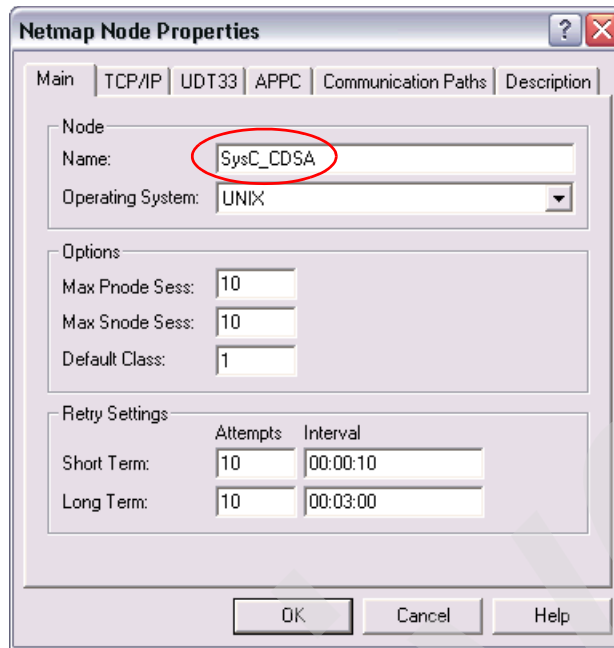


Figure 6-6 netmap Main tab

- Under the TCP/IP tab, ensure that the values shown in Figure 6-7 are filled in.

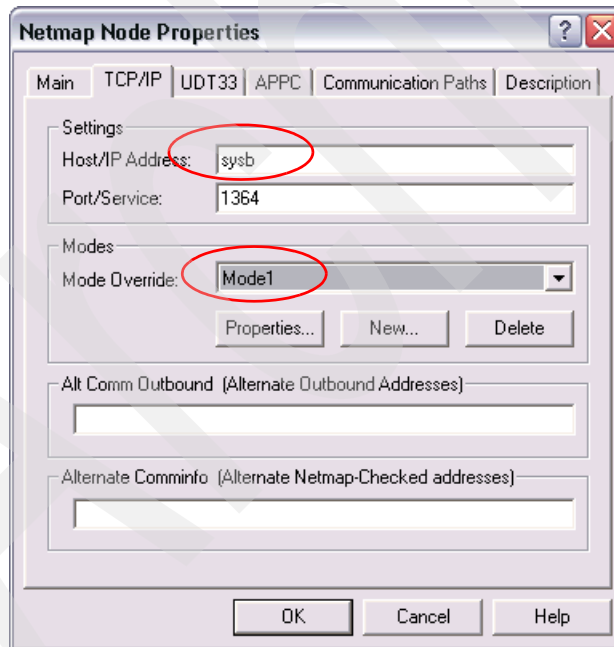


Figure 6-7 netmap TCP/IP tab

- Under the Communications Paths tab, select and highlight **TCPCommPath**.

7. Click the right arrow to add TCPCommPath to the Selected Path box (Figure 6-8).

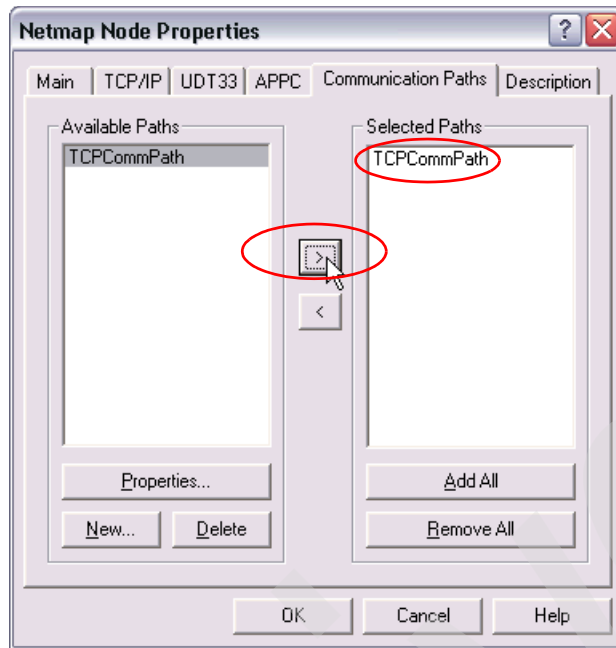


Figure 6-8 netmap Communications Paths tab

8. Click **OK** to stage the new entry.
9. Back on the main Netmap panel, select **Netmap** → **Apply** (Figure 6-9).

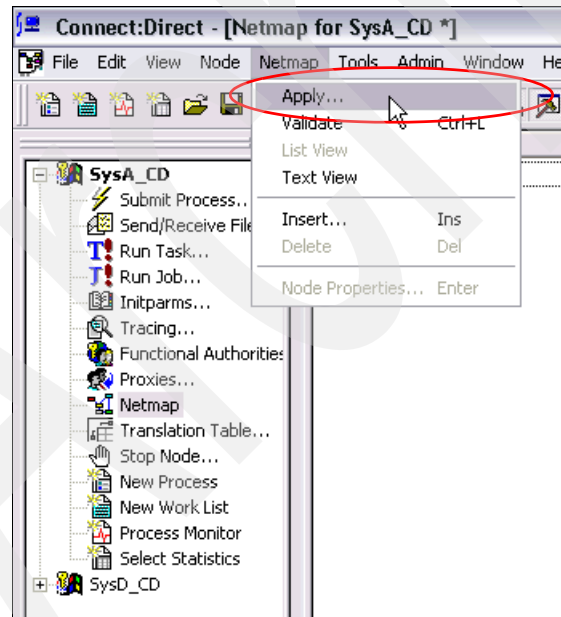


Figure 6-9 Applying a new netmap entry

10..If prompted to select a Connect:Direct node to apply the changes to, select **SysA_CD** (Figure 6-10).

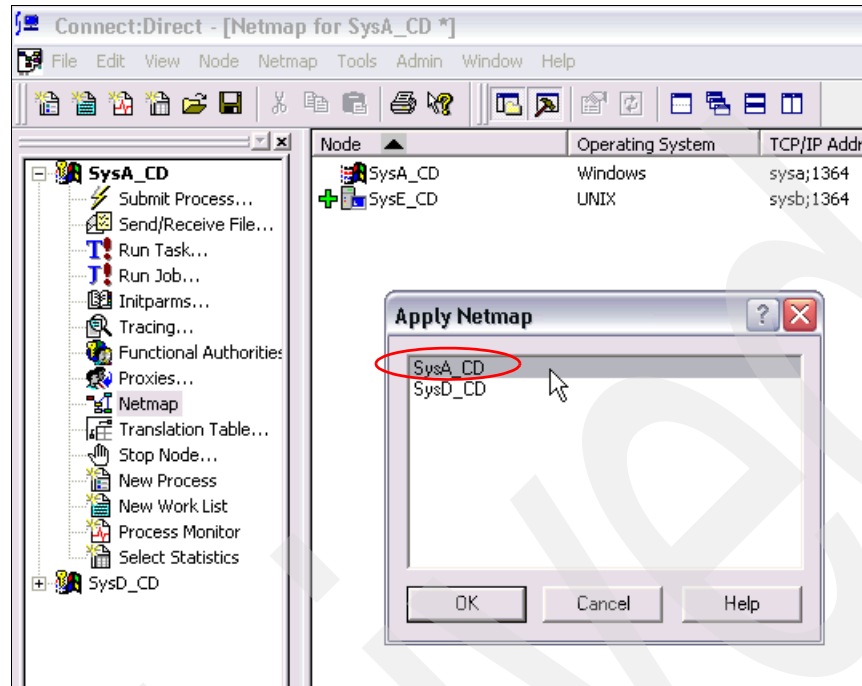


Figure 6-10 Applying a new netmap entry to multiple nodes

11. Create two directories on SysA_CD where Sterling Connect:Direct for Microsoft Windows is installed. Ensure that the cdadmin user ID has operating system permissions to read and write to these two directories:

- C:\CDWindows_files\upload
- C:\CDWindows_files\download

12. Expand the entry for **SysA_CD** and double-click **Functional Authorities** (Figure 6-11).

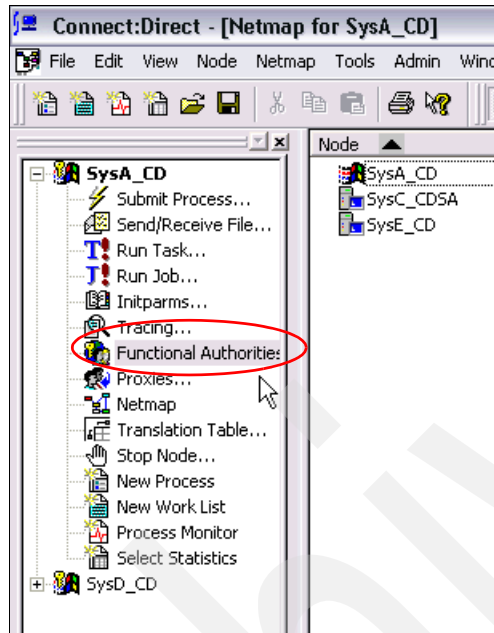


Figure 6-11 Launching the Sterling Connect:Direct Requester Functional Authorities configuration

13. In the Functional Authorities dialog box, click **New Admin** (Figure 6-12).

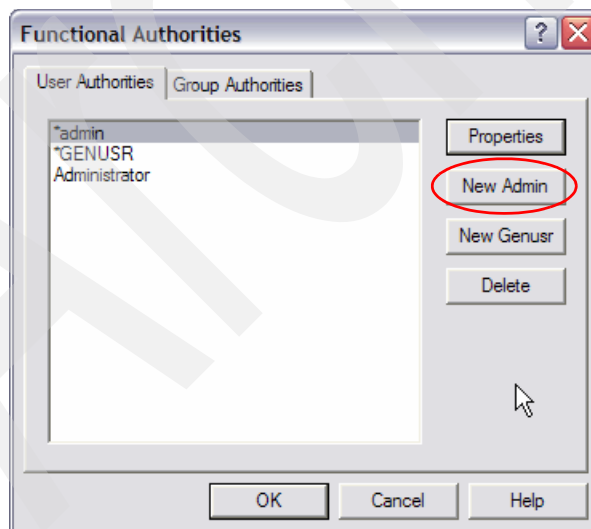


Figure 6-12 Functional Authorities User tab

14. Under the Main tab enter cdadmin in the Name field (Figure 6-13).

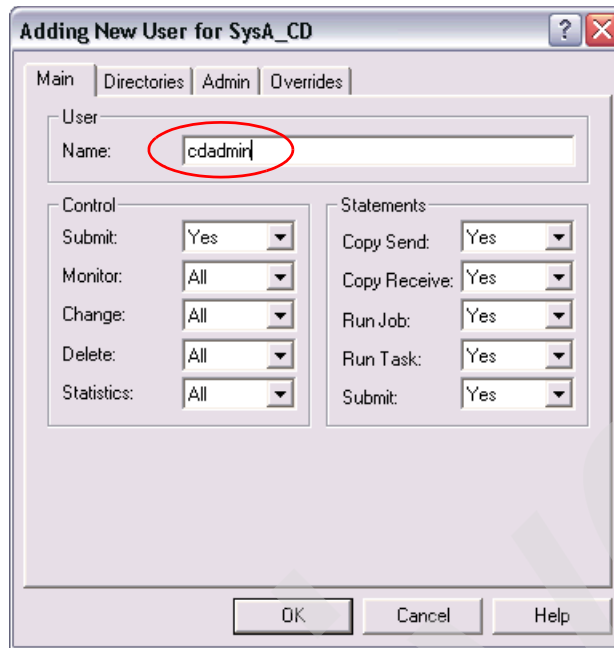


Figure 6-13 Functional Authorities new user Main tab

15. On the Directories tab, enter the upload and download directories created on SysA_CD in step 11 on page 189 (Figure 6-14).

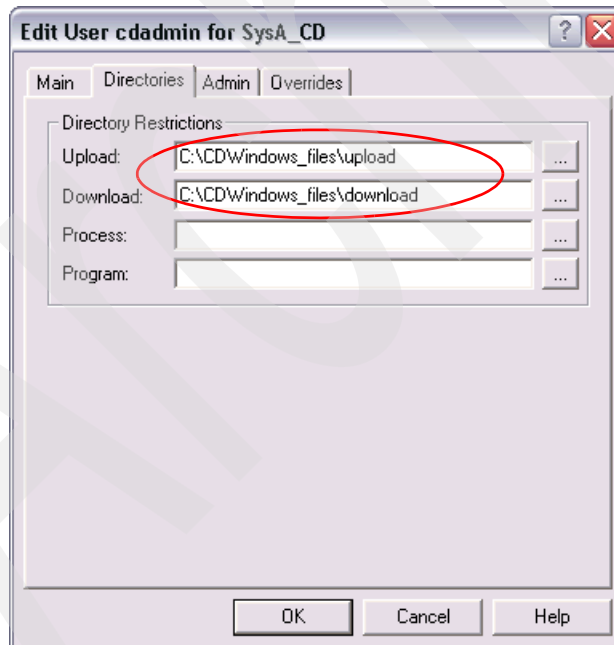


Figure 6-14 Functional Authorities new user Directories tab

16. Click **OK** to save your changes.

17. Expand the entry for **SysA_CD** and double-click **Proxies** (Figure 6-15).

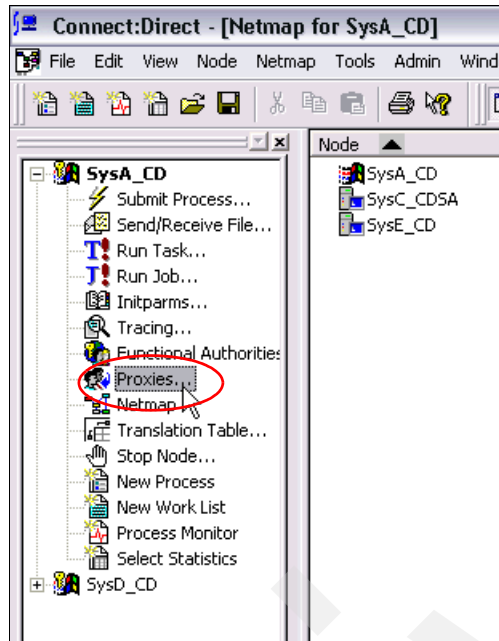


Figure 6-15 Launching the Sterling Connect:Direct Requester Proxies configuration

18. Click **Insert** (Figure 6-16).

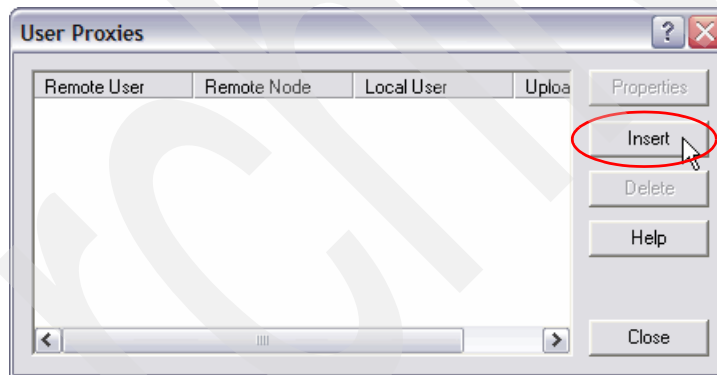


Figure 6-16 User Proxies dialog

19. Under the Main tab, enter the text as shown in Figure 6-17. All fields must be populated.

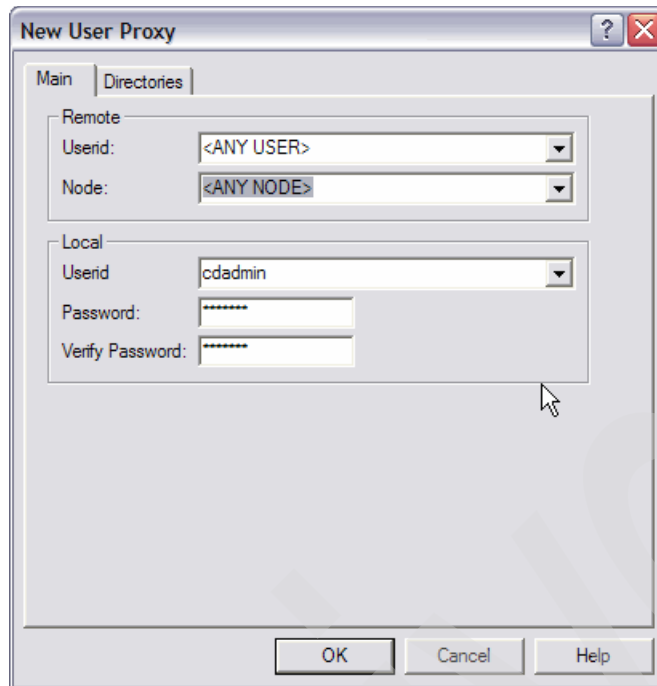


Figure 6-17 User Proxies new user Main tab

20. On the Directories tab, change the copy send and copy receive permissions to **Yes** (Figure 6-19 on page 194). Enter the upload and download directories created on SysA_CD in step 11 on page 189.

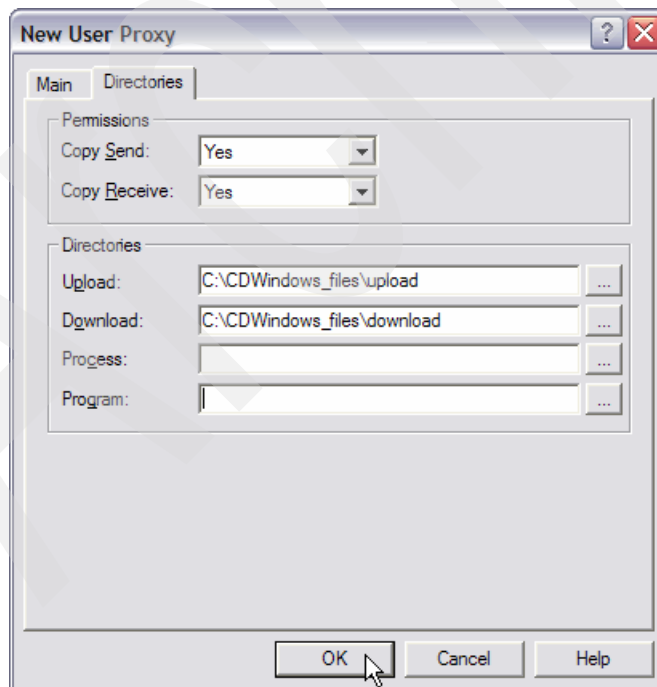


Figure 6-18 User proxies new user Directories tab

6.3.5 Installing the Connect:Direct server adapter on Sterling B2B Integrator

To use Sterling Connect:Direct to perform file transfers in Sterling B2B Integrator and the Sterling File Gateway, we need to install and configure a Connect:Direct server adapter (CDSA) in Sterling B2B Integrator.

If you have not already configured the Connect:Direct server adapter, follow the steps in 5.2.12, “Installing and configuring the Connect:Direct server adapter” on page 116, including the steps to create the netmap in Sterling B2B Integrator.

6.3.6 Configuring the proxy

The netmap on Sterling B2B Integrator has already been configured to point to the proxy node on SysB instead of going directly to SysA. The Sterling Connect:Direct netmap for SysA_CD on SysA has also been configured to point to SysC_CDSA on the SysB node instead of going directly to SysC.

The chosen proxy server used in your scenario needs to be configured to route requests from the Connect:Direct node SysA_CD on SysA to the Connect:Direct server adapter SysC_CDSA on SysC and vice-versa.

6.3.7 Configuring Sterling B2B Integrator to use the WebSphere MQ Adapter

The custom business process to start a WebSphere MQ File Transfer Edition file transfer will write a file transfer request message to the WebSphere MQ File Transfer Edition agent’s (SYSDBRIDGEAGT) command queue. To write the message to the WebSphere MQ queue you need to install the WebSphere MQ Adapter into Sterling B2B Integrator.

As you should already have WebSphere MQ installed, you have access to the required IBM WebSphere MQ library packages. To install the WebSphere MQ Adapter into Sterling B2B Integrator, follow these steps:

1. Copy all the jar files in the `mq_install_dir/java/lib` directory (a default WebSphere MQ installation would mean that they are in `C:\Program Files\IBM\WebSphere MQ\java\lib`) to a temporary directory on your Sterling B2B Integrator machine (for example, `c:\temp\mq`).
2. Stop Sterling B2B Integrator if it is running by issuing the command:
`<SI_install_dir>/bin/stopWindowsService.cmd`
3. From the bin directory of the Sterling B2B Integrator installation (`<SI_install_dir>/bin`), install all the vendor library packages by running the `install3rdParty` script included with Sterling B2B Integrator by entering the following commands:

```
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.commonservices.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.defaultconfig.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.fta.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.headers.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.jmqi.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.jms.Nojndi.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.pcf.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.postcard.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.soap.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mq.tools.ras.jar
install3rdParty.cmd ibm 7_0 -j c:/temp/mq/com.ibm.mqetclient.jar
```

- Restart Sterling B2B Integrator by double-clicking the desktop icon **Sterling_Integrator_at_8080**.

6.3.8 Configuring FTP Server adapter in Sterling B2B Integrator

The Sterling B2B Integrator FTP Server adapter receives and processes messages from partners that are submitted using the FTP protocol.

In the customization to Sterling B2B Integrator for WebSphere MQ File Transfer Edition, the FTP Server adapter is used to transfer files between the mailboxes of the partners configured in Sterling File Gateway and the WebSphere MQ File Transfer Edition agents.

Sterling B2B Integrator installs an FTP Server adapter by default, so you just need to start it. To start the FTP Server adapter, follow these steps:

- Start Sterling B2B Integrator and Sterling File Gateway if it is not already running by double-clicking the desktop icon **Sterling_Integrator_at_8080**.
- Start Internet Explorer and go to:
`http://<servername>:<port>/filegateway/`
- Log in using the Sterling File Gateway administrator user ID and password (Figure 6-19). The default administrator userid is `fg_sysadmin`.

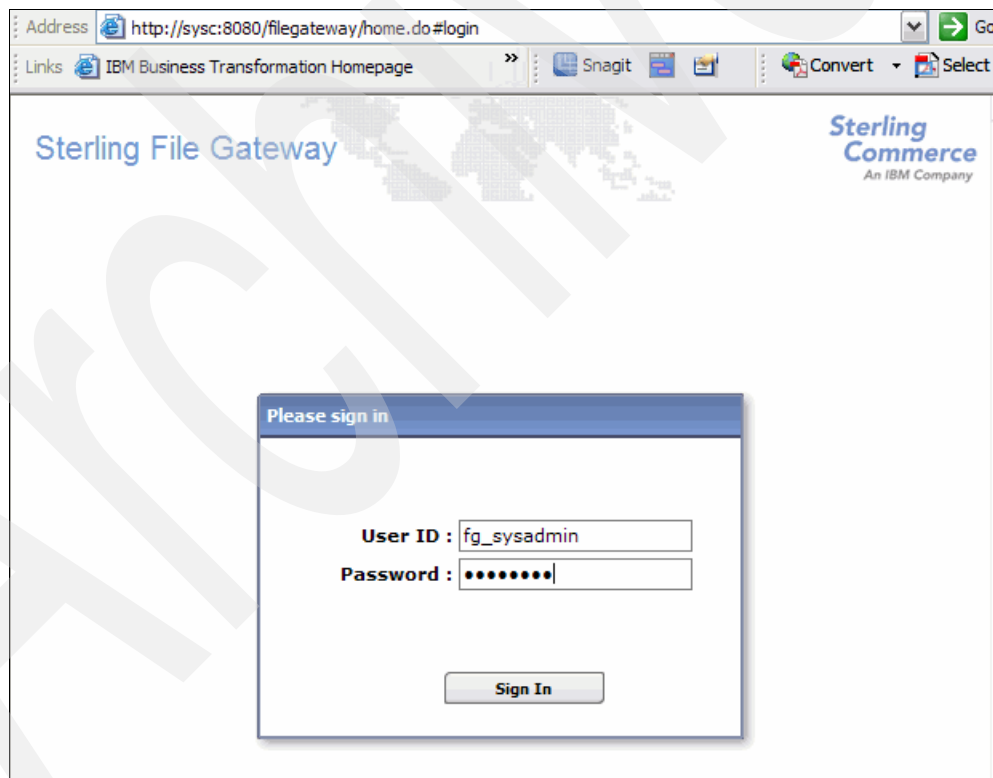


Figure 6-19 Sterling File Gateway login screen

- In Sterling File Gateway, go to **Tools** → **B2B Console** (Figure 6-20). This brings up the Sterling B2B Integrator console in a new browser window.



Figure 6-20 Open the Sterling B2B Integrator browser window

- Go to **Deployment** → **Services** → **Configuration** and enter FTP Server Adapter into the Service Name field. Click **Go!** (Figure 6-21).

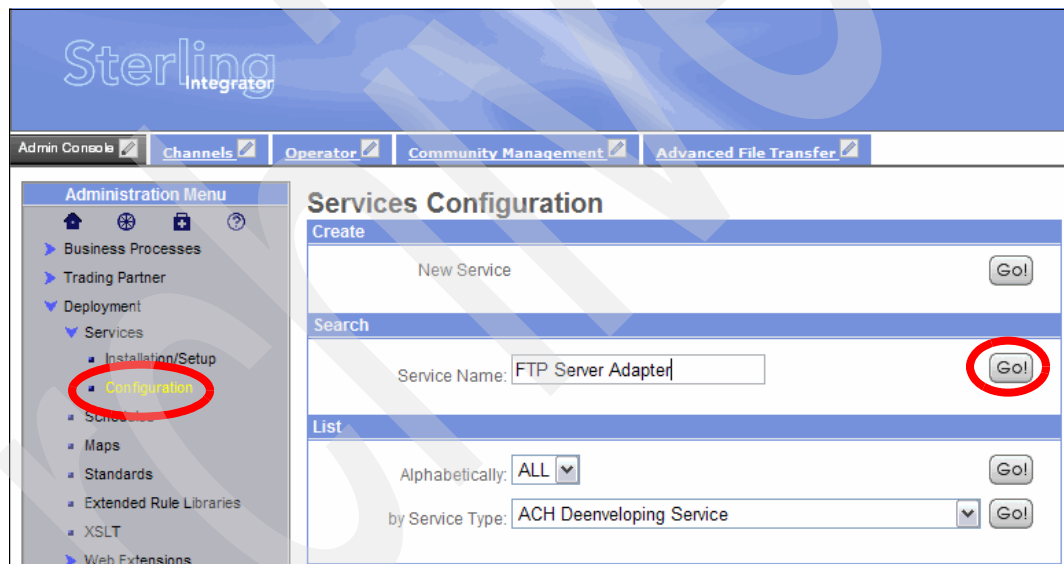


Figure 6-21 Search for existing FTP Server Adapter

- Click **FTP Server Adapter** to determine its listening port number (Figure 6-22).

The screenshot shows a table titled 'Services Configuration' with the subtitle 'Services 1-2 of 2'. The table has columns: Enabled, Select, Name, Type, and Advanced State. The 'Name' column contains 'FTP Server Adapter' and 'SFTP Server Adapter', with 'FTP Server Adapter' circled in red. The 'Type' column contains 'FTP Server Adapter' and 'SFTP Server Adapter'. The 'Advanced State' column contains 'Stopping' and 'Stopped'.

Enabled	Select	Name ▲▼	Type ▲▼	Advanced State
<input type="checkbox"/>		FTP Server Adapter	FTP Server Adapter	Stopping
<input type="checkbox"/>		SFTP Server Adapter	SFTP Server Adapter	Stopped

Figure 6-22 List of FTP Server adapters installed in Sterling B2B Integrator

- The configuration details are displayed. As shown in Figure 6-23, the listening port in our environment is 8112. Make a note of the FTP port number in your environment.

▶ **FTP Server Adapter**

Service Settings

Service Type	FTP Server Adapter
Description	FTP Server Adapter
System Name	FTP_SERVER_ADAPTER
Group Name	None
FTP Server Listen Port	8112
Active Data Port Range	None provided
Passive Data Port Range	None provided
Perimeter Server	Node1 local
Transfer Buffer Size (bytes)	32000
Minimum Number of Threads	3
Maximum Number of Threads	6

Figure 6-23 FTP listen port

- Close the details window, and click the **Enabled** check-box for the FTP Server Adapter to start the service. The state should change to Enabled (Figure 6-24).

Services Configuration

Services 1-2 of 2

Enabled	Select	Name ▲▼	Type ▲▼	Advanced State
<input checked="" type="checkbox"/>		FTP Server Adapter	FTP Server Adapter	Enabled
<input type="checkbox"/>		SFTP Server Adapter	SFTP Server Adapter	Stopped

Figure 6-24 The FTP Server adapter is now running

- The FTP Server adapter is now started and ready to begin accepting requests for file transfers.

6.3.9 Configuring WebSphere MQ File Transfer Edition bridge agent

This section describes how to create and configure the SYSDBRIDGEAGT agent on the Sterling B2B Integrator server in the protected network. We assume that the queue manager FTPQMGR is already implemented. The bridge agent requires a local agent queue manager to which to connect. In our scenario, the SYSDBRIDGEAGT agent is created on the same machine on which the queue manager FTPQMGR resides.

- Create the agent using the **fteCreateBridgeAgent** command.

Open a command console and run the following command from the command line:

```
fteCreateBridgeAgent -agentName SYSDBRIDGEAGT -agentQMGR FTPQMGR -bt FTP -bh
sysc -btz US/Eastern -bm UNIX -bsl en_US -bfe UTF8 -bp 8112
```

This command creates the bridge agent in bindings mode to the FTPQMGR queue manager and is connected to a FTP server adapter in Sterling B2B Integrator on SysC. Note that the FTP port number is set to 8112. This is the port number that the FTP server adapter is listening on in Sterling B2B Integrator.

The parameters used in this command are:

- agentname: Name of the agent
- agentQMGr: Name of the agent queue manager
- bt: Protocol type (FTP or SFTP)
- bh: Host name or IP address of the FTP server machine
- bp: Port of the FTP server adapter
- btz: (Optional) FTP server time zone
- bm: FTP server platform
- bs1: Locale of the FTP server adapter
- bfe: FTP server encoding

Note that although Sterling B2B Integrator is installed on a Microsoft Windows operating system, the FTP server adapter in Sterling B2B Integrator behaves as if it were a UNIX FTP server, so when creating the bridge agent we need to specify the platform as UNIX.

For more information: For a detailed description of the `fteCreateBridgeAgent` command, see the WebSphere MQ File Transfer Edition 7.0.3 Information Center at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/index.jsp?topic=/com.ibm.wmqfte.home.doc/help_home_wmqfte.htmfirst

The `fteCreateBridgeAgent` command also creates three files. Two MQSC script files are created with the commands required to define and to delete the agent's system queues. It also creates a credential XML file that you must modify in a subsequent step. Information about these files is shown in the command output in Example 6-1.

Example 6-1 Results of the fteCreateBridgeAgent command

```
BFGCL0277I: A credential XML file has been created. This file must be
completed with credential details for accessing the protocol file server before
the bridge agent can be brought into service. The file can be found here:
'C:\Documents and Settings\All Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDBRIDGEAGT\ProtocolBridgeCredentials.x
ml'.
.....
.....
BFGCL0069I: A file has been created containing the MQSC definitions to create
your agent. The file can be found here: 'C:\Documents and Settings\All
Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDBRIDGEAGT\SYSDBRIDGEAGT_create.mqsc'.
BFGCL0070I: A file has been created containing the MQSC definitions to delete
your agent. The file can be found here: 'C:\Documents and Settings\All
Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDBRIDGEAGT\SYSDBRIDGEAGT_delete.mqsc'.
BFGCL0053I: Agent configured and registered successfully.
```

Make sure that at the end of the output you see that the agent is successfully registered.

If you see a message that the agent was configured but could not be registered, this means that the coordination queue manager could not be contacted because it is not available or your configuration parameters are not correct.

The effect is that the agent can be started and transfer files, but that it is not listed by the `ftelListAgents` command or in the WebSphere MQ File Transfer Edition Explorer. The status messages of this agent are also not shown in the WebSphere MQ File Transfer Edition Explorer Transfer Log view.

The WebSphere MQ reason code issued with the error provides more information about the reason for the problem. Explanations for reason codes can be found in the WebSphere MQ V7 Information Center at:

<http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>

Creating the bridge agents: The `fteCreateBridgeAgent` command creates a bridge agent for a specific FTP server. You have to create a bridge agent for each FTP server to which you want to connect.

2. Create the bridge agent's MQ objects on the queue manager, FTPQMGR, using the script generated in the previous step. Run the `SYSDBRIDGEAGT_create.mqsc` script from the command line using the `runmqsc` utility:

```
runmqsc FTPQMGR < C:\Documents and Settings\All Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDBRIDGEAGT\SYSDBRIDGEAGT_create.mqsc
```

Example 6-2 Results of the SYSDBRIDGEAGT_create.mqsc command

```
C:\Documents and Settings\fteadmin>runmqsc FTPQMGR < "C:\Documents and Settings
All Users\Application Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDBRIDGEAGT\SYSDB
IDGEAGT_create.mqsc"
5724-H72 (C) Copyright IBM Corp. 1994, 2009. ALL RIGHTS RESERVED.
Starting MQSC for queue manager FTPQMGR.
```

```
11 MQSC commands read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

Make sure that there are no errors at the bottom of the script output.

3. Configure the bridge agent credentials.

The bridge agent user must be authenticated when the `SYSDBRIDGEAGT` agent connects to the FTP server adapter. The authentication of the bridge agent user at the FTP server can be done based on user ID and password credentials or by using a public/private key pair. The `ftecreateBridgeAgent` command creates the `ProtocolBridgeCredentials.xml` file in which the credential mapping for this specific agent is defined. In our scenario we use the user ID/password credentials for the authentication of our local user `admin` at the FTP server adapter:

- a. Navigate to the bridge agent's home directory:

```
C:\Documents and Settings\All Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDBRIDGEAGT
```

- b. Edit the `ProtocolBridgeCredentials.xml` file.
- c. Insert the following credentials:

```
<tns:user name="SYSTEM" serverUserId="admin"
serverPassword="<your_SI_password>" />
<tns:user name="fteadmin" serverUserId="admin"
serverPassword="<your_SI_password>" />
```

Note: Replace `<your_SI_password>` with your admin password for Sterling B2B Integrator.

Your ProtocolBridgeCredentials.xml file should look like Example 6-3.

Example 6-3 ProtocolBridgeCredentials.xml

```
<tns:credentials xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials"
                xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeCredentials
ProtocolBridgeCredentials.xsd ">

    <tns:serverHost name="sysc">
        <!-- Insert user elements here -->
        <tns:user name="SYSTEM" serverUserId="admin"
serverPassword="<your_SI_password>" />
        <tns:user name="fteadmin" serverUserId="admin"
serverPassword="<your_SI_password>" />
    </tns:serverHost>
</tns:credentials>
```

4. Start the bridge agent from the command line.

Open a command console and enter the following command:

```
fteStartAgent SYSDBRIDGEAGT
```

Check the **fteListAgents** command as in Example 6-4. If the agent is successfully registered, the agent name is output.

Example 6-4 fteListAgents output

```
>fteListAgents
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED
Agent Name:                Queue Manager Name:    Status:
SYSCAGT                    FTPQMGR                READY
SYSDAGT                    FTEQMGR                READY
SYSDBRIDGEAGT (FTP Bridge)  FTPQMGR                READY
```

6.3.10 Creating custom WebSphere MQ File Transfer Edition protocol

This section details the steps involved to customize Sterling B2B Integrator and Sterling File Gateway to enable WebSphere MQ File Transfer Edition as an available protocol.

Disclaimer: The Sterling B2B Integrator customization has been written to demonstrate interoperability between Sterling File Gateway and WebSphere MQ File Transfer Edition. It has been tested in basic scenarios, but the code is given as is. We suggest thoroughly analyzing the solution to ensure that it meets the requirements of your setup and to ensure that the solution is fully tested before it is used in any production environment.

Importing an XSD Schema File

A business process running in Sterling B2B Integrator initiates a WebSphere MQ File Transfer Edition file transfer by creating an XML message and placing it on the agent command queue to request a transfer.

An XSD Schema file has been created that defines the structure of the XML message:

1. Create a file called SFGFTECreateTransfer.xslt. The sample code for this file can be found in “SFGFTECreateTransfer.xslt file” on page 406.
2. In the Sterling B2B Integrator dashboard, go to **Deployment** → **XSLT** and click **Go!** to next to Check in new XSL Style Sheet (Figure 6-25).

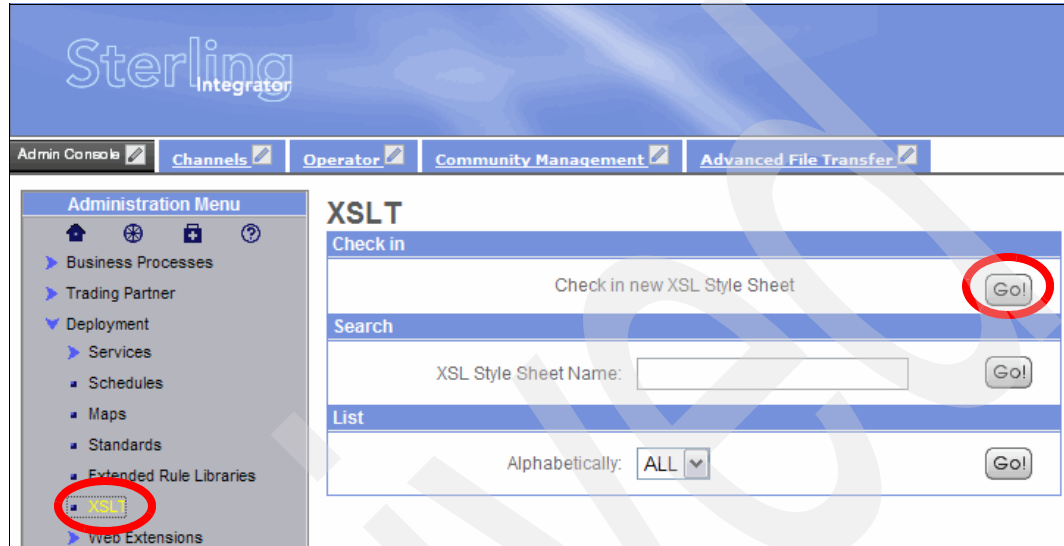


Figure 6-25 Check in XSL Style Sheet

3. Enter a name of SFGFTECreateTransfer (Figure 6-26). Click **Next**.

Note: The name of the XSLT style sheet is important. It must be the name of the file, minus the .xslt extension. This is due to the way that the business process is implemented. If this rule is not followed, the business process will not be able to find the necessary style sheet to create the XML file transfer request message.

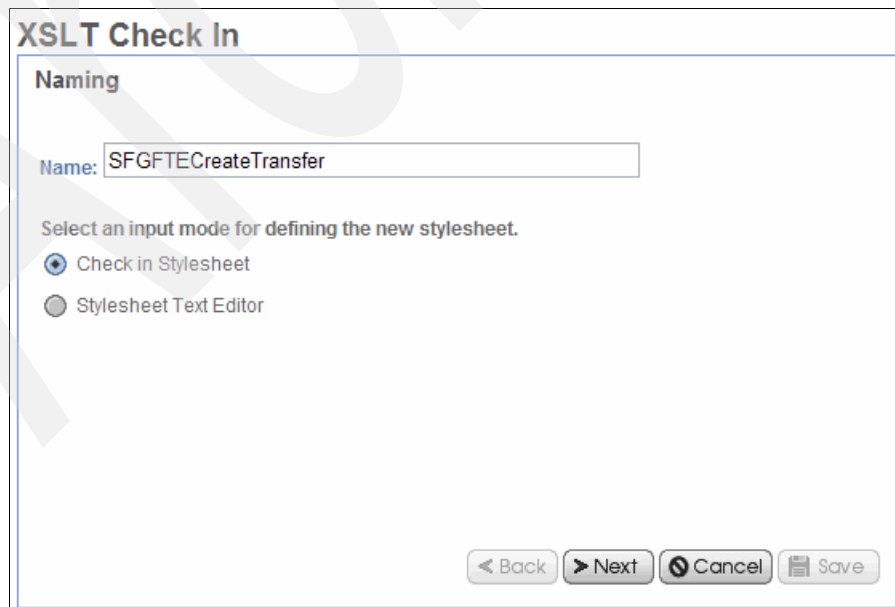


Figure 6-26 Enter name for XSLT style sheet

- Click **Browse** and select the **SFGFTECreateTransfer.xslt** file that you created or downloaded and enter a suitable comment in the Check-in Comments field (Figure 6-27). Click **Next**.

XSLT Check In

Naming: Select XSLT

XSL Stylesheet filename(.xslt): Z:\Temp\SFGFTECreateTransfer.xslt

Check-in Comments: XSLT file for WMQFTE transfer message

Encoding Type: UTF-8

XSL Stylesheet Type: SYSTEM

Figure 6-27 Browse to XSLT style sheet file

- Click **Finish** (Figure 6-28).

XSLT Check In

SFGFTECreateTransfer - XSL Style Sheet: Confirm

Enable for Business Processes

Check-In XSL Stylesheet

XSL Stylesheet Name	SFGFTECreateTransfer
XSL Stylesheet Type	0
Encoding Type	UTF-8
Check-in Comments	XSLT file for WMQFTE transfer message
XSL Stylesheet File Name	Z:\noback\Temp\SFGFTECreateTransfer.xslt

Figure 6-28 Final import XSLT screen

Checking in a custom business process

A custom business process is invoked to initiate the transfer to route a file from Sterling File Gateway using WebSphere MQ File Transfer Edition. To install the business process, follow these steps:

1. Create a file called CustomFileGatewayDeliveryFTE.bmp1. Sample code that can be used to create this file can be found in “CustomFileGatewayDeliveryFTE.bmp1 file” on page 408.
2. On the Sterling B2B Integrator dashboard, go to **Business Processes** → **Manager**, and click **Go!** in the Create section (Figure 6-29).

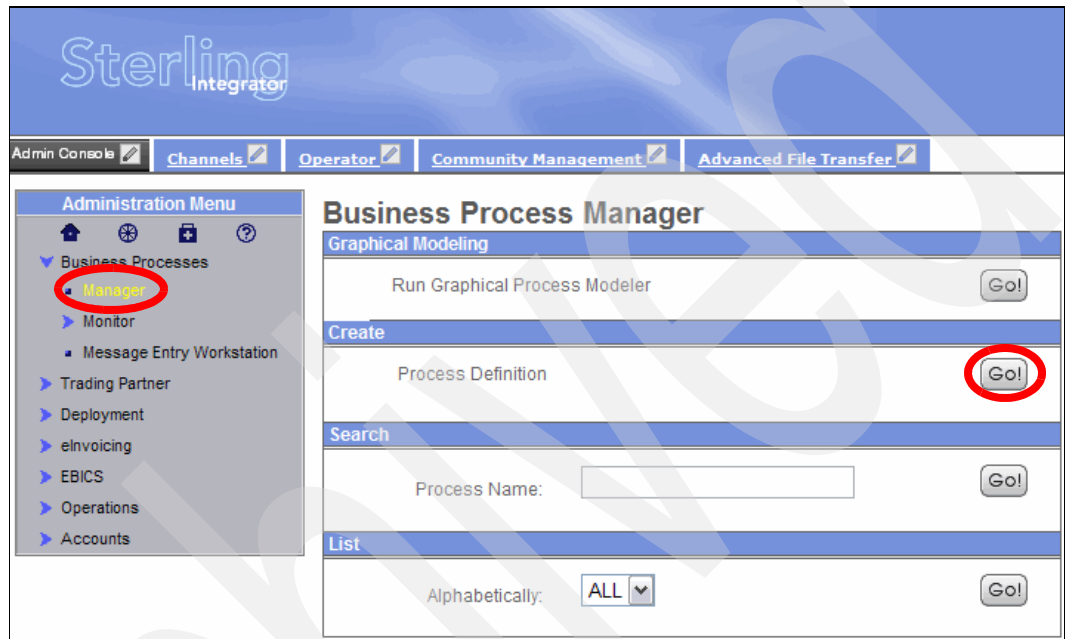
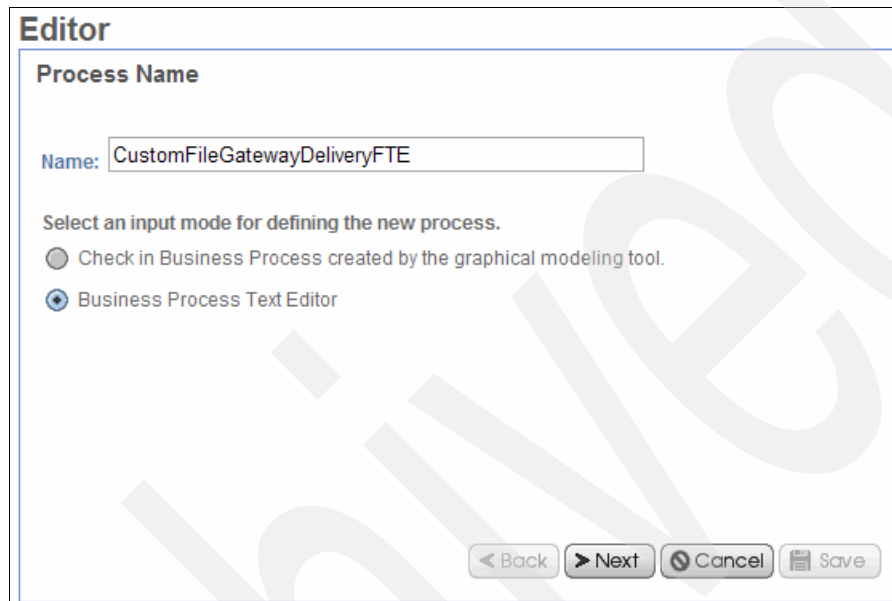


Figure 6-29 Import the custom business process

3. Enter the name CustomFileGatewayDeliveryFTE. Select the radio button next to **Business Process Text Editor** and click **Next** (Figure 6-30).

Note: The name of the business process is important. It must be CustomFileGatewayDeliveryFTE. If the name is different, the code involved in the customization to Sterling File Gateway will not be able to find the necessary business process to initiate the WebSphere MQ File Transfer Edition file transfer.



The screenshot shows a dialog box titled "Editor" with a "Process Name" section. Below the title bar, the text "Process Name" is displayed. A text input field labeled "Name:" contains the text "CustomFileGatewayDeliveryFTE". Below the input field, the instruction "Select an input mode for defining the new process." is shown. There are two radio button options: "Check in Business Process created by the graphical modeling tool." (which is unselected) and "Business Process Text Editor" (which is selected). At the bottom right of the dialog, there are four buttons: "< Back", "> Next", "Cancel", and "Save".

Figure 6-30 Enter a name for the business process

4. Enter a suitable description for the business process. Copy and paste the business process into the Business Process box (Figure 6-31). Click **Validate** if you want to validate the business process, and then click **Next**.

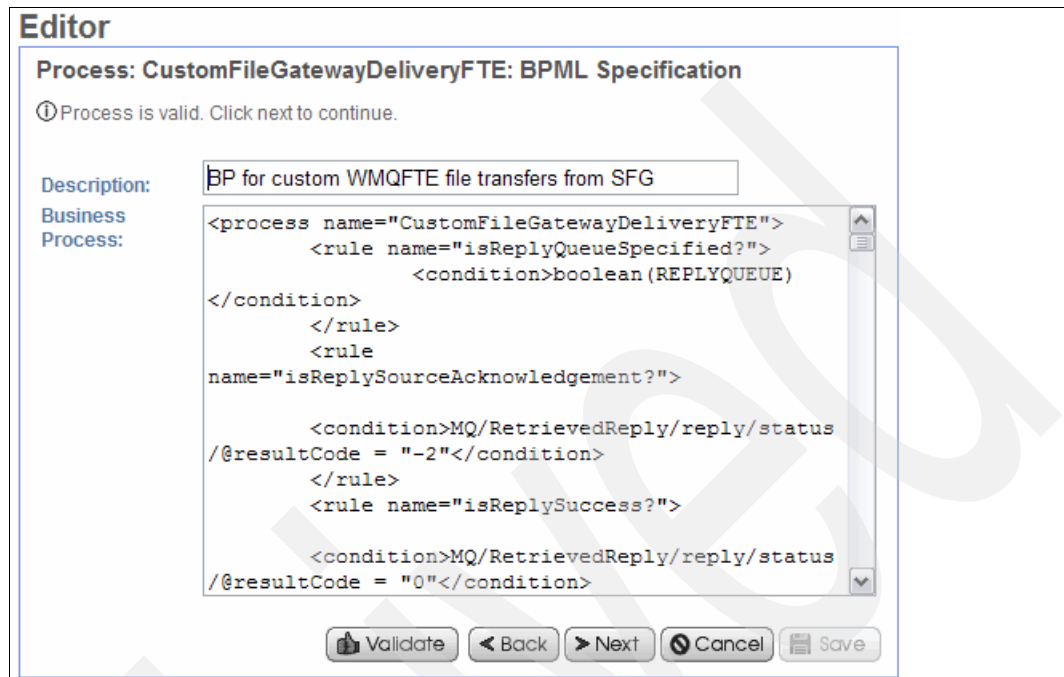


Figure 6-31 Copy and paste in the business process code

- Accept all the remaining default options in the wizard and keep clicking **Next** until you reach the final page (Figure 6-32). Click **Finish**.

Editor

Process: CustomFileGatewayDeliveryFTE: Confirm

Enable Business Process

Create Permission

Process Specification

Definition Name	CustomFileGatewayDeliveryFTE
Document Tracking	False
Set onfault processing	False
Queue	4
Start Mode	async
Transaction	FALSE
Category	None Available
Persistence Level	System Default
Event Reporting Level	None
Recovery Level	Manual
Document Store	System Default
Complete by - Deadline	None Available
First Notification	None Available
Second Notification	None Available
Life Span (System Default)	2 days and 0 hours
Removal Method (System Default)	Archived
Description	BP for custom WMQFTE file transfers from SFG
Default Version	This Business Process
Filename	Z:\noback\Temp\CustomFileGatewayDeliveryFTE.bpml
Errors	None
Business Process	CustomFileGatewayDeliveryFTE

< Back
Next >
⏹ Cancel
📄 Finish

Figure 6-32 Final business process import page

Adding customer_overrides.properties file to properties directory

The customer_overrides.properties file defines custom Sterling File Gateway events that are used in the CustomFileGatewayDeliveryFTE business process. This file needs placing into a specific directory as described in these steps:

1. You can download the customer_overrides.properties file, as describe in Appendix E, “Additional material” on page 425, or copy and paste the code provided in “Customer_overrides.properties file” on page 419 into a file called customer_overrides.properties.
2. Copy the file into the <SI_install_root>/properties directory.

Note: These steps assume that you do not already have custom overrides defined in your environment. If you do, refer to the Sterling Commerce documentation to add these changes to your existing configuration, as opposed to overwriting your existing configuration.

Adding Sterling File Gateway customization files

WebSphere MQ File Transfer Edition is being added as a custom protocol in Sterling File Gateway. WebSphere MQ File Transfer Edition will be available for selection when creating new partners as listening consumers in Sterling File Gateway.

The two files used in this section define the input parameters required for the new protocol and customize the Sterling File Gateway user interface to accept the new protocol as a selection and to allow the user to input the necessary parameter values. Follow these steps to import the files:

1. Download the AFTExtensionsCustomer.properties and AFTExtensionsCustomer.xml files as described in Appendix E, “Additional material” on page 425. Alternatively, create two new files and add the code from “AFTExtensionsCustomer.properties source file” on page 421 and “AFTExtensionsCustomer.xml source file” on page 422.
2. Copy the AFTExtensionsCustomer.properties file to <SI_ROOT>/container/Applications/aft/WEB-INF/classes/resources.
3. Copy the AFTExtensionsCustomer.xml file to <SI_ROOT>/container/Applications/aft/WEB-INF/classes/resources/xml.

Note: These steps assume that you do not already have custom overrides defined in your environment. If you do, refer to the Sterling Commerce documentation to add these changes to your existing configuration, as opposed to overwriting your existing configuration.

Modifying settings to stop forwarding of .part files

During a WebSphere MQ File Transfer Edition transfer between agents, the file is initially stored on the target file system with a .part extension. When the transfer has successfully completed, the .part extension is removed.

The default configuration of Sterling File Gateway would mean that these .part files would be picked up and forwarded immediately, meaning that incomplete files would be routed to the end destination, which is not the desired result. To alter this default behavior to ignore .part file names until the transfer is complete:

1. Open the `<SI_ROOT>/properties/customer_overrides.properties` file. Add the following line to the top of the file:

```
filegateway.ignoreFilename=.[.]part[0-9]*
```

2. Save and close the file.

Disallowing duplicate file names

Default behavior of Sterling File Gateway is to allow two files with identical names to exist in the same mailbox at the same time. This can cause problems when the mailbox is being accessed by FTP clients, as is the case in this scenario. To stop these problems, change the behavior to disallow duplicate messages existing in one mailbox:

1. Open the `<SI_ROOT>/properties/customer_overrides.properties` file. Add the following line to the top of the file:

```
mailbox.disallowDuplicateMessages=true
```

2. Save and close the file.

This ensures that every message in a single mailbox has a unique name. It also ensures that a message and a mailbox do not have the same name. If you write a message to a mailbox and the name matches the name of a message in the mailbox, the service deletes the old message before adding the new message.

Deploy the custom files and restart Sterling B2B Integrator

With all necessary customization files in place, we need to deploy these customizations into Sterling File Gateway and restart Sterling B2B Integrator for the changes to take effect:

1. Stop Sterling B2B Integrator. Run:

```
<SI_install_dir>/bin/stopWindowsService.cmd
```
2. Run `<SI_install_dir>/bin/deployer.cmd`.
3. Restart Sterling B2B Integrator by double-clicking the desktop icon **Sterling_Integrator_at_8080**.

6.3.11 Creating a WebSphere MQ reply queue

Through its WebSphere MQ adapter, Sterling B2B Integrator listens to a reply queue to receive replies back from WebSphere MQ File Transfer Edition. These replies contain information regarding the status of the transfer and the transfer command Sterling B2B Integrator placed on the SYSDBRIDGEAGT's command queue. To capture these responses, a local queue must be created on the WebSphere MQ queue manager local to SYSDBRIDGEAGT and Sterling File Gateway, FTPQMGR.

Instructions for creating a local queue can be found at the following link pointing to the WebSphere MQ Information Center as part of Tutorial 1: Sending a message to a local queue.

http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/topic/com.ibm.mq.explorer.tutorials.doc/bi00257_.htm?resultof=%22%63%72%65%61%74%65%22%20%22%63%72%65%61%74%22%20%22%6c%6f%63%61%6c%22%20%22%71%75%65%75%65%22%20

We named our local queue REPLYMSGQ. This queue name is needed in “Creating new trading partner SysD_Partner” on page 220.

6.3.12 Configuring Sterling File Gateway

This section describes the steps involved in configuring Sterling File Gateway to enable communication from an external partner on SysA using the Sterling Connect:Direct protocol to an internal WebSphere MQ File Transfer Edition agent on SysD.

Starting Sterling File Gateway

To log in to the Sterling File Gateway user interface, follow these steps:

1. Sterling File Gateway should already be started. If this is not the case and your server is stopped, start it by double-clicking the desktop icon **Sterling_Integrator_at_8080**.
2. Start Internet Explorer and go to:
`http://<servername>:<port>/filegateway/`
3. Log in using your administrator user ID and password (Figure 6-33). The default user ID for Sterling File Gateway is `fg_sysadmin`.

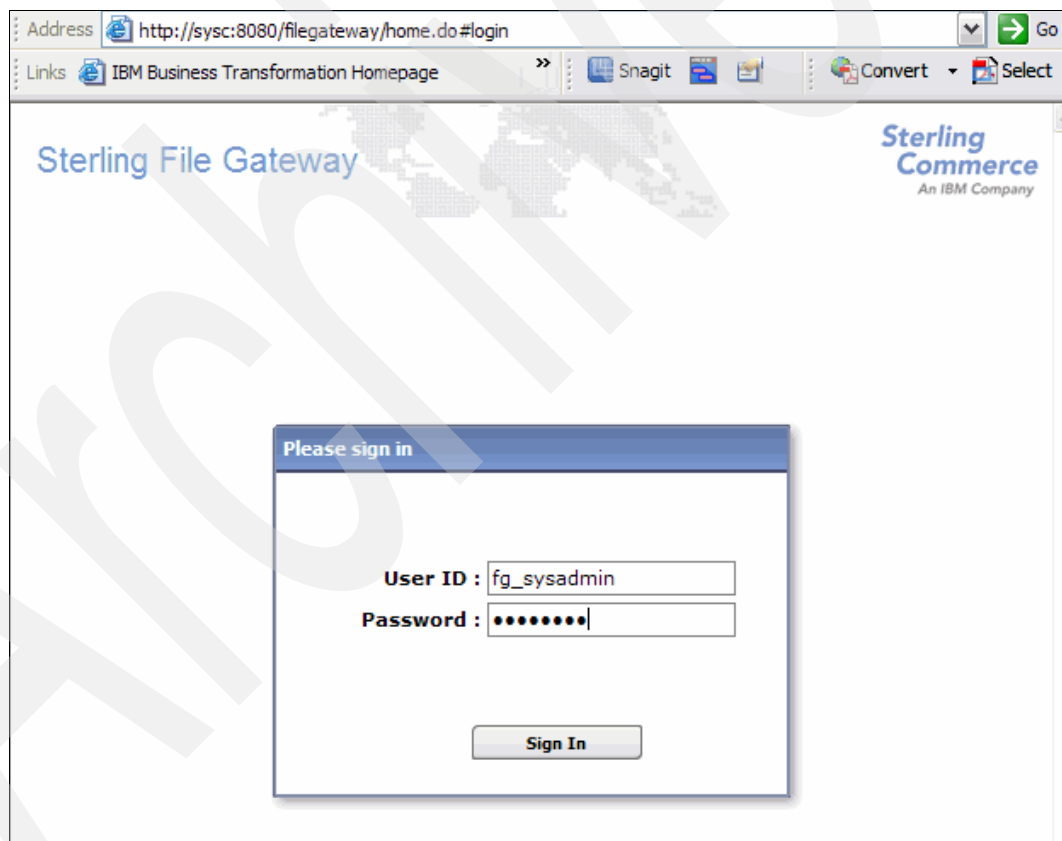


Figure 6-33 Login screen for Sterling File Gateway

This brings up the main page for Sterling File Gateway (Figure 6-34).

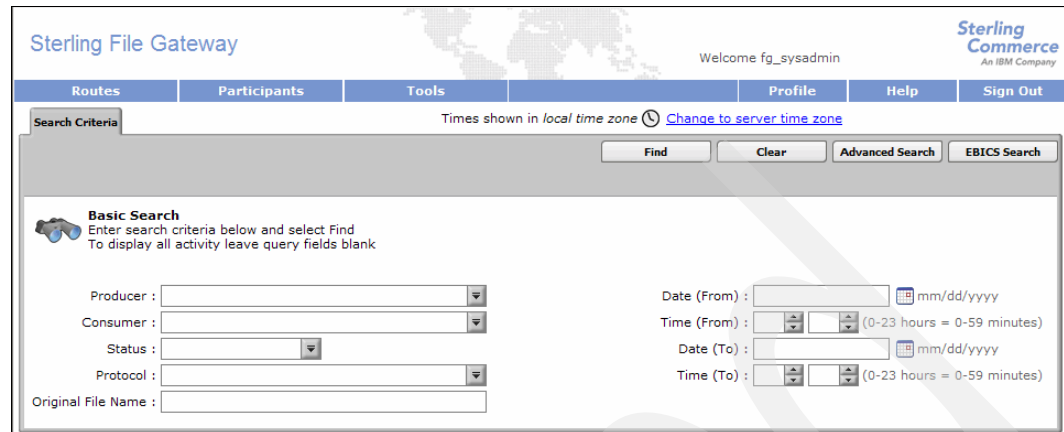


Figure 6-34 First screen when logged in to Sterling File Gateway

Configuring FirstCommunity community in Sterling File Gateway

A community defines the protocols that partners within this community can use. We need a community in Sterling File Gateway that has WebSphere MQ File Transfer Edition as an available protocol. Choose one of the following two options:

- ▶ If you have already followed the steps in “Creating a community” on page 131, follow the steps below to modify the FirstCommunity community in Sterling File Gateway to add WebSphere MQ File Transfer Edition as an available protocol.
- ▶ If you have not already created the FirstCommunity community in Sterling File Gateway, skip to “Creating the FirstCommunity community in Sterling File Gateway” on page 212.

If you followed the steps in the previous chapter to create community FirstCommunity, you currently have a community that can use the following protocols:

- ▶ SSH/SFTP
- ▶ Sterling Connect:Direct
- ▶ FTP or FTPS
- ▶ MAILBOX

We can modify this community to allow WebSphere MQ File Transfer Edition to be used by members of this community.

To add WebSphere MQ File Transfer Edition to the community, follow these steps:

1. Select **Participants** → **Communities** (Figure 6-35).

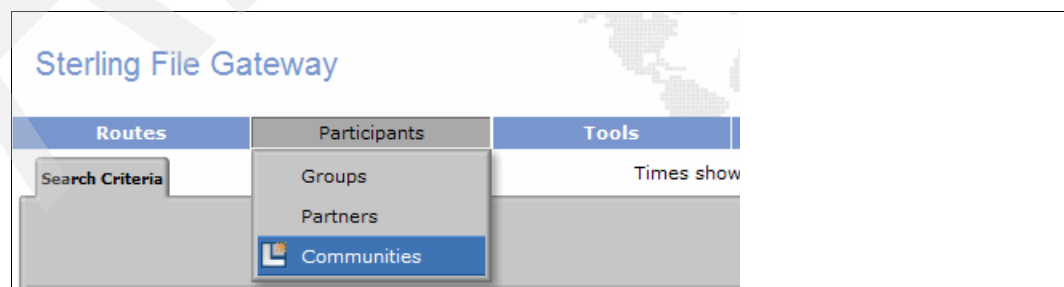


Figure 6-35 Modify FirstCommunity

2. On the Communities page that is opened, click **edit** to edit the settings for FirstCommunity (Figure 6-36).

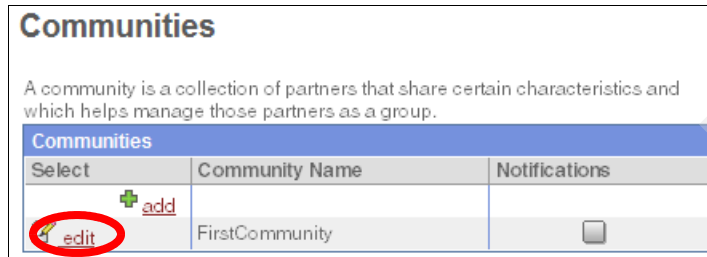


Figure 6-36 Edit FirstCommunity

3. Click **Edit** in the Protocols section (Figure 6-37).

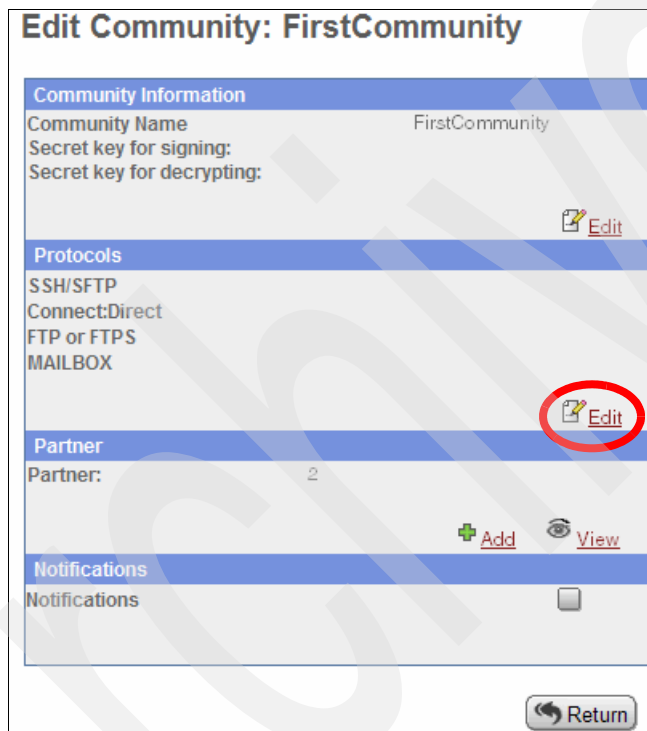


Figure 6-37 Edit the available protocols

4. On the next page, you should see that WebSphere MQ FTE is now listed as an available protocol. Select it and click the right-arrow to move it into the Selected field (Figure 6-38). Click **Save**.

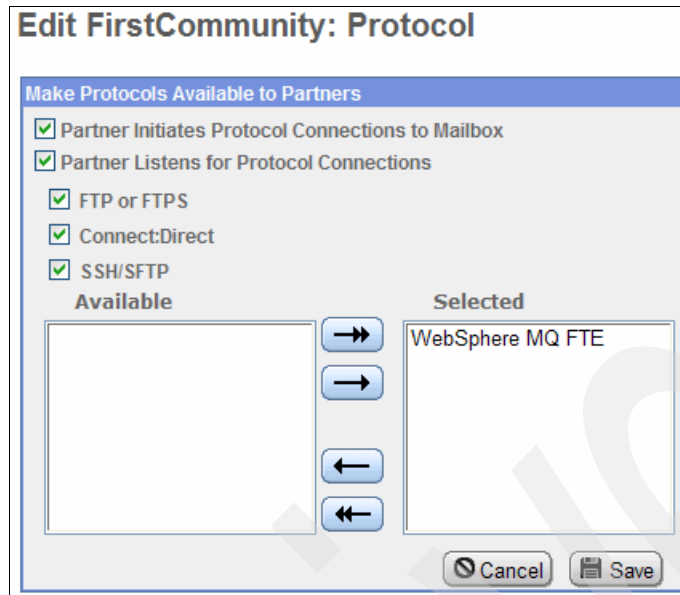


Figure 6-38 Select WebSphere MQ FTE

5. You should now see Websphere MQ FTE listed in the Protocols section of the Edit Community:FirstCommunity panel (Figure 6-39). Close this window.

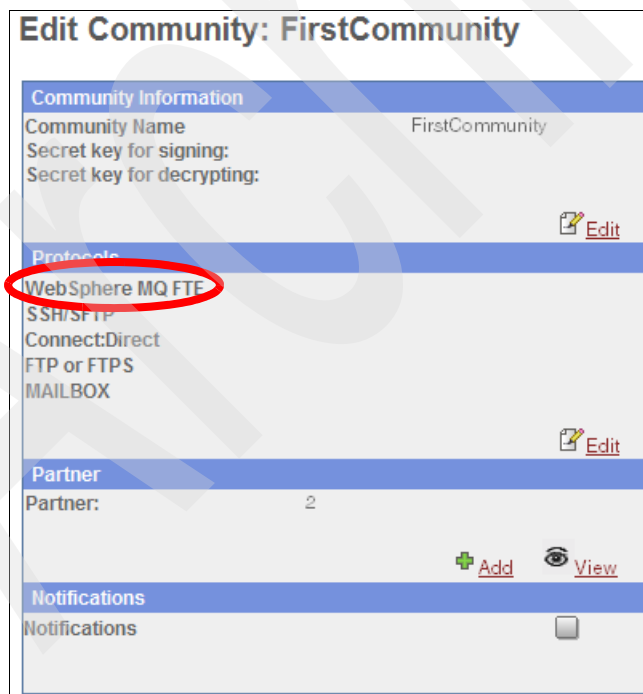


Figure 6-39 WebSphere MQ FTE is now listed

Creating the FirstCommunity community in Sterling File Gateway

If you do not already have a community named FirstCommunity in Sterling File Gateway, follow the steps below to create it and enable it for WebSphere MQ File Transfer Edition:

1. Select **Participants** → **Communities** (Figure 6-40).

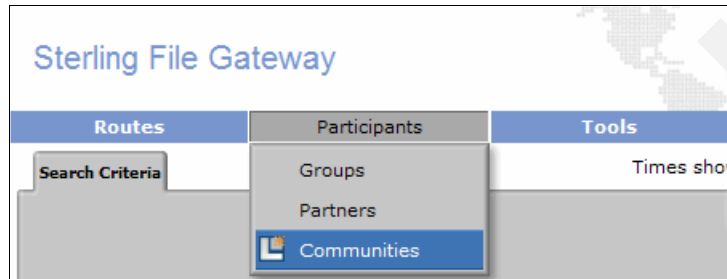


Figure 6-40 Create a new community

2. In the pop-up window, select **add** (Figure 6-41).



Figure 6-41 Click add

3. Enter a name for your community, FirstCommunity (Figure 6-42) and click **Next**. For simplicity, we create only one community for this book and enable all protocols within that community.

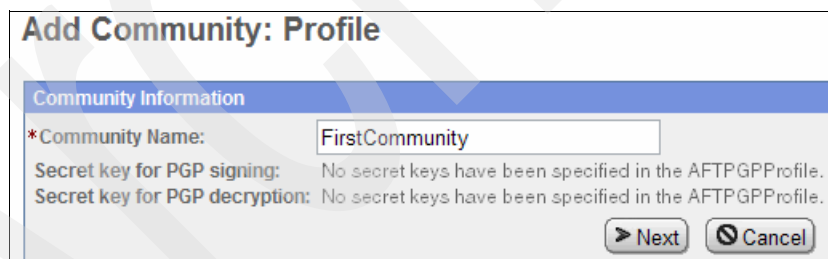


Figure 6-42 Name the community

4. Select all available check boxes to allow all partners in this community to communicate using all available protocols (FTP or FTPS, Sterling Connect:Direct, SSH/SFTP) and move WebSphere MQ FTE across into the Selected box (Figure 6-43). Click **Next**.

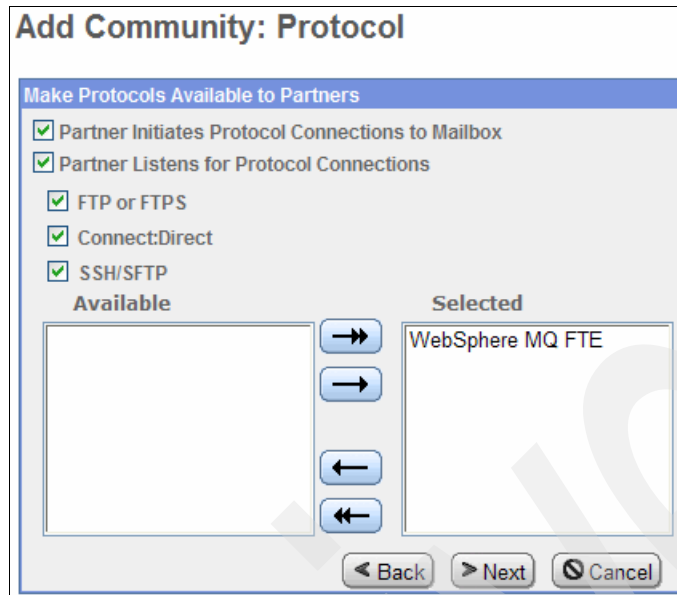


Figure 6-43 Select all protocols

5. On the Notifications page, accept the default setting of **No** and click **Next** (Figure 6-44).

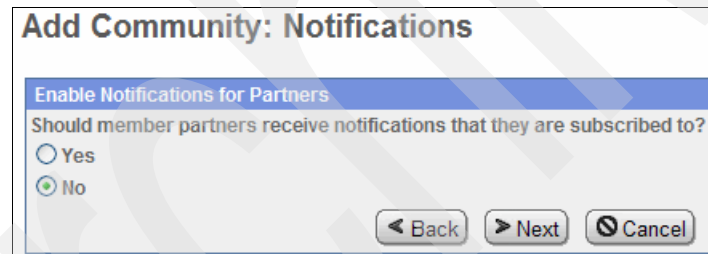


Figure 6-44 Accept the default setting for notifications

- On the confirmation page, click **Finish** (Figure 6-45), then close the window and go back to the main Sterling File Gateway page.

Figure 6-45 Add community confirmation screen

Creating routing channel templates in Sterling File Gateway

If you have not already done so, create two routing channel templates:

- Follow the steps to create the template PassThrough_RouteByMailbox described in “Creating the PassThrough_RouteByMailbox routing channel template” on page 139.
- Follow the steps to create the second template PassThrough described in “Creating the PassThrough routing channel template” on page 159.

Creating Partner SysA_CD_Partner

If you have not already created the SysA_CD_Partner in Sterling File Gateway (described in “Creating partners” on page 132), follow the steps below to do so now.

The partner you will create now is an external partner (Company A) that owns the Connect:Direct node SysA_CD on external machine SysA.

- Go to **Participants** → **Partners** (Figure 6-46).

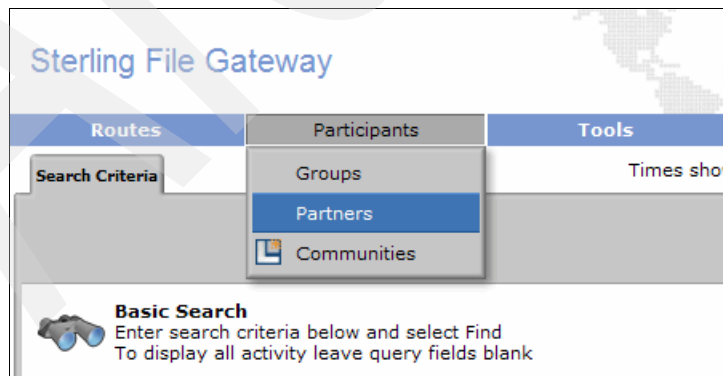


Figure 6-46 Create partners

2. Click **Create** (Figure 6-47).

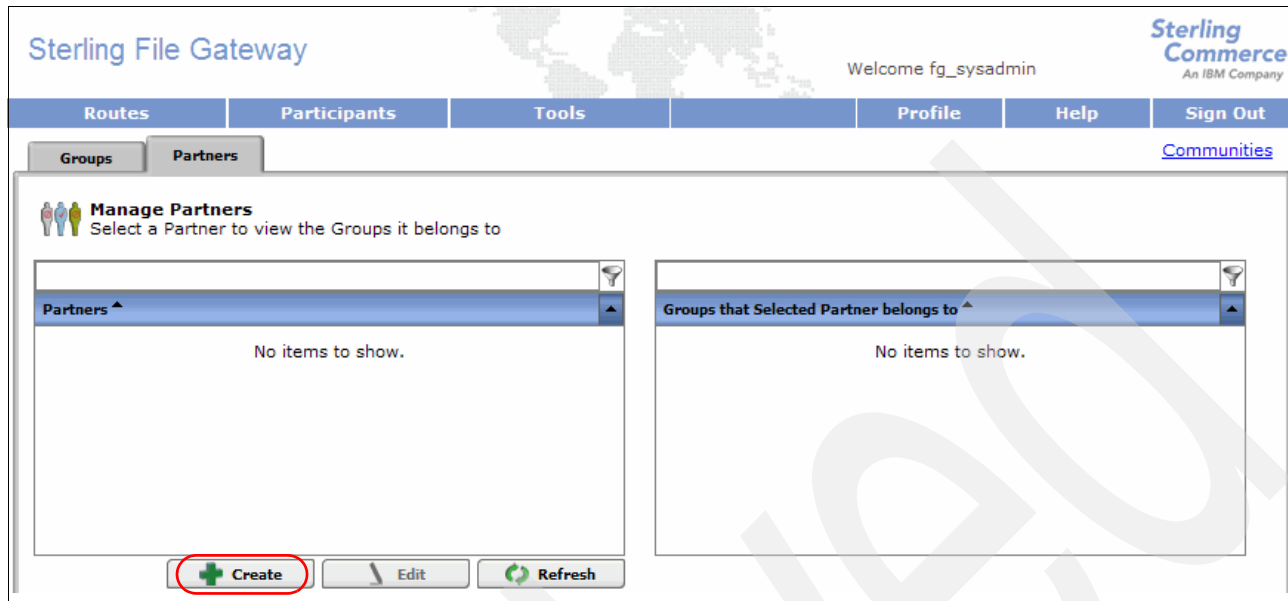


Figure 6-47 Create new partner

3. In the pop-up window, select the **FirstCommunity** community and click **Next** (Figure 6-48).

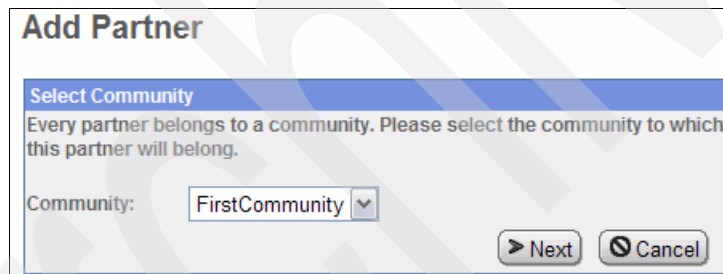


Figure 6-48 Select community that the new partner is a member of

4. Create the SysA partner, who in this scenario will be communicating via Sterling Connect:Direct into Sterling File Gateway. Use the name SysA_CD_Partner. The Phone and Email Address fields are mandatory even though we have notifications disabled in this example. We used a false telephone number and email address for this example (Figure 6-49). Click **Next**.



The screenshot shows a web form titled "Add Partner: Information" with a sub-header "Contact Information". The form contains the following fields and values:

- * Partner Name: SysA_CD_Partner
- Address: (empty)
- City: (empty)
- State: (empty)
- Postal Code: (empty)
- * Phone: 1234
- Country: UNITED STATES (dropdown menu)
- Time Zone: (GMT-05:00) Eastern Time (US & Canada) (dropdown menu)
- * Email Address: sysacd@itso_redbooks.com

At the bottom right of the form are three buttons: "< Back", "> Next", and "Cancel".

Figure 6-49 Enter details for the SysA external business partner

5. Enter a user name and password for the SysA external partner. These are values that need to be passed into Sterling File Gateway from Sterling Connect:Direct to place files into SysA_CD_Partner's mailbox. They are also the same values that would need to be used if the SysA partner wanted to use either the myFileGateway interface to access their mailboxes or the FTP/SFTP to put files directly into (or get files from) their mailboxes.

Use SysA_CD_Partner as the user name and its04you as the password. As this is an example scenario, choose appropriate values for first name and last name (Figure 6-50). Click **Next**.

Note: You cannot use a password of password with Connect:Direct nodes, as it is considered a reserved keyword, and file transfer scripts in Sterling Connect:Direct will fail validation.

Add Partner: User Account

User Account

Create a user account by filling the following information. This user account is to be used by SysA_CD_Partner when logging into myFileGateway.

*User Name: SysA_CD_Partner

*Password: ●●●●●●

*Confirm Password: ●●●●●●

*First Name: SysA_CD_Firstname

*Last Name: SysA_CD_Lastname

< Back > Next Cancel

Figure 6-50 Enter login details for the SysA external business partner

- On the Partner Role page, select that the partner is both a **Consumer of Data** and a **Producer of Data**. This is because SysA_CD_Partner puts files into a mailbox on Sterling File Gateway (Producer of Data) and also retrieves files that have been placed into its mailbox (Consumer of Data). That is, this partner is used for both inbound and outbound file transfers.

In the Consumer of Data section, select the **Partner will listen for a connection to consume data** option. This means that SysA_CD_Partner will listen on a designated mailbox for files to consume any files placed there (Figure 6-51). Click **Next**.

Add Partner: Partner Role

Select Partner Role

There are no consumers configured in the system right now. It is recommended that the first partner that is configured is one that is a consumer or both a consumer and a producer.

SysA_CD_Partner is a Consumer of Data

Partner will initiate the connection to consume data.

Partner will listen for a connection to consume data.

SysA_CD_Partner is a Producer of Data

< Back > Next Cancel

Figure 6-51 Partner role screen

- On the Initiate Connection Settings page, accept the default answer of **No**. This partner only communicates via Sterling Connect:Direct, not via SSH/SFTP or SSH/SCP protocols (Figure 6-52). Click **Next**.

Figure 6-52 Initiate Connections Settings page

- On the Protocol page, select **Listen for Connect:Direct Connections** from the drop-down menu and select **Next** (Figure 6-53).

Figure 6-53 Select Connect:Direct from the drop-down menu

- On the Sterling Connect:Direct page, enter the details for the local and remote Connect:Direct nodes according to SysA_CD_Partner. In this case, the local Connect:Direct node name is the name of the Connect:Direct server adapter configured on Sterling B2B Integrator on SysC, SysC_CDSA. The remote Connect:Direct node name is SysA_CD. Enter the user ID and password for the remote Connect:Direct node on SysA, for example, cdadmin/cdadmin. Click **Next** (Figure 6-54).

Figure 6-54 Enter the Connect:Direct node details for SysA_CD_Partner

10. On the PGP Settings page, accept the defaults (both **No**) (Figure 6-55). Click **Next**.

PGP Settings

PGP Settings

When SysA_CD_Partner sends PGP packaged files
 ... the files are processed in accordance with the Routing Channels (and their templates) that SysA_CD_Partner is a producer for.

Note:
 *If SysA_CD_Partner sends data that is encrypted, it will be decrypted using the Router's secret PGP key.
 *If SysA_CD_Partner sends data that is signed, it will be verified using SysA_CD_Partner's public PGP key if that key is present in the public key ring.

Does SysA_CD_Partner require data to be signed by the Router ?
 Yes No

Does SysA_CD_Partner require data to be encrypted by the Router ?
 Yes No

< Back Next > Cancel

Figure 6-55 PGP Settings screen

11. The confirmation page displays your choices. Click **Finish** to complete the creation of SysA_CD_Partner (Figure 6-56).

Add Trading Partner: Confirm

Profile
 Partner Name: SysA_CD_Partner
 Address: ,
 Phone: 1234
 Email Address: sysacd@itso_redbooks.com

User Account
 User Name: SysA_CD_Partner
 Password: *****
 First Name: SysA_CD_Firstname
 Last Name: SysA_CD_Lastname

Protocol
 Partner Role: Producer of Data, Consumer of Data
 Connection Direction: Listen Connection
 Transport Method: Connect:Direct
 • Local Node Name: SysC_CD_SA
 • Remote Node Name: SysA_CD
 • Remote User Id: cdadmin
 • Remote Password: *****
 • Remote File Name:
 • Disposition to be applied to remote file:
 • Sysopts string to be applied to remote file:
 • CheckpointInterval to be used for checkpoint restart

Does SysA_CD_Partner require data to be signed by the Router : No
 Does SysA_CD_Partner require data to be encrypted by the Router : No

Community Membership
 Community Name: FirstCommunity
 Joined Date: Today

< Back Cancel Finish

Figure 6-56 Confirmation page

12. When the partner has been added successfully, close the browser window and return to the main Sterling File Gateway browser window.

Creating new trading partner SysD_Partner

Create a trading partner in Sterling File Gateway to send files to internal system SysD using the WebSphere MQ File Transfer Edition protocol:

1. Back in the main Sterling File Gateway window, go to **Participants** → **Partners** (Figure 6-57).



Figure 6-57 Create partners

2. Click **Create** (Figure 6-58).

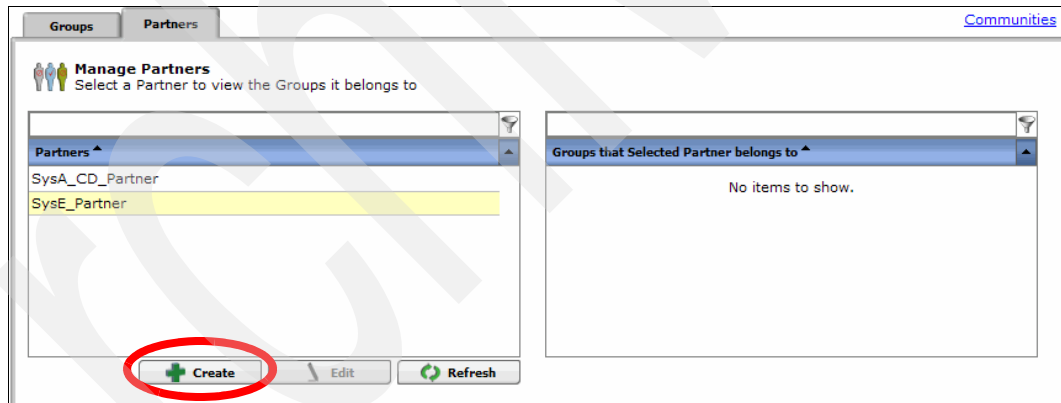


Figure 6-58 Create new partner

3. In the pop-up window, select community **FirstCommunity** and click **Next** (Figure 6-59).

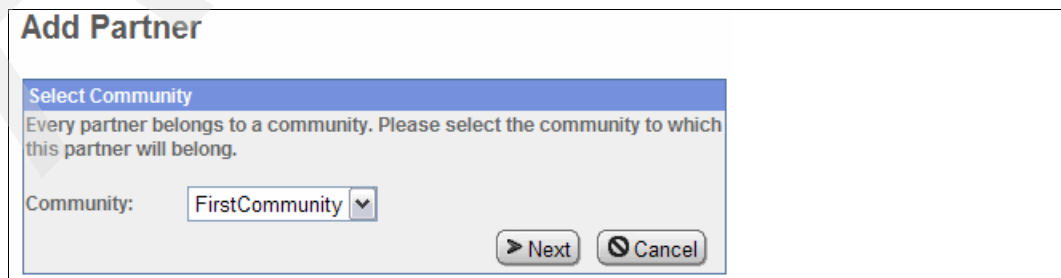


Figure 6-59 Select community that the new partner is a member of

4. Create the SysD business partner, who in this case will be communicating with Sterling File Gateway via WebSphere MQ File Transfer Edition. Use the name SysD_Partner. The Phone and Email Address fields are mandatory even though we have notifications disabled in this example. We used a false telephone number and email address for this example (Figure 6-60). Click **Next**.

Figure 6-60 Enter details for the SysD internal business partner

5. Enter a user name and password for the SysD internal business partner. Use SysD_Partner as the user name and itso4you as the password. As this is an example scenario, make up values for first name and last name (Figure 6-61). Click **Next**.

Figure 6-61 Enter log in details for the SysD internal business partner

6. On the Partner Role page, select the partner to be both a **Consumer of Data** and a **Producer of Data**. This is because SysD_Partner will put files into a mailbox on Sterling File Gateway (Producer of Data), and will also retrieve files that have been placed into its mailbox (Consumer of Data). That is, this partner will be used for both inbound and outbound file transfers.

Under the Consumer of Data section, select the **Partner will listen for a connection to consume data** option. This means that SysD_Partner will listen on a designated mailbox for files to consume any files placed there (Figure 6-62). Click **Next**.

Add Partner: Partner Role

Select Partner Role

There are no consumers configured in the system right now. It is recommended that the first partner that is configured is one that is a consumer or both a consumer and a producer.

SysD_Partner is a Consumer of Data

Partner will initiate the connection to consume data.

Partner will listen for a connection to consume data.

SysD_Partner is a Producer of Data

< Back > Next Cancel

Figure 6-62 Partner role page

7. On the Initiate Connection Settings page, accept the default answer of **No**. This partner only communicates via WebSphere MQ File Transfer Edition, not via SSH/SFTP or SSH/SCP protocols (Figure 6-63). Click **Next**.

Add Partner: Initiate Connections Settings

Initiate Connections Settings

Will SysD_Partner use either SSH/SFTP or SSH/SCP protocol to initiate connections?

Yes No

< Back > Next Cancel

Figure 6-63 Initiate Connections Settings screen

8. On the Protocol page, select **Listen for WebSphere MQ FTE Connections** from the drop-down menu, and then select **Next** (Figure 6-64).

Add Partner: Protocol

Select a Protocol Published by FirstCommunity

Select Protocol:

SysD_Partner will ... Listen for WebSphere MQ FTE Connections

< Back > Next Cancel

Figure 6-64 Select Listen for WebSphere MQ FTE Connections from the drop-down menu

9. On the WebSphere MQ FTE page, enter the details provided in Table 6-3.

Table 6-3 WebSphere MQ File Transfer Edition protocol values for SysD_Partner

Parameter	Value
Source agent name (-sa)	SYSDBRIDGEAGT
Source agent queue manager (-sm)	FTPQMGR
Destination agent name (-da)	SYSDAGT
Destination agent queue manager (-dm)	FTEQMGR
Destination agent's directory (-dd)	c:\downloads\
Queue for transfer status reply messages	REPLYMSGQ
Destination file already exists (-de)	overwrite
Transfer timeout (seconds)	600

Note: A queue has been defined for the transfer status reply messages. Although this field is optional, we suggest using a queue for transfer status messages. If no queue is specified here, the business process that initiates the WebSphere MQ File Transfer Edition cannot monitor the status of the file transfer. If no queue is specified, then as long as the business process can successfully write the transfer request message to the command queue, it will report a successful file transfer even if the file transfer fails. This is because the business process will have effectively completed due to there being no way specified for Sterling B2B Integrator to monitor the status.

If a queue is specified, WebSphere MQ File Transfer Edition will place status messages on that queue. The business process will listen on that queue and can report with more certainty any transfer successes or failures.

Click **Next** (Figure 6-65).

Figure 6-65 Enter the WebSphere MQ File Transfer Edition details for SysD_Partner

Table 6-4 lists the parameters collected when configuring a Sterling File Gateway partner to use the WebSphere MQ File Transfer Edition protocol.

Table 6-4 Parameters used by a WebSphere MQ File Transfer Edition partner

Parameter name	Values/use
Source agent name (-sa)	Source agent name (name of local FTP Bridge Agent).
Source agent queue manager (-sm)	Source agent's queue manager.
Source agent queue manager host name	Host or IP of source agent queue manager (required only if connecting to WebSphere MQ in client mode).
Source agent queue manager port	TCP port of source agent queue manager (required only if connecting to WebSphere MQ in client mode).
Source agent queue manager user ID	User ID used when connecting to source agent's queue manager.
Source agent queue manager password	Password used when connecting to source agent's queue manager.
Destination agent name (-da)	Destination agent name.
Destination agent queue manager (-dm)	Destination agent queue manager.

Parameter name	Values/use
Destination agent's directory (-dd)	Destination directory.
Destination file already exists (-de)	Disposition when file exists on destination agent (error or overwrite).
Queue for transfer status reply messages	Name of queue on source agent's queue manager used to contain transfer status reply messages. If blank, Sterling File Gateway considers a delivery successful if the request message is successfully added to the source agent's command queue. If populated, Sterling File Gateway looks for reply messages on this queue to determine whether the transfer completed or failed.
Priority (-pr)	Priority of transfer (0 - 9).
Conversion (-t)	File conversion performed during transfer (binary or text).
Checksum method (-cs)	Checksum method computed by agents (MD5 or none).
Transfer timeout (seconds)	The number of seconds that Sterling File Gateway waits for a reply from the source agent to determine whether the transfer completed. If no reply is received before timeout occurs, Sterling File Gateway marks delivery as failed <i>even though the transfer might still be pending in WebSphere MQ File Transfer Edition and will eventually complete.</i>

10. On the PGP Settings page, accept the defaults (both **No**) (Figure 6-66). Click **Next**.

PGP Settings

PGP Settings

When SysD_Partner sends PGP packaged files
 ... the files are processed in accordance with the Routing Channels (and their templates) that SysD_Partner is a producer for.

Note:
 *If SysD_Partner sends data that is encrypted, it will be decrypted using the Router's secret PGP key.
 *If SysD_Partner sends data that is signed, it will be verified using SysD_Partner's public PGP key if that key is present in the public key ring.

Does SysD_Partner require data to be signed by the Router ?
 Yes No

Does SysD_Partner require data to be encrypted by the Router ?
 Yes No

◀ Back Next ▶ Cancel

Figure 6-66 PGP Settings page

11. The confirmation page displays your choices. Click **Finish** to complete the creation of SysD_Partner (Figure 6-67).

Profile	
Partner Name:	SysD_Partner
Address:	
Phone:	5555
Email Address:	sysd@itso_redbooks.com
 edit	
User Account	
User Name:	SysD_Partner
Password:	*****
First Name:	SysD_Firstname
Last Name:	SysD_Lastname
 edit	
Protocol	
Partner Role:	Producer of Data, Consumer of Data
Connection Direction:	Listen Connection
Transport Method:	WebSphere MQ FTE
<ul style="list-style-type: none"> • Source Agent Name (-sa) • Source Agent Queue Manager (-sm) • Source Agent Queue Manager Host Name • Source Agent Queue Manager Port • Source Agent Queue Manager User Id • Source Agent Queue Manager Password • Destination Agent Name (-da) • Destination Agent Queue Manager (-dm) • Destination Agent's Directory (-dd) • Destination File Already Exists (-de) • Queue For Transfer Status Reply Messages • Priority (-pr) • Conversion (-t) • Checksum Method (-cs) • Transfer Timeout (seconds) 	SYSDBRIDGEAGT FTPQMGR ***** SYSDAGT FTEQMGR c:\downloads\ overwrite REPLYMSGQ 0 binary MD5 600
Does SysD_Partner require data to be signed by the Router :	No
Does SysD_Partner require data to be encrypted by the Router :	No
 edit	
Community Membership	
Community Name:	FirstCommunity
Joined Date:	2010-11-22 15:30:27.953

Figure 6-67 Confirmation page

12. When the partner has been added successfully, close the browser window and return to the main Sterling File Gateway browser window.

Creating inbound routing channel

The final step in the inbound scenario is to create the routing channel to define the route from SysA_CD_Partner to SysD_Partner. Back in the main Sterling File Gateway browser, select **Routes** → **Channels**. Click **Create**. Select the values listed in Table 6-5 and click **Save**.

Table 6-5 Values for routing channel

Parameter	Value
Template	PassThrough_RouteByMailbox
Producer	SysA_CD_Partner
Consumer	SysD_Partner

Figure 6-68 shows this.

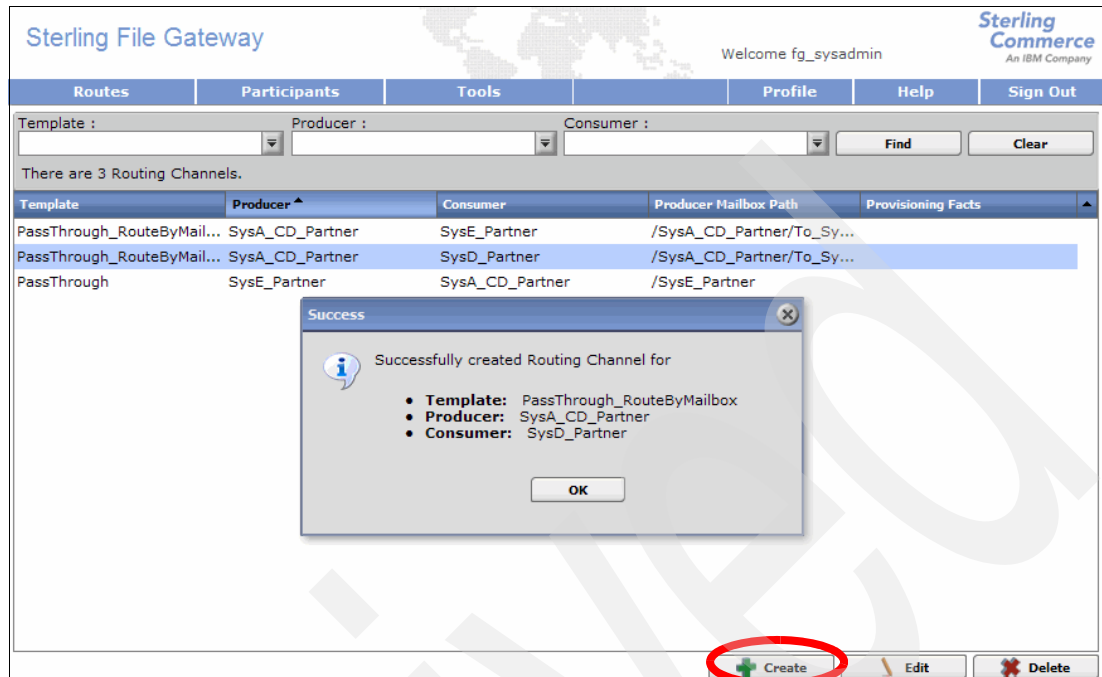


Figure 6-68 Create routing channel

This route is now configured and ready for inbound file transfers. If a Sterling Connect:Direct transfer from the Connect:Direct node on SysA (SysA_CD) sends a file into the Connect:Direct server adapter on SysC (SysC_CDSA), into the mailbox path for SysA_CD_Partner of /mailbox/To_SysD_Partner, then the file will be routed to SysD_Partner by initiating a WebSphere MQ File Transfer Edition file transfer. The transfer will use the WebSphere MQ File Transfer Edition bridge agent (SYSDBRIDGEAGT) to send the file to the WebSphere MQ File Transfer Edition agent on SysD (SYSDAGT). This transfer places the file into the c:\downloads directory on the destination machine SysD.

Creating outbound routing channel

The final step is to create the routing channel to define the route from SysD_Partner to SysA_Partner:

1. Back in the main Sterling File Gateway browser, select **Routes** → **Channels**.
2. Click **Create** (Figure 6-69).

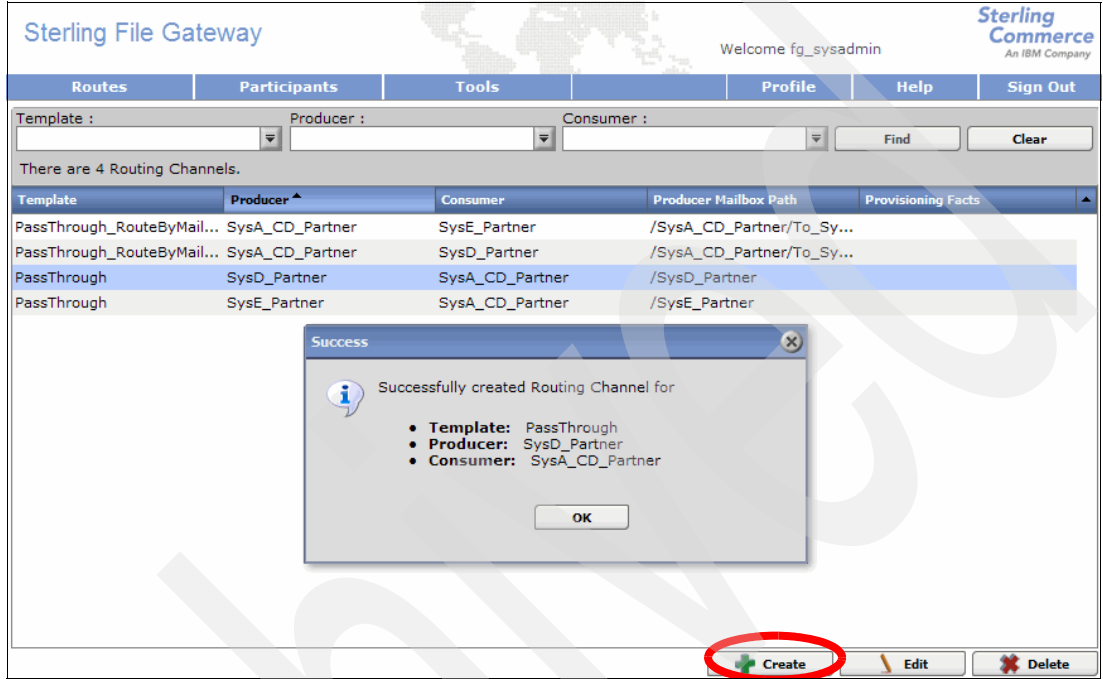


Figure 6-69 Create routing channel

3. Select the values listed in Table 6-6 and click **Save**.

Table 6-6 Values for routing channel

Parameter	Value
Template	PassThrough
Producer	SysD_Partner
Consumer	SysA_Partner

This route is now configured and ready for outbound file transfers. If a WebSphere MQ File Transfer Edition transfer is sent from the agent on SysD (SYSDAGT) into the WebSphere MQ File Transfer Edition bridge agent on SysC (SYSDBRIDGEAGT), into the root mailbox path for SysD_Partner, then the file will be routed to SysA_CD_Partner by initiating a Sterling Connect:Direct file transfer. The transfer will use the Connect:Direct server adapter on SysC (SysC_CD_SA) to send the file to the remote Connect:Direct node on SysA (SysA_CD). This transfer places the file into the C:\CDWindows_files\download directory on the destination machine SysA.

6.4 Testing the flows

This section describes the steps to run file transfers through the scenario in both the inbound direction (from the external Connect:Direct node on SysA to the internal WebSphere MQ File Transfer Edition node on SysD) and the outbound direction (from the internal WebSphere MQ File Transfer Edition node on SysD to the external Connect:Direct node on SysA).

6.4.1 Inbound scenario

This section details how to run the file transfer from an external partner using Sterling Connect:Direct to an internal partner using WebSphere MQ File Transfer Edition.

A file is sent from Company A's Connect:Direct node SysA_CD (on machine SysA) through a proxy server and into the Company B Sterling File Gateway via the Connect:Direct server adapter SysC_CDSA running on SysC.

The routing channel from SysA_CD_Partner to SysD_Partner uses the PassThrough_RouteByMailbox routing channel template, which means that the file momentarily arrives into SysA_CD_Partner's mailbox in Sterling File Gateway, into mailbox path /SysA_CD_Partner/To_SysD_Partner. The routing channel instantly routes this file on to internal partner SysD_Partner using the WebSphere MQ File Transfer Edition protocol as configured in Sterling File Gateway.

The custom business process CustomFileGatewayDeliveryFTE is started to initiate the WebSphere MQ File Transfer Edition file transfer from SYSDBRIDGEAGT to SYSDAGT. The business process constructs an xml message to define the parameters needed in the file transfer and places the message on the SYSDBRIDGEAGT command queue. WebSphere MQ File Transfer Edition takes over control at this stage and performs the file transfer to the agent on SysD. The file transfer status is reported back to the business process by placing messages on the REPLYMSGQ MQ queue.

Assuming that the transfer is a success, the file is written to the c:\downloads\ directory on machine SysD, and a success message is reported back to the business process when it completes.

Initiating transfer

To run the inbound scenario, follow the steps below:

1. Log on to machine SysA and start the CD Requester tool by going to **Start** → **All Programs** → **Sterling Commerce Connect Direct 4.5.01** → **CD Requester** (Figure 6-70).

Note: This scenario is demonstrated using the CD Requester tool and using the Graphical User Interface (GUI) Send/Receive File option. This could also be demonstrated using a process file to achieve the same file transfer.

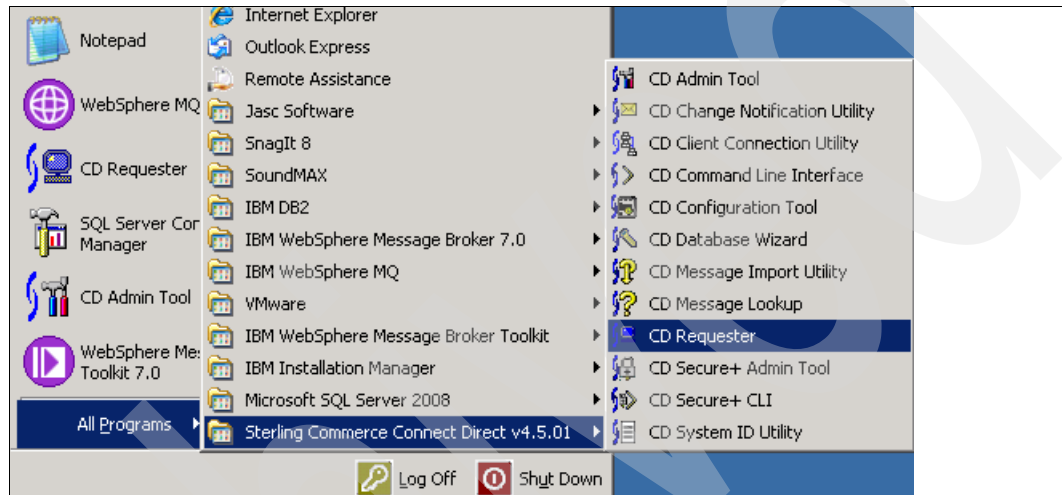


Figure 6-70 Start CD Requester

This starts the CD Requester (Figure 6-71).

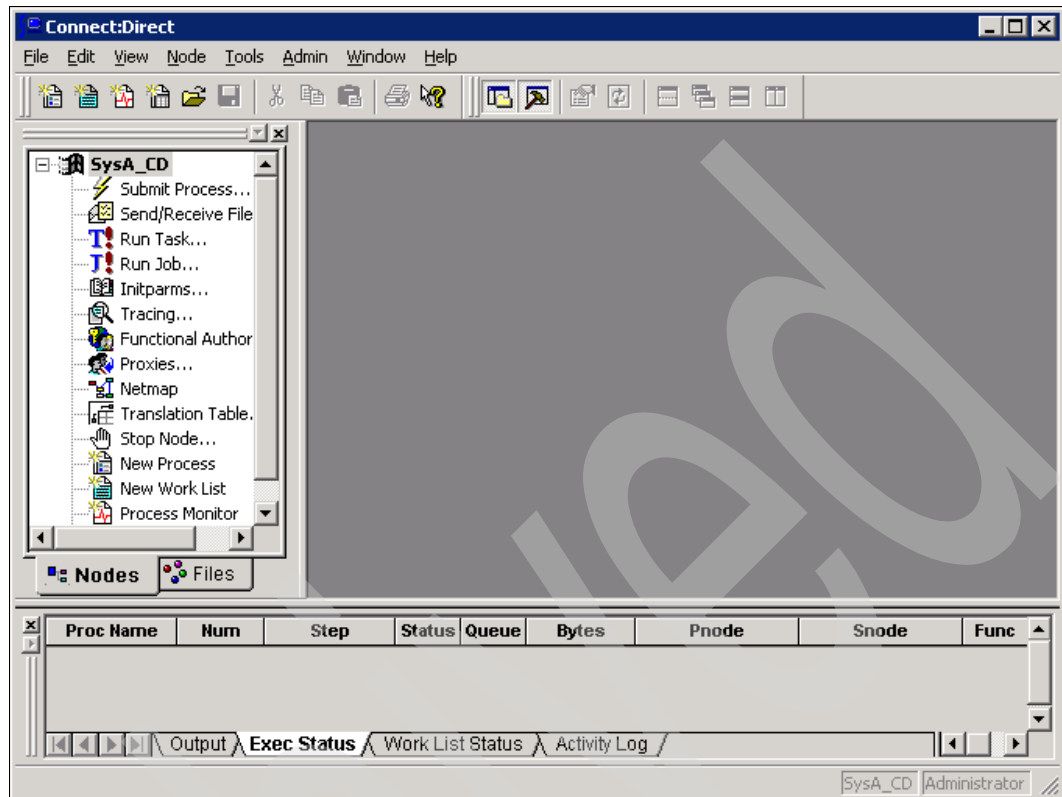


Figure 6-71 CD Requester

2. Right-click **SysA_CD** and click **Attach** to connect to the Connect:Direct node SysA_CD (Figure 6-72).

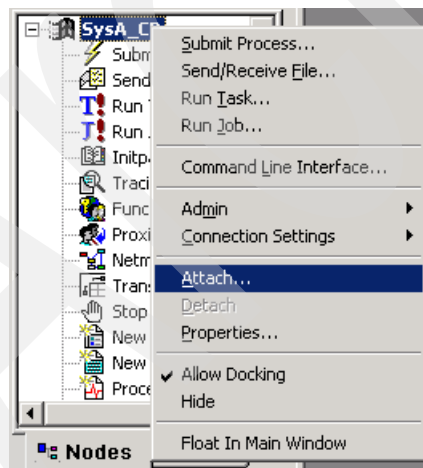


Figure 6-72 Connect to the SysA_CD Connect:Direct node

3. Enter the user name and password for the Connect:Direct node and click **OK** (Figure 6-73).

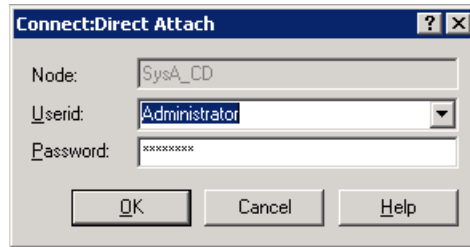


Figure 6-73 Enter user ID and password for the SysA_CD node

4. The icon next to SysA_CD changes from black-and-white to color to indicate that it has successfully connected to the Connect:Direct node.
5. Double-click **Send/Receive File**. This opens a window to enter details of the file to be transferred to SysD. Select **SNODE** from the SysC_CDSA drop-down menu. Select a file to send from the C:\CDWindows_files\upload directory and enter the file name without the directory path. (If no suitable file exists, create a simple text file in that directory first. We created a text file called SysA_to_SysD.txt for this demonstration and populated it with sample text.)

Enter the destination as:

/mailbox/To_SysD_Partner/SysA_to_SysD.txt

This mailbox path, along with the user name and password supplied in the transfer, will be used to put the file in the correct mailbox and cause the routing channel to forward the file onwards using WebSphere MQ File Transfer Edition.

Select a disposition of **RPL - Replace or create a new file** (Figure 6-74).

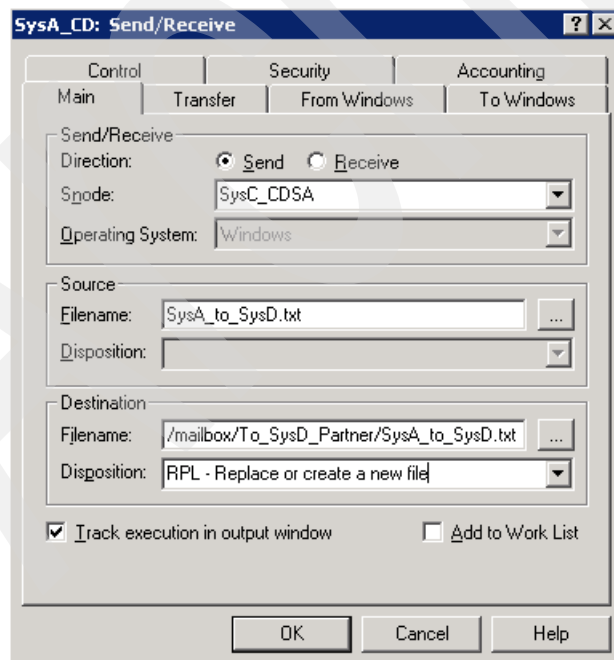


Figure 6-74 Values for initiating the transfer

6. Select the **Security** tab and enter the user name and password for the SNODE SysA_CD_Partner partner as it is defined in Sterling File Gateway. The user ID defines

the destination mailbox path, as it will only give access to SysA_CD_Partner's mailboxes. The password must match the one defined in Sterling File Gateway to allow access to that partner's mailbox (Figure 6-75).

The screenshot shows a dialog box titled "SysA_CD: Send/Receive" with a "Security" tab selected. The dialog has a tabbed interface with "Main", "Transfer", "From Windows", and "To Windows" tabs. Under the "Security" tab, there are three sub-sections: "Control", "Security", and "Accounting". The "Security" section contains two groups of fields. The first group, labeled "Pnode", has "Userid:" and "Password:" fields. The second group, labeled "Snode", has "Userid:" (containing "SysA_CD_Partner"), "Password:" (containing "xxxxxxxx"), "New Password:", and "Verify New Password:" fields. At the bottom of the dialog are "OK", "Cancel", and "Help" buttons.

Figure 6-75 Enter Sterling File Gateway partner credentials

- Click **OK** to start the transfer. When the transfer is started, it appears in the **Exec Status** window in the CD Requester tool. Click the grey box to the left of that window to show the status of the Sterling Connect:Direct file transfer between SysA_CD and SysC_CDSA Connect:Direct nodes (Figure 6-76).

Note: Sterling Connect:Direct has no knowledge of the ongoing routing to the consumer partner. As far as Company A is concerned, they have now sent their file to Company B and they have no need to know how it is routed or processed inside the internal company.

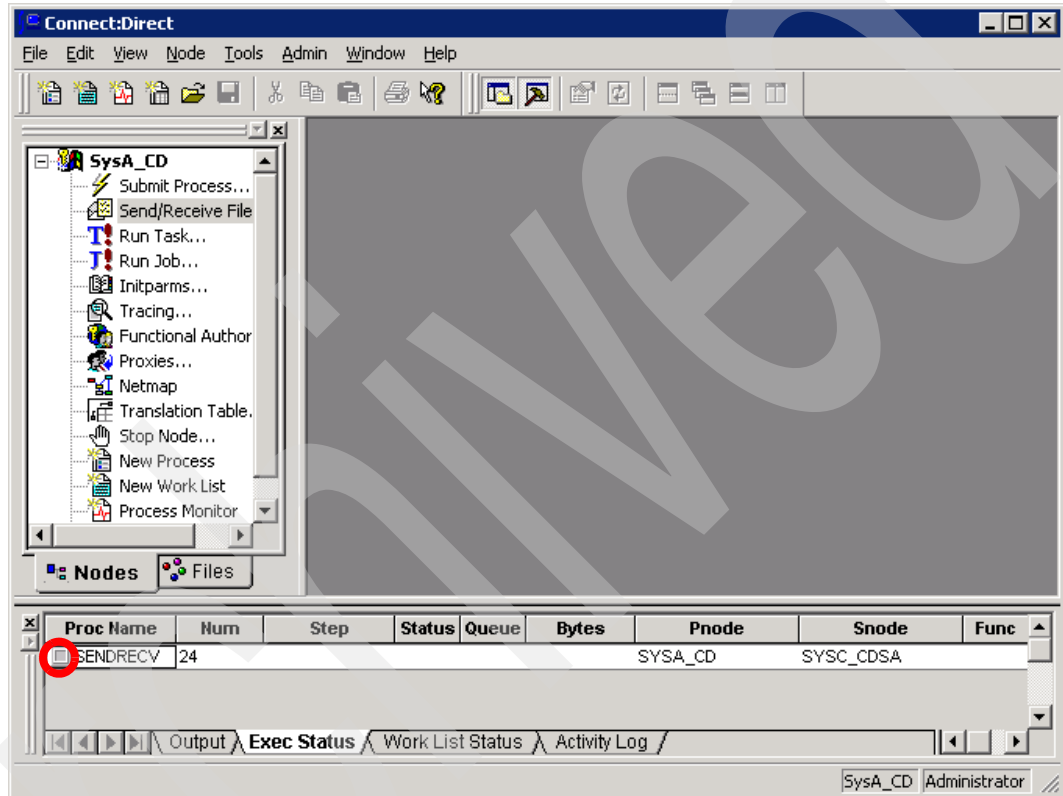


Figure 6-76 Click the grey box to open the transfer status

8. The transfer process status is displayed in the Process Execution Statistics window. Scrolling to the bottom of the window reveals a Copy Operation Successful message (Figure 6-77).

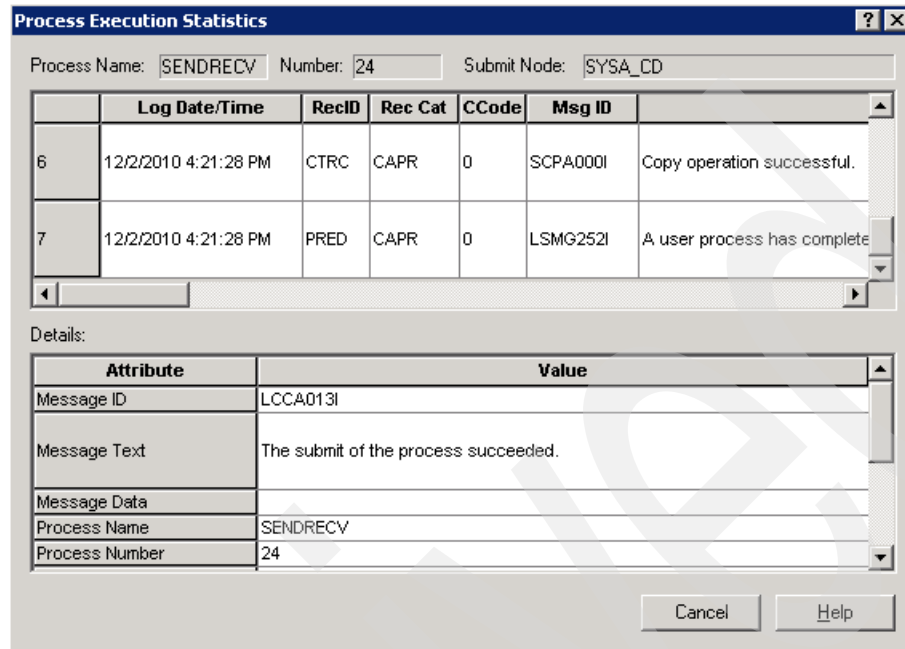


Figure 6-77 File transfer status

9. The most basic way to determine end-to-end success is to look in the C:\downloads\ directory on machine SysD to make sure that the file arrived successfully (Figure 6-78).



Figure 6-78 The transferred file arrived on the file system of the Company B SysD machine

10. The file transfer route can be tracked in Sterling File Gateway on SysC. Start Internet Explorer and go to:

<http://<servername>:<port>/filegateway/>

11. Log in using your administrator user ID and password (Figure 6-79). The default user ID for Sterling File Gateway is fg_sysadmin.

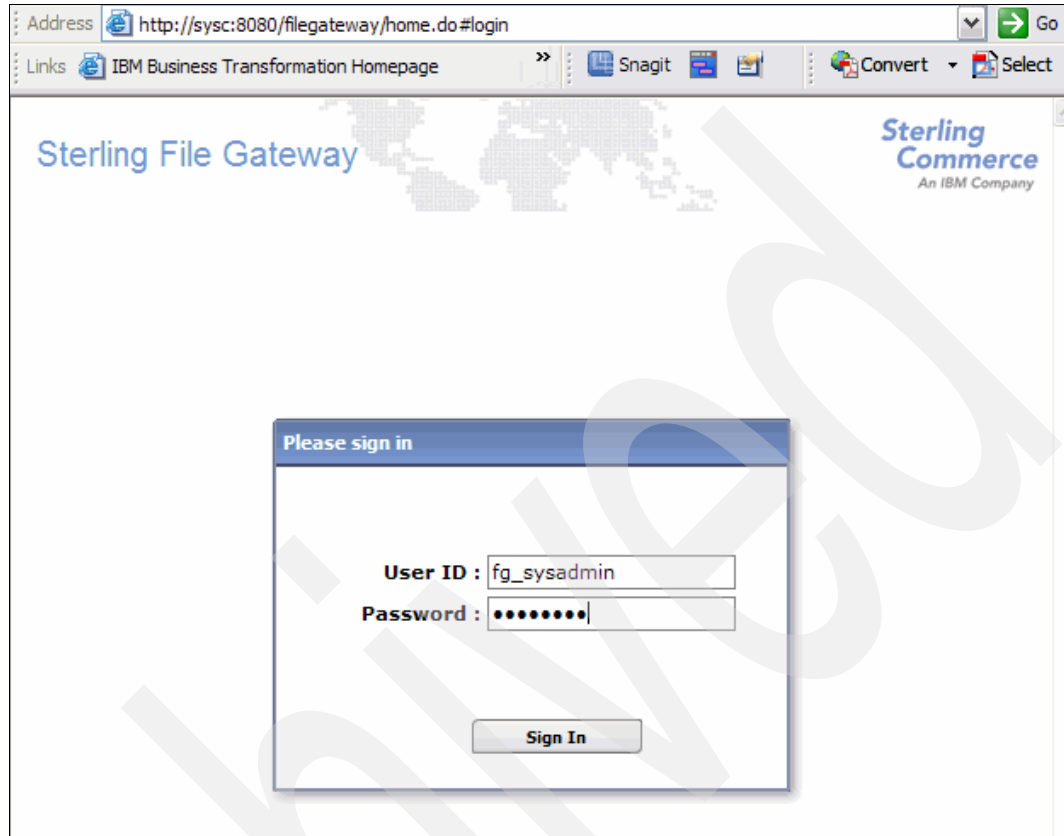


Figure 6-79 Log in screen for Sterling File Gateway

12. This brings up the main screen for Sterling File Gateway (Figure 6-80). Click **Find** to view all transfers through Sterling File Gateway.

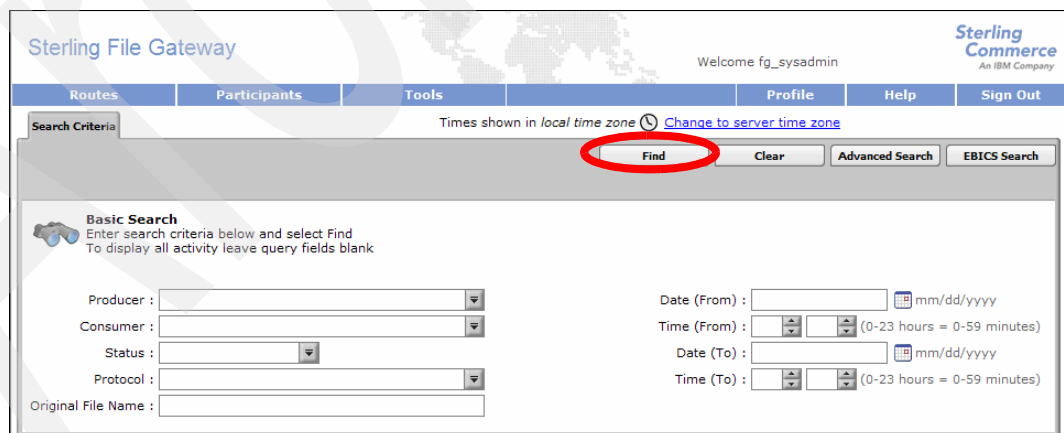


Figure 6-80 First screen when logged in to Sterling File Gateway

13. The Arrived File tab shows all Sterling File Gateway activity, and the file that you sent should be visible and have a status of Routed. The status is only present in Sterling File Gateway if the Sterling Connect:Direct section of the file transfer was successful from SysA_CD to SysC_CDSA (Figure 6-81).



Figure 6-81 Sterling File Gateway showing that the file was routed to SysD

14. Click the entry to open more details about the transfer (Figure 6-82).

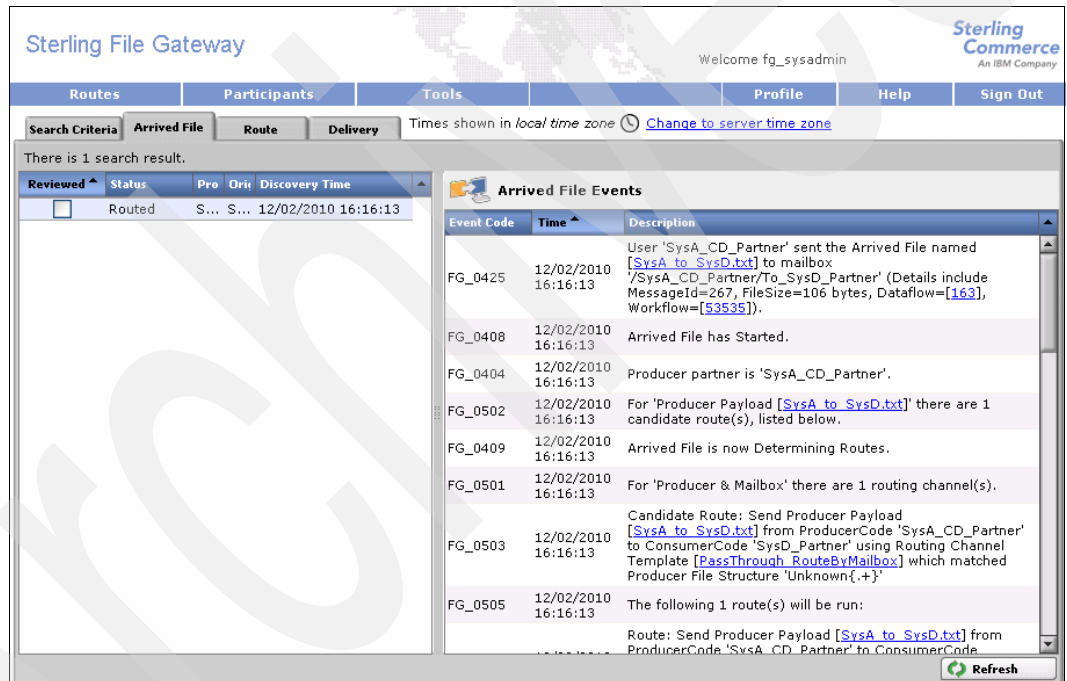


Figure 6-82 View more detailed status on the file transfer

By clicking the **Arrived File**, **Route**, and **Delivery** tabs on the page shown in Figure 6-82, it is possible to view detailed information about where the file was routed. The details on the Delivery tab will also contain a unique workflow number, which can be clicked to view the details of the steps executed by the custom business process, which can be very useful for debugging purposes if the file transfer fails.

- The final place to view the file transfer status is in the MQ Explorer on SysC or SysD to view the details of the WebSphere MQ File Transfer Edition section of the file transfer process. Start the WebSphere MQ Explorer by going to **Start → All Programs → IBM WebSphere MQ → WebSphere MQ Explorer**. When it has loaded, go to **Managed File Transfer → FTEQMGR → Transfer Log** and the file transfer should show a successful transfer (Figure 6-83).

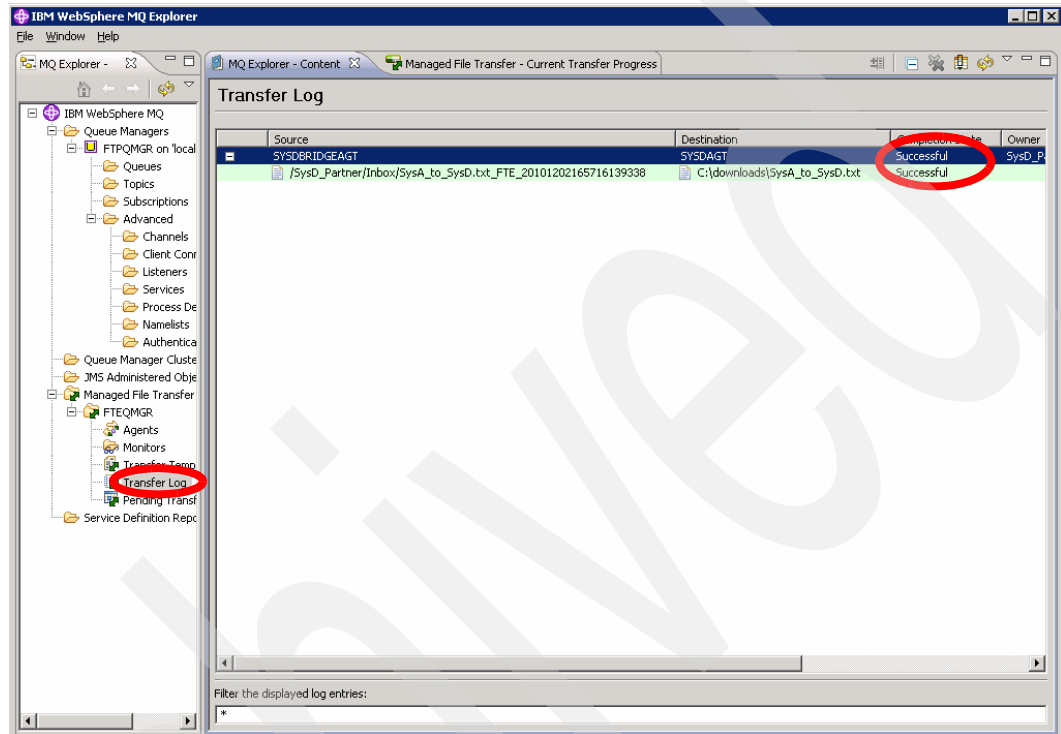


Figure 6-83 MQ Explorer showing success

6.4.2 Outbound scenario

This section details how to perform managed file transfer from an internal partner using WebSphere MQ File Transfer Edition to an external partner using Sterling Connect:Direct.

A file is sent from the Company B SYSDAGT WebSphere MQ File Transfer Edition agent (on machine SysD) to SYSDBRIDGEAGT WebSphere MQ File Transfer Edition bridge agent on SysC. SYSDBRIDGEAGT uses FTP to place the file onto SysD_Partner's mailbox in the root mailbox directory.

The routing channel from SysD_Partner to SysA_CD_Partner uses the PassThrough routing channel template. The routing channel instantly routes this file on to SysA_CD_Partner using the Sterling Connect:Direct protocol as it is configured in Sterling File Gateway.

The Connect:Direct server adapter takes over control at this stage and initiates a transfer from PNODE SysC_CDSA, running on SysC, to SNODE SysA_CD, which is the external Connect:Direct node owned by Company A. The routing is performed via a proxy server.

Assuming that the transfer is a success, the file is written to the C:\CDWindows_files\download directory on machine SysA and a success message is reported back to Sterling File Gateway.

Initiating transfer

To run the outbound scenario, follow these steps:

1. Log on to SysD.
2. Start the MQ Explorer by going to **Start → All Programs → IBM WebSphere MQ → WebSphere MQ Explorer**.
3. In WebSphere MQ Explorer, go to **Managed File Transfer → FTEQMGR → New Transfer** (Figure 6-84).

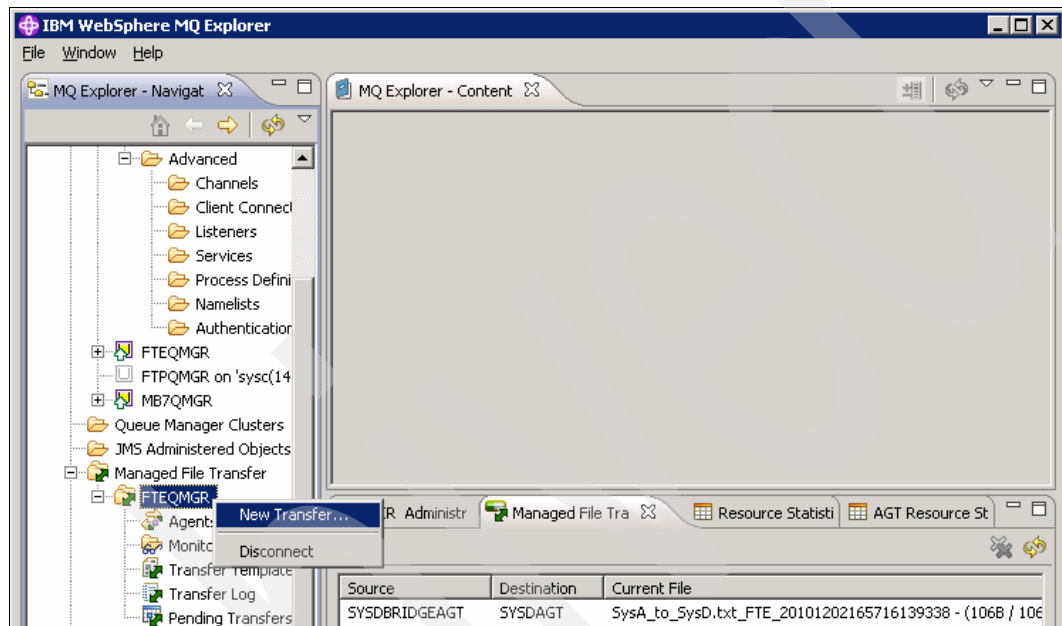


Figure 6-84 Start a new WebSphere MQ File Transfer Edition transfer

4. Perform these actions:
 - a. Enter the source agent as **SYSDAGT**.
 - b. Enter the path and file name of a file that you want to transfer. In this example we created a simple text file called `C:\temp\SysD_to_SysA.txt` and populated it with sample text.
 - c. Select the destination agent **SYSDBRIDGEAGT**, which is the WebSphere MQ File Transfer Edition bridge agent on SysC. Enter a destination directory of `/SysD_Partner/`.

These actions place the file in the root mailbox of SysD_Partner. The user name and password associated with SysD_Partner are stored in the `ProtocolBridgeCredentials.xml` file and are currently set to `admin/<your_SI_password>`. Because the user ID is set to be the administrator, we must specify which partner mailbox to place the file in (which is why we need `/SysD_Partner/` defined in the mailbox path). If the user ID and password in the `ProtocolBridgeCredentials.xml` file were set to `SysD_Partner/itso4you`, the user would only have access to that mailbox by default. This would be a more secure setup for production systems, but we have chosen to implement it using `admin/<your_SI_password>` for simplicity in this example scenario.

- d. Select the **Overwrite files on the destination file system that have the same name** option to eliminate the chance of a failure due to the file already existing in the mailbox. Figure 6-85 shows these settings.
- a. Click **Finish**.

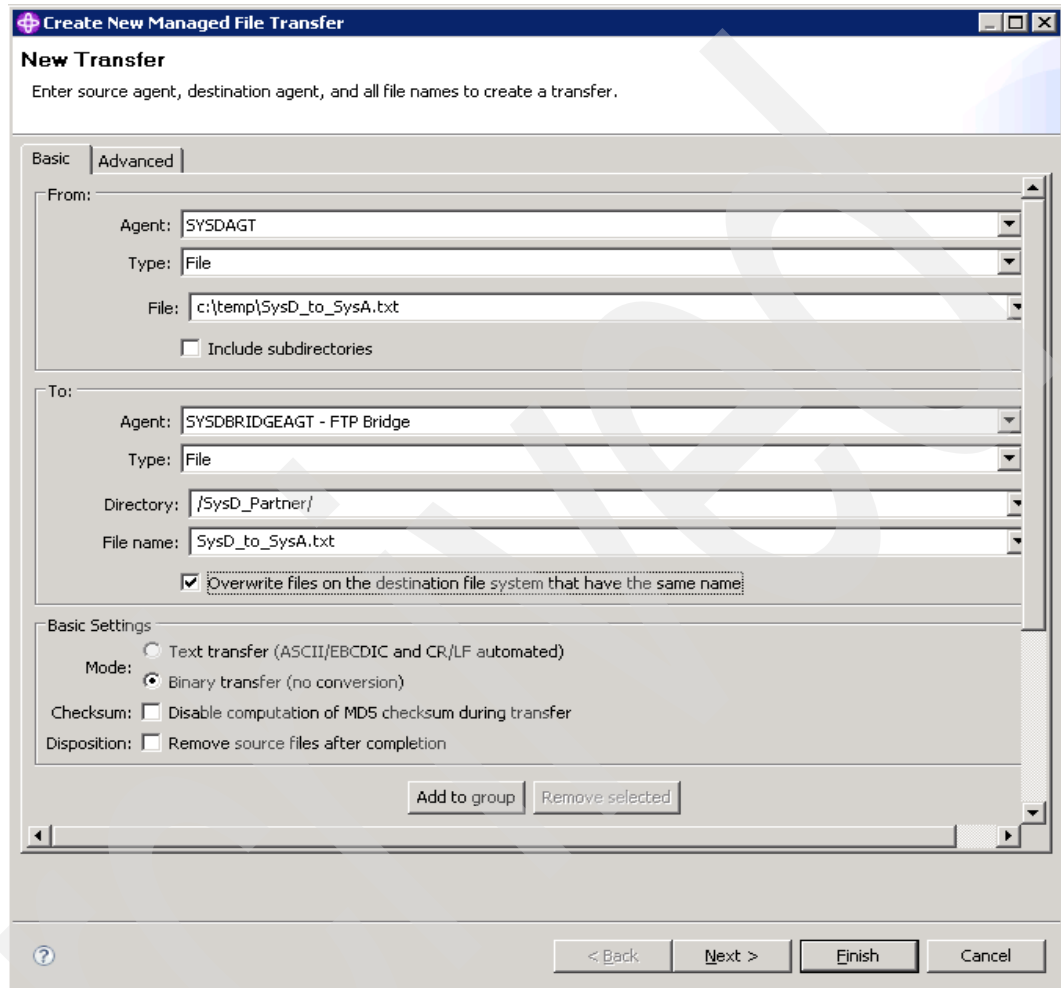


Figure 6-85 Settings for the WebSphere MQ File Transfer Edition transfer

- The status of the WebSphere MQ File Transfer Edition transfer from SYSDAGT on machine SysD to the bridge agent SYSDBRIDGEAGT on SysC can be viewed from within the MQ Explorer window. Click **Managed File Transfer** → **FTEQMGR** → **Transfer Log** (Figure 6-86).

Note that WebSphere MQ File Transfer Edition has no knowledge of the onward routing to the external partner SysA_CD_Partner of Company A. The status shown in the MQ Explorer window represents only the WebSphere MQ File Transfer Edition part of the transfer.

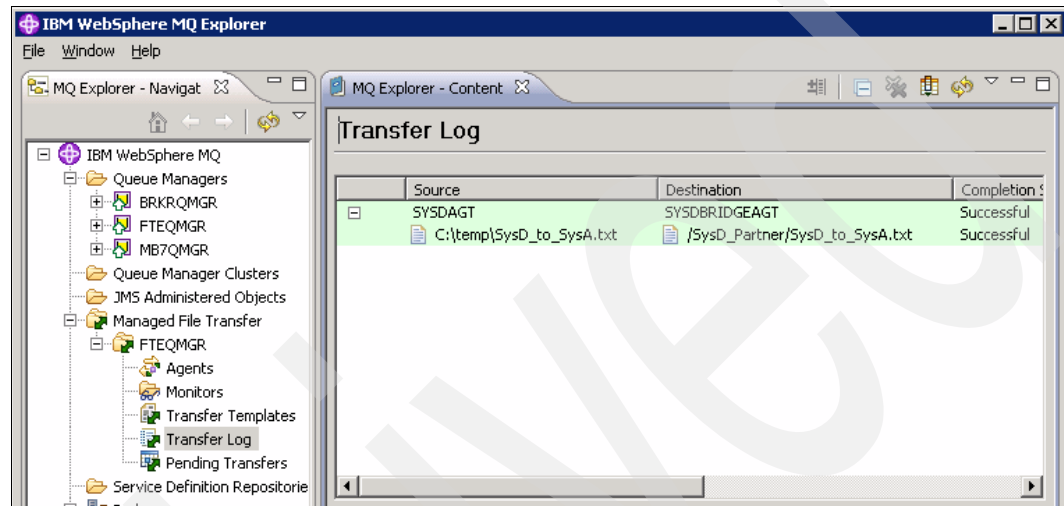


Figure 6-86 Successful transfer to the bridge agent on SysC

- The most basic way to determine end-to-end success is to look in the C:\CDWindows_files\download directory on SysA to make sure that the file arrived successfully to the external Sterling Connect:Direct partner (Figure 6-87).

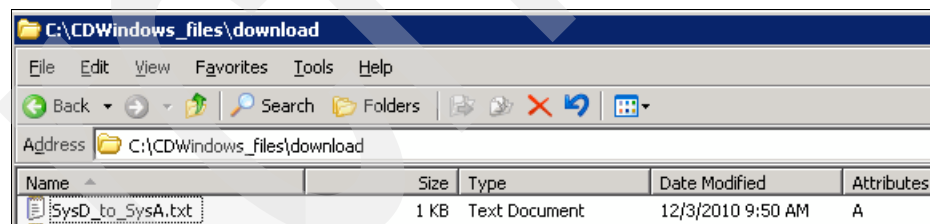


Figure 6-87 The transferred file arrived on the file system of Company A's SysA machine

- The file transfer route can be tracked in Sterling File Gateway on SysC. Start Internet Explorer and go to:

<http://<servername>:<port>/filegateway/>

8. Log in using your administrator user ID and password (Figure 6-88). The default user ID for Sterling File Gateway is fg_sysadmin.



Figure 6-88 Login page for Sterling File Gateway

9. This brings up the main screen for Sterling File Gateway (Figure 6-89). Click **Find** to view all transfers through Sterling File Gateway.

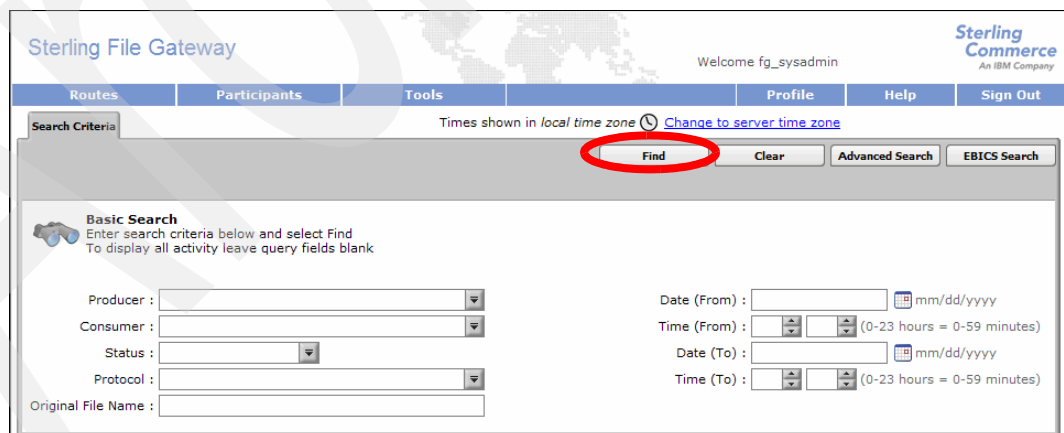


Figure 6-89 First page when logged in to Sterling File Gateway

10. The Arrived File tab shows all Sterling File Gateway activity, and the file that you sent should be visible and have a status of Routed. The status is only present in Sterling File Gateway if the WebSphere MQ File Transfer Edition section of the file transfer was successful from SYSDAGT to SYSDBRIDGEAGT (Figure 6-90).

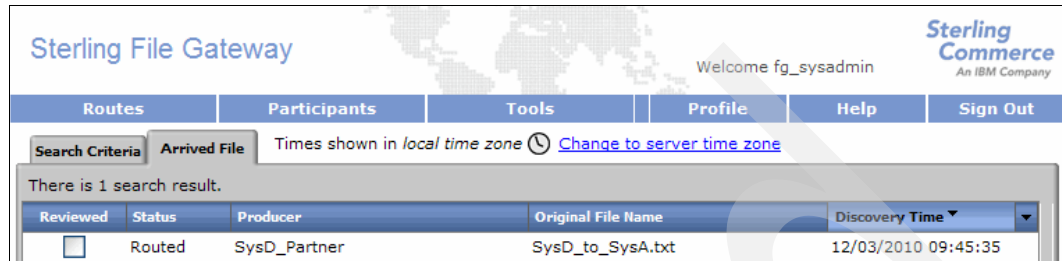


Figure 6-90 Sterling File Gateway showing that the file was routed to SysA

11. Click the entry to open more details about the transfer (Figure 6-91).

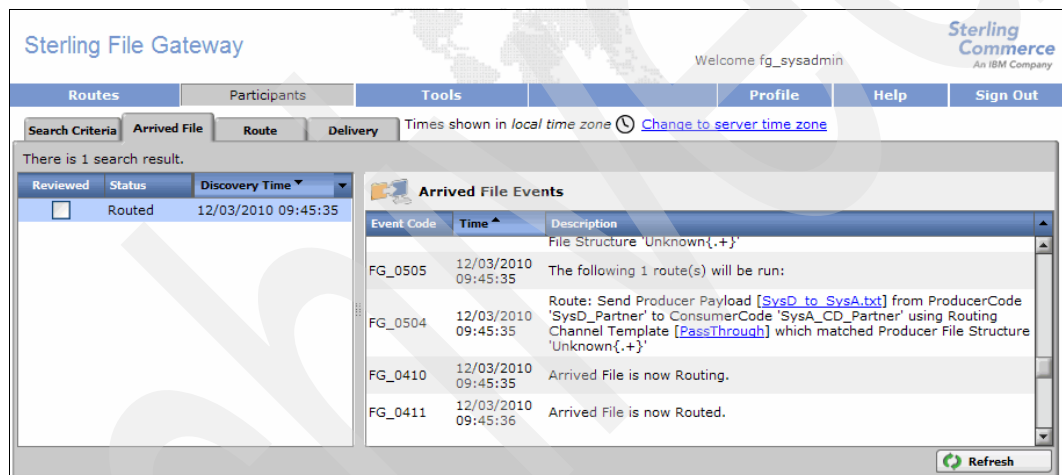


Figure 6-91 View more detailed status on the file transfer

If you click the **Arrived File**, **Route**, and **Delivery** tabs on the page, it is possible to view detailed information about where the file was routed. The details on the Delivery tab will also contain a unique workflow number that can be clicked to view the details of the steps executed by the custom business process, which can be very useful for debugging purposes if the file transfer fails.

6.5 Troubleshooting tips

If you experience difficulty running the scenario, see to Appendix C, "Troubleshooting" on page 379.

Archived

External transfers using IBM WebSphere Message Broker and IBM Sterling File Gateway

This chapter shows how you can enhance multi-enterprise file transfers with the addition of an enterprise service bus. Organizations often receive files that need transformation, parsing, and routing. By combining an enterprise service bus (ESB) with a managed file transfer solution, a file's journey into, throughout, and out of an organization to an external trading partner can function seamlessly. Many multi-enterprise transfers often use more than one protocol throughout the transmission of a single file. The ability to inter-operate with multiprotocols is becoming a standard for many organizations.

To demonstrate the use of an enterprise service bus and multiprotocol transfers for multi-enterprise file transfer, we integrated IBM Sterling File Gateway and IBM WebSphere Message Broker using IBM WebSphere MQ File Transfer Edition as the internal protocol for intra-enterprise managed file transfer. In this chapter, we highlight the use of the Sterling File Gateway myFileGateway for interaction with the external partner over Hypertext Transfer Protocol Secure (HTTPS). We also utilize Sterling File Gateway's ability to inter-operate with Secure FTP (SFTP) and WebSphere MQ File Transfer Edition.

This chapter includes the following topics:

- ▶ Solution overview
- ▶ Scenario details
- ▶ Configuring the solution components
- ▶ Testing the flows
- ▶ Troubleshooting tips

7.1 Solution overview

This scenario shows how to transfer a file into an organization over the internet through a secure, hardened interface in the DMZ, into a secure, internal network. The transfer is initiated using myFileGateway, a web browser-based interface that is provided with Sterling File Gateway for use by external trading partners. The myFileGateway URL is routed through the Sterling Secure Proxy to create a secure connection to Sterling File Gateway in the internal network. The external trading partner can upload and download files, search for activity, generate reports, subscribe for notifications, and change passwords using the myFileGateway interface.

After a file is uploaded using myFileGateway, it is placed in a mailbox and routed to an internally managed file transfer protocol to move throughout the organization. The routing of the file is enhanced through the use of an ESB, WebSphere Message Broker. WebSphere Message Broker then can perform actions such as extracting data from the file, routing the file based on content, sending the file or records of the file to multiple back-end applications, or working with databases.

With this type of configuration, organizations can interact with any protocol that is used by external trading partners while maintaining existing protocol standards internally.

7.1.1 Appropriate use

This scenario demonstrates the SFTP and HTTP communication between an external trading partner and Sterling File Gateway. The HTTP communication is shown by using the myFileGateway interface, which allows external trading partners to access a web page to upload and download files using Sterling File Gateway. In addition to the SFTP and HTTP application-layer protocols, you can use the following protocols with Sterling File Gateway to exchange files with external trading partners:

- ▶ FTP
- ▶ FTP/S
- ▶ SSH/SCP
- ▶ Sterling Connect:Direct
- ▶ AS2
- ▶ AS3
- ▶ Odette FTP

To modify the scenario to use one of the other protocols, you need to verify that the appropriate adapter is installed and configured in IBM Sterling B2B Integrator. You also need to configure partners and routing channels in Sterling File Gateway that might be needed.

You can vary the WebSphere MQ File Transfer Edition deployment to meet specific needs by using more sophisticated topologies. These topologies are described in detail in *Getting Started with WebSphere MQ File Transfer Edition V7*, SG24-7760.

7.1.2 Business value

Adding an ESB to multi-enterprise managed file transfer further extends an organization's ability to receive multiple data formats. The ESB allows an organization to perform actions such as breaking apart records to go to multiple back-end applications, reading to or writing

from databases, transforming data types, and matching data values. Organizations do not need to limit data formats or specify particular data types. The addition of an ESB provides the following benefits:

- ▶ A wide range of connectivity options and capabilities that allow an organization more options to send incoming data to various back-end applications.
- ▶ Simplification and productivity so that you can build a mediation once and use it multiple times with little to no modifications.
- ▶ Dynamic operational management to modify mediations on the fly in order to remain flexible in today's business climate.

Integrating Sterling File Gateway, WebSphere MQ File Transfer Edition, and WebSphere Message Broker gives organizations a robust multi-enterprise managed file transfer solution. A file coming into an organization passes through a secure mechanism in the DMZ to enable security and to terminate the external connection before passing the incoming data to secured, internal systems. This secure mechanism resides in the DMZ to protect against unauthorized access.

After the external partner is granted access through the DMZ, Sterling File Gateway extends Sterling B2B Integrator's ability to switch between various protocols by allowing for movement of large and high-volume transfers, with end-to-end visibility of file movement in a process-oriented manner. Sterling File Gateway then uses the WebSphere MQ File Transfer Edition managed file transfer backbone to move files to the ESB, WebSphere Message Broker, and back-end applications, and back to Sterling File Gateway.

This integration from outside an organization, through a DMZ, and into an organization's managed file transfer infrastructure featuring an ESB creates a universal architecture that meets the business needs of many departments. The flexibility and versatility of Sterling B2B Integrator and WebSphere Message Broker teamed with the partner interfaces and administration featured in Sterling File Gateway give organizations a way to remain flexible to meet changing business demands.

Sterling B2B Integrator features protocol switching, which allows for almost any protocol to be accepted from a trading partner while still allowing for organizations to maintain the opportunity to use a single, managed file transfer protocol internally. Sterling B2B Integrator paired with WebSphere Message Broker enables organizations to accept a wide variety of protocols and file types to meet external trading partners' needs. Additionally, Sterling B2B Integrator and WebSphere Message Broker make use of templates, which promotes reuse to reduce administration and development time.

This scenario provides the following additional benefits:

- ▶ Security
 - Real-time monitoring through a portal in Sterling File Gateway allows for IT and trading partners to gain visibility to in-flight file transfers.
 - Connections in the protected network can be configured with SSL using Sterling File Gateway, WebSphere Message Broker, and WebSphere MQ File Transfer Edition.
 - You can encrypt data in-flight and at rest in Sterling File Gateway.
 - Data transport security and data encryption is supported in Sterling B2B Integrator.
 - Sterling B2B Integrator features a secured mailbox repository to hold files.
 - Identity management, including authorization and authentication for trading partners, can be defined in Sterling B2B Integrator.

- ▶ Administration, operation, and logging
 - Having the ability to trace the file transfers end-to-end with Sterling File Gateway and WebSphere MQ File Transfer Edition reduces the resources that are required to troubleshoot file transfer failures and retries.
 - WebSphere MQ File Transfer Edition allows you to set up file transfers to occur at specified times or dates or to be repeated at specific intervals. File transfers can also be triggered by a range of system events, such as new files or updated files.
 - Sterling File Gateway provides the ability for external partners to view the state of their own transfers and the ability to initiate upload and download requests.
 - Sterling File Gateway uses auditing and reporting to provide metrics that verify regulatory compliance and adherence to service level agreements.
 - You can monitor file transfer activity in Sterling File Gateway on an exception basis using event management notifications from the event logging, which provides a complete audit trail of file transfer activities.
 - WebSphere MQ File Transfer Edition provides full logging of transfers at both the source and destination systems for internal transfers.
 - Reusable templates in Sterling File Gateway and WebSphere MQ File Transfer Edition reduce staff time to build and maintain file transfer processes.
 - Sterling File Gateway can intelligently route files based on sender, file name, file type, and file contents.

7.2 Scenario details

We provide two similar file transfer scenarios in this chapter. Both scenarios use an application layer protocol to communicate with an external partner. Data is transferred through a Sterling Secure Proxy in the DMZ to Sterling File Gateway. The data is then routed inside Sterling File Gateway to WebSphere Message Broker using WebSphere MQ File Transfer Edition to move the file between systems.

One scenario highlights the use of myFileGateway, a component that installs into a HTTP adapter in Sterling B2B Integrator, through a web browser. The other scenario uses the SFTP adapter in Sterling B2B Integrator to communicate with an external trading partner. The file arrives in Sterling File Gateway and is routed to WebSphere Message Broker Explorer for mediation and processing from myFileGateway. Depending on the outcome of the processing in WebSphere Message Broker, an error message can be sent back to myFileGateway or an order can be sent over SFTP to another trading partner.

With myFileGateway, the external partner manually uploads a XML file to the mailbox in Sterling File Gateway. The file is then directed through Sterling File Gateway routing channels to WebSphere Message Broker. To route the file to WebSphere Message Broker, Sterling File Gateway creates a WebSphere MQ message that is placed in the command queue of a WebSphere MQ File Transfer Edition protocol bridge agent using the Sterling B2B Integrator WebSphere MQ adapter. The protocol bridge agent then sends the file to the agent that is running in the WebSphere Message Broker execution group.

When the XML file arrives in WebSphere Message Broker, it is routed to a database retrieve node or database route node. Information is retrieved from the database based on the input criteria from the XML file. Inside the database route node, the retrieved data is evaluated and routed based on the appropriate response as defined in the node. In our case, this might result in a WebSphere MQ message being placed on a queue for other application processing, a response being sent over WebSphere MQ File Transfer Edition to

myFileGateway, or a file being sent over WebSphere MQ File Transfer Edition to Sterling File Gateway.

7.2.1 Solution components

This section describes the components that are associated with each product in this solution (Figure 7-1). Certain components require a specific configuration for the solution to work. We discuss the configuration steps that are required when necessary.

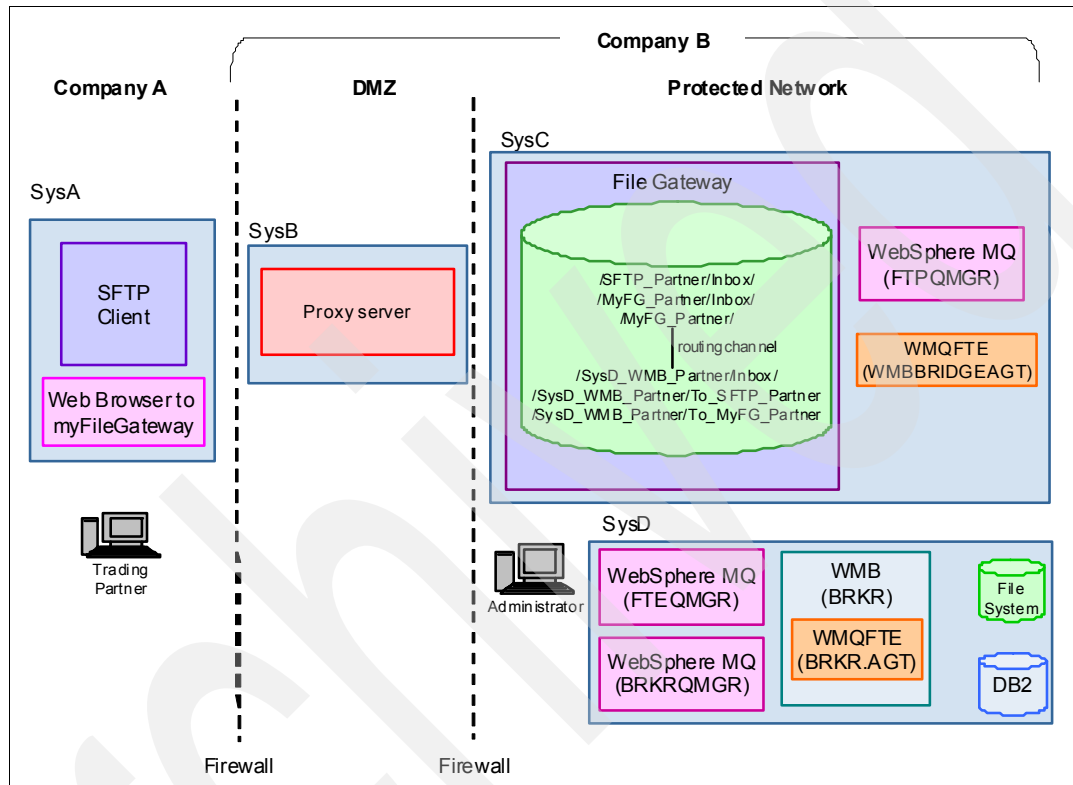


Figure 7-1 Solution components

Proxy server

The proxy server can be any proxy server or other DMZ-hardened security mechanism. This security piece validates that the external connection is coming from an approved domain over the port that is specified for the protocol used, terminates the external session, begins a secure session to continue back to the protected network, and authenticates users.

Sterling File Gateway

Sterling File Gateway routes files, incoming and outgoing, based on defined partners and their mailboxes. Sterling File Gateway uses Sterling B2B Integrator to switch protocols in this scenario. The mailboxes are created based on partner definitions. Sterling File Gateway can be administered through the following methods:

- ▶ The Sterling File Gateway Administration Console is a web-based GUI that allows administrators to create partners, routes, and channel templates.
- ▶ The Sterling B2B Integrator Administration Console is a web-based GUI that allows administrators to create business processes, create server adapters, and configure protocols.
- ▶ The Sterling B2B Integrator Import/Export Configuration allows administrators to configure objects for one Sterling File Gateway instance that can be exported and then imported into a different Sterling File Gateway instance.

In our scenario, we created additional mailboxes for each unique routing. Table 7-1 lists the use of each mailbox and to which mailbox Sterling File Gateway sends files.

Table 7-1 Sterling File Gateway mailboxes

External trading partner mailbox	Usage	File direction	Internal mailbox	Usage
/MyFG_Partner/	File is uploaded from myFileGateway and is placed in the root mailbox for the trading partner, MyFG_Partner.	Inbound	/SysD_WMB_Partner/Inbox/	The file is routed from the MyFG_Partner mailbox to this mailbox for delivery to WebSphere Message Broker using WebSphere MQ File Transfer Edition. The setup of this mailbox calls the WebSphere MQ File Transfer Edition protocol bridge agent to perform the transfer.
/SysD_WMB_Partner/To_MyFG_Partner/	Receives a file from WebSphere Message Broker through WebSphere MQ File Transfer Edition. This file is then routed to myFileGateway for the trading partner that originated the transfer, (MyFG_Partner).	Outbound	/MyFG_Partner/Inbox/	Receives a file from /SysD_WMB_Partner/To_MyFG_Partner/. This file remains in myFileGateway until the MyFG_Partner logs in to download the file and remove it from the mailbox.
/SysD_WMB_Partner/To_SFTP_Partner/	Receives a file from WebSphere Message Broker using WebSphere MQ File Transfer Edition. This file is then routed to a defined SFTP partner (SFTP_Partner).	Outbound	/SFTP_Partner/Inbox/	The file received from /SysD_WMB_Partner/To_SFTP_Partner/ remains in this mailbox until the SFTP client connects to receive the file.

myFileGateway

myFileGateway is the partner interface that is featured in Sterling File Gateway. It is accessible through a web browser and allows trading partners to upload/download files, subscribe to event notifications, manage passwords, search and view file transfer activity, and generate reports about file transfer activity.

In our scenario, we use myFileGateway to upload a file to Sterling File Gateway. We initiate a file transfer using WebSphere MQ File Transfer Edition to send a file to WebSphere Message Broker. WebSphere Message Broker mediates the file and sends a file out to two different destinations, depending on the results of the mediation.

If a file contains employee information that does not match an employee entry in the database, an error message that is appended to the input file is returned to myFileGateway for the trading partner to review. The trading partner can then optionally choose to correct the submission file and upload it again for processing. The second file transfer outcome from WebSphere Message Broker sends a file using WebSphere MQ File Transfer Edition to Sterling File Gateway to route to an SFTP partner.

Sterling B2B Integrator

Sterling B2B Integrator is a transaction engine and set of components that is designed to run processes that you define and manage according to your business needs. The suite supports high-volume electronic message exchange, complex routing, translation, and flexible interaction with multiple internal systems and external business partners.

Sterling File Gateway is a separate product that runs within Sterling B2B Integrator. The file routing and protocol switching functionality that is configured in Sterling File Gateway is actually performed by Sterling B2B Integrator. Sterling B2B Integrator provides the following features:

- ▶ The ability to manage and grow trading partner communities
- ▶ Adapters for back-end applications
- ▶ Role-based data access and system operation
- ▶ Data transport security and data encryption support
- ▶ Digital signature support
- ▶ Identity management, including authorization and authentication

For our scenario, we use Sterling B2B Integrator's ability to switch between the external trading partner's protocol and WebSphere MQ File Transfer Edition. To achieve this switching, Sterling B2B Integrator uses the WebSphere MQ adapter, working through a business process, to write a message on the command queue of a WebSphere MQ File Transfer Edition agent. This process initiates the file transfer. When the transfer is initiated, Sterling B2B Integrator listens on a reply queue for status updates from WebSphere MQ File Transfer Edition.

WebSphere MQ Queue Manager (FTEQMGR)

WebSphere MQ Queue Manager (FTEQMGR) is the WebSphere MQ File Transfer Edition coordination queue manager. The coordination queue manager publishes status messages received from the agents showing the state of transfers and the agents' status.

WebSphere MQ Queue Manager (FTPQMGR)

WebSphere MQ Queue Manager (FTPQMGR) acts as the command and agent queue manager for the WebSphere MQ File Transfer Edition agent WMBBRIDGEAGT. This queue manager must be local to the protocol bridge agent.

WebSphere MQ Queue Manager (BRKRQMGR)

WebSphere MQ Queue Manager (BRKRQMGR) is a queue manager that is used to run WebSphere Message Broker. This queue manager is created by the Create Broker Wizard in the WebSphere Message Broker Explorer. This queue manager also acts as the command and agent queue manager for BRKR.AGT.

WebSphere MQ File Transfer Edition Server Agent (BRKR.AGT)

WebSphere MQ File Transfer Edition Server Agent (BRKR.AGT) is a WebSphere MQ File Transfer Edition agent that runs in the WebSphere Message Broker execution group. This agent is defined when message flows with WebSphere MQ File Transfer Edition nodes are deployed to the execution group. BRKR.AGT connects to the BRKRQMGR in bindings mode. This agent can receive files using the FTEInput node and can send files using the FTEOutput node.

WebSphere MQ File Transfer Edition Bridge Agent (WMBBRIDGEAGT)

WebSphere MQ File Transfer Edition Bridge Agent (WMBBRIDGEAGT) is a protocol bridge agent. This is a special agent that comes with WebSphere MQ File Transfer Edition Version 7.0.1 or later and requires a local queue manager. The protocol bridge agent cannot read or write a file to a file system without sending the file to a WebSphere MQ File Transfer Edition client or server agent.

WMBBRIDGEAGT connects to a local queue manager in bindings mode. WMBBRIDGEAGT is configured to communicate with Sterling File Gateway using FTP. When Sterling File Gateway is ready to send a file over the MQ FTE backbone, it uses its WebSphere MQ adapter to put a message on WMBBRIDGEAGT's command queue residing on FTPQMGR. This message instructs WMBBRIDGEAGT to send the file received from Sterling File Gateway to BRKR.AGT.

WebSphere Message Broker

A *broker* is an ESB that routes messages, converts protocols, transforms data, and emits events. It has input and output nodes for various protocols and applications. Each broker can have one or more execution groups. An *execution group* is a separate operating system process that provides an isolated runtime environment for a set of deployed message flows. Each message flow in an assigned execution group runs in a different thread pool.

In our scenario, we create a broker runtime environment called *BRKR*, with one execution group called *AGT*. This broker and execution group host a single message flow called *InboundFileTransferFlow*. This message flow receives a file from Sterling File Gateway, performs a mediation, and then delivers an output based on the results of the mediation.

WebSphere Message Broker Toolkit

The WebSphere Message Broker Toolkit is used to create the message flows and artifacts that are deployed to the runtime broker, BRKR.

DB2

The DB2 relational database is used to store sample data for the purpose of creating a file transfer scenario. This table is loaded based on a sample SQL that is provided with the sample used to build the WebSphere Message Broker flow. Refer to Appendix B, "Building the WebSphere Message Broker flow" on page 365, for more information. The sample information is queried and analyzed to determine output actions in WebSphere Message Broker.

WebSphere MQ Explorer

The WebSphere MQ Explorer is used to view and administer the WebSphere MQ queue managers and queue manager objects, such as queues, topics, and channels. WebSphere MQ Explorer is built on an Eclipse-integrated development environment. The Eclipse-based platform allows plug-ins to be added to the base platform.

WebSphere MQ File Transfer Edition Explorer

The WebSphere MQ File Transfer Edition Explorer is a plug-in to the WebSphere MQ Explorer. It is used to schedule file transfer requests and view the status of current requests. The tool includes a Transfer Log view that subscribes to the coordination queue manager to obtain audit information. The audit information is displayed in the Transfer Log view for every transfer that occurs in the given topology. Beginning in WebSphere MQ File Transfer Edition Version 7.0.3, WebSphere MQ File Transfer Edition Explorer also includes the ability to view the status of an agent.

WebSphere Message Broker Explorer

The WebSphere Message Broker Explorer is a plug-in to the WebSphere MQ Explorer. It is used to administer the broker runtime environment. It shows the real-time status of brokers, execution groups, and message flows and their properties. With this plug-in, these components can be started and stopped using the WebSphere MQ Explorer. There are other ways to preform these tasks, such as using the CLI or the configuration manager proxy API.

7.2.2 Inbound file transfer flow

Figure 7-2 shows the flow for this scenario when files are inbound to the organization from an external partner. Note that we refer to our external partner as *Company A*. Company B is the other trading partner and is our lab environment.

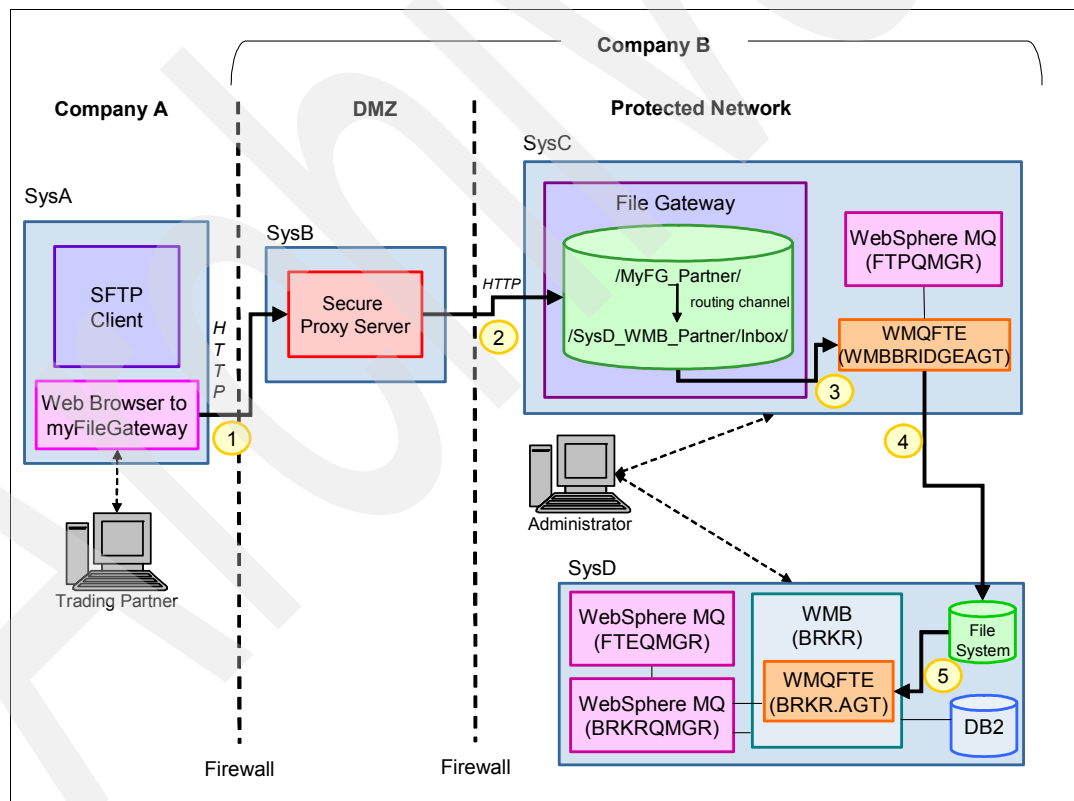


Figure 7-2 Inbound file transfer flow using WebSphere Message Broker

Ports through the internal firewall need to be opened to connect to the policies that are defined in Sterling File Gateway. The protected network is protected by appropriate firewall rules.

Figure 7-2 on page 253 shows how the file flows through the following steps:

1. The external partner, Company A, logs in to myFileGateway through a web browser with the provided partner user ID and password. When logged in, Company A initiates a file transfer by uploading a file through myFileGateway.
When Company A's web browser connects to myFileGateway, the Sterling Secure Proxy performs security checks and passes the file back to Sterling File Gateway.
2. The file arrives in Sterling File Gateway in the root mailbox for Company A, which in our scenario is /MyFG_Partner/. Based on routing channels that are defined for the mailbox in Sterling B2B Integrator, the file is passed to SysD_WMB_Partner.
3. Sterling File Gateway is configured to pass the file to partner, SysD_WMB_Partner, automatically using the WebSphere MQ File Transfer Edition WMBBRIDGEAGT bridge agent by placing a message on its command queue. WMBBRIDGEAGT uses FTP to retrieve the file from the Sterling File Gateway proprietary file system.
4. WMBBRIDGEAGT follows the instructions in the message placed on its command queue by Sterling File Gateway's WebSphere MQ adapter. The message tells WMBBRIDGEAGT to transmit the file to BRKR.AGT and place the file in a pre-configured location specified by the FTEInput Node.
5. After the file is written to the directory that is specified in the properties of the FTEInput node, the FTEInput node of the message flow reads the file and parses it according to the domain that is defined. The FTEInput node passes the parsed message to the other WebSphere Message Broker nodes for further processing. In our flow, the parsed information can be used to retrieve information from DB2.

7.2.3 Outbound file transfer flow

The outbound file transfer flow for this scenario originates in WebSphere Message Broker. The trading partner destination and protocol is based on the result of the mediation of the input file in WebSphere Message Broker. If the input file meets the criteria of an employee with 10 years of service, the file is designated as a service anniversary.

WebSphere Message Broker Explorer uses an FTEOutput node to send a file to the mailbox /SysD_WMB_Partner/To_SFTP_Partner/ in Sterling File Gateway. This mailbox is configured to route the file to a designated SFTP client trading partner to order a clock as recognition for the employee's service anniversary. The file remains in the mailbox on Sterling File Gateway until the SFTP client connects to retrieve the file. If the input file does not match any employee record in the employee database, WebSphere Message Broker routes the file back to the sender using myFileGateway with an FTEOutput node to place the file in the /SysD_WMB_Partner/To_MyFG_Partner/ mailbox. The file remains in myFileGateway until the trading partner logs on to download and review the file.

Figure 7-3 illustrates this flow.

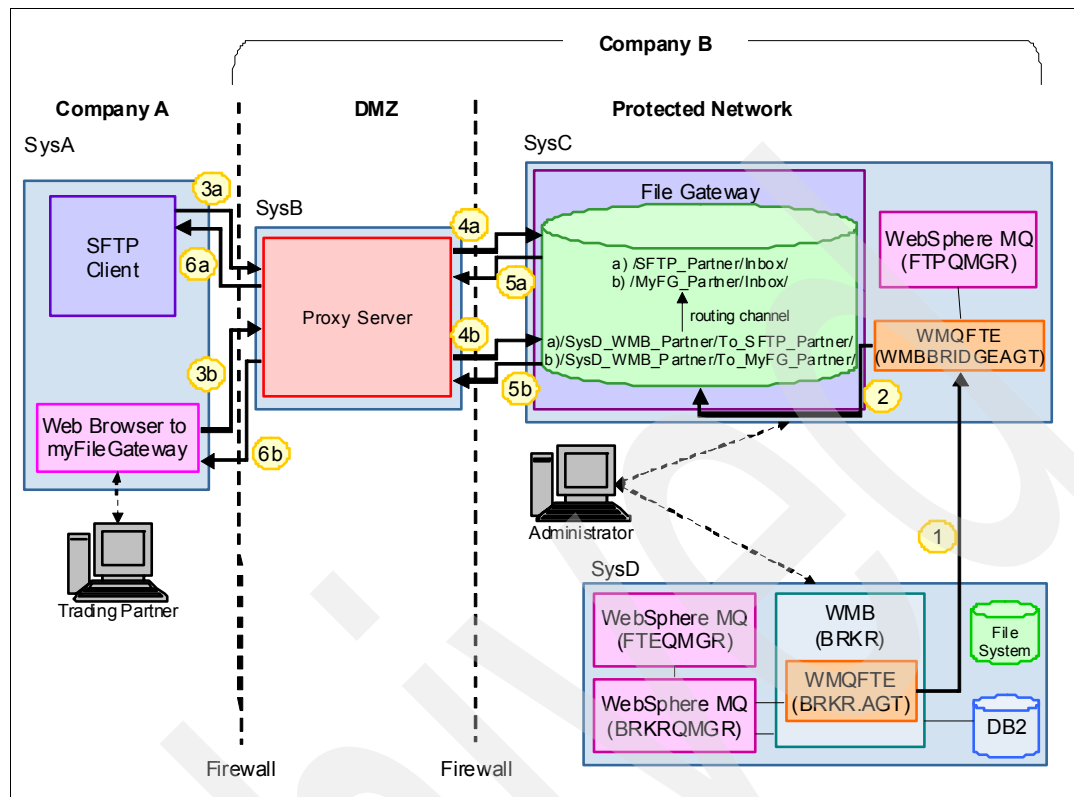


Figure 7-3 Outbound file transfer flow using WebSphere Message Broker

The sequence of steps for the outbound file transfer is as follows:

1. At the end of the mediation in the broker message flow, WebSphere Message Broker uses an output node to send a file to the WebSphere MQ File Transfer Edition protocol bridge agent, WMBBRIDGEAGT, on SysC.
2. Based on the results of the mediation, WebSphere Message Broker sends a message to either an awaiting back-end application or an output file. The scenario shows only the transmission of the output file to Company A. There are two possible output files that are transmitted.

If the input file contains criteria for an employee at the 10-year service anniversary, then WebSphere Message Broker uses an FTEOutput node to send the file from BRKR.AGT to WMBBRIDGEAGT, which is instructed to write the file out to the Sterling File Gateway /SysD_WMB_Partner/To_SFTP_Partner/ mailbox. This mailbox is configured to use a routing channel to send the file to the /SFTP_Partner/Inbox/ mailbox. The output file remains in the /SFTP_Partner/Inbox/ mailbox until the SFTP client trading partner connects to retrieve the output file.

If the input file given to WebSphere Message Broker does not contain matching information for an employee in the employee database, WebSphere Message Broker routes the file back to myFileGateway. The file is sent out of WebSphere Message Broker using a FTEOutput node that is configured to send the file to WMBBRIDGEAGT, which writes the file to the Sterling File Gateway /SysD_WMB_Partner/To_MyFG_Partner/ mailbox. The /SysD_WMB_Partner/To_MyFG_Partner/ mailbox has a routing channel that is configured to route the file in the /MyFG_Partner/Inbox/ mailbox, where the file remains until the MyFG_Partner user ID logs in and downloads the file from myFileGateway.

3. You can retrieve output files from their respective Sterling File Gateway mailbox using one of the following methods:

- For Company A's SFTP client trading partner to retrieve the file, it must connect to the SFTP server adapter that is running in Sterling File Gateway. For our scenario, we use a batch job to retrieve the file over SFTP. Company A runs the batch job at scheduled intervals. Example 7-1 shows the contents of the batch job that we created.

Example 7-1 SFTP batch job

```
C:\Putty\psftp.exe -pw itso4you -P 10052 SFTP_Partner@sysa -b  
SFTP_commands.txt
```

The SFTP batch job executes commands that include the following information and parameters:

- C:\Putty\psftp.exe** The SFTP executable, which varies based on the client SFTP program that you choose to use.
- pw** Passes the specified password configured in Sterling File Gateway. In our scenario, that password is itso4you.
- P** Specifies the port that is configured in the Sterling Secure Proxy for SFTP transmissions. We elected to use port 10052.
- SFTP_Partner@sysa** The user ID and host from which the Company A SFTP client connects. The SFTP_Partner user ID is specified in Sterling File Gateway and must be provided to the trading partner.
- b** Allows you to input an additional file. In our scenario, we use the SFTP_commands.txt file to issue the commands after the SFTP session begins. Example 7-2 shows the SFTP_commands.txt file that we used.

Example 7-2 SFTP_commands.txt

```
cd Inbox  
get ClockOrder.xml  
quit
```

The SFTP_commands.txt file changes from its default directory of /SFTP_Partner/ to /SFTP_Partner/Inbox/ to retrieve the file. The file name, ClockOrder.xml, is set at the time WebSphere Message Broker sends the file to Sterling File Gateway. The file name was provided to Company A, the SFTP client trading partner.

- A different department or individual at Company A logs on to myFileGateway with the MyFG_Partner user ID from a web browser.

4. Company A initiates the following methods to pull down the two possible output files, which are verified in the Company B DMZ by a Sterling Secure Proxy:

- The SFTP connection, shown in Figure 7-3 on page 255 as step 4a
- The HTTPS connection, shown in Figure 7-3 on page 255 as step 4b

After the Sterling Secure Proxy has validated the connection, communication with Sterling File Gateway in the Company B protected network is established.

5. The SFTP and HTTP protocols pull down the output file from their partner inbox in Sterling File Gateway.
 - a. Company A's batch job executes the `SFTP_commands.txt` file (Example 7-2 on page 256) to pull the file from the `/SFTP_Partner/Inbox/` mailbox in Sterling File Gateway.
 - b. The user or department logged in to `myFileGateway` with the `MyFG_Partner` user ID navigates to the Download Files tab to retrieve any files in the `/MyFG_Partner/Inbox/` mailbox. The user might need to refresh the Downloads display to show the latest transfers.
6. The output file is sent from Sterling File Gateway over the protocol that is used by Company A (Figure 7-3 on page 255) for step 6a or 6b. The output file passes through the Sterling Secure Proxy in the DMZ for Company B over the transport layer secured protocol and is written to the specified location on Company A's system.

7.2.4 Protocols

This scenario uses SFTP and HTTPS to transfer data between the external trading partner (called Company A in this scenario) and Company B. These protocols are used to communicate with Sterling File Gateway in Company B's protected zone. The protocol that is used for integration between Sterling File Gateway and WebSphere MQ File Transfer Edition is FTP and WebSphere MQ for message-level integration. The WebSphere MQ File Transfer Edition backbone uses WebSphere MQ to move files from one location to the next.

WebSphere Message Broker uses WebSphere MQ messaging and WebSphere MQ File Transfer Edition agents driven by `FTEInput`, `FTEOutput`, and `MQOutput` nodes in the message flow.

7.2.5 Security

In this scenario, we are concerned with securing the external protocols, SFTP and HTTPS. Authorization of outside protocols, domains, ports, and key exchanges is handled by the Sterling Secure Proxy in Company B's DMZ. This security mechanism can be any DMZ-hardened security mechanism, such as Sterling Secure Proxy, IBM WebSphere DataPower B2B Appliance XB60, IBM HTTP Server, or another publicly available reverse proxy security serves. In Company B's protected zone, Sterling B2B Integrator authenticates the partner user IDs against what is defined.

Transport security

The protocols that are used between the external partner and Sterling File Gateway are SFTP and HTTPS. SFTP refers to the SSH File Transfer Protocol, which is a network protocol that provides file transfer over any reliable data stream. SFTP assumes that it is run over a secure channel such as SSH, that the server has already authenticated with a client, and that the identity of the client is available to the server.

The HTTPS protocol is encrypted using Secure Socket Layer (SSL). HTTPS is typically used for web-browser-based sessions that contain sensitive information. The SSL encryption provides protection from being snooped during data transmission by exchanging certificates. Web browsers can choose to trust the certificates to establish the HTTPS session.

WebSphere MQ File Transfer Edition security

For any file transfer request, the agent processes require a certain level of access to local file systems. In addition, both the user identifier that is associated with the agent process and the

user identifiers that are associated with users performing file transfer operations must have the authority to use certain WebSphere MQ objects. Because the BRKR.AGT agent is running in the broker's execution group, it assumes the identity of the user running the broker process on SysD.

Commands are issued by users who might be in operational roles in which they typically start file transfers. Alternatively, the users might be in administrative roles, in which they can additionally control when agents are created, started, deleted, or cleaned (that is, when messages from all agent system queues are removed). Messages that contain command requests are placed on an agent's SYSTEM.FTE.COMMAND queue when a user issues a command. The agent process retrieves messages that contain command requests from the SYSTEM.FTE.COMMAND queue. The agent process also uses the following system queues:

- ▶ SYSTEM.FTE.DATA.*agent_name*
- ▶ SYSTEM.FTE.EVENT.*agent_name*
- ▶ SYSTEM.FTE.REPLY.*agent_name*
- ▶ SYSTEM.FTE.STATE.*agent_name*

WebSphere MQ File Transfer Edition supports finer-grained checking of users' authorities, which permits access to be granted (or denied) to specific product functions for each user. For example, you can choose which users have the authority to schedule transfer operations to happen at a future time. Because users issuing commands use these queues in different ways from the agent process, assign different WebSphere MQ authorities to the user identifiers or user groups that are associated with each. For more information, see *Using groups to manage authorities for resources specific to WebSphere File Transfer Edition* at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/group_resource_access.htm

The agent process has additional queues that can be used to grant users the authority to perform certain actions. The agent does not put or get messages on these queues. However, you must ensure that the queues are assigned the correct WebSphere MQ authorities, both for the user identifier that is used to run the agent process and for the user identifiers that are associated with users who are authorized to perform certain actions. The following authority queues are available:

- ▶ SYSTEM.FTE.AUTHADM1.*agent_name*
- ▶ SYSTEM.FTE.AUTHAGT1.*agent_name*
- ▶ SYSTEM.FTE.AUTHMON1.*agent_name*
- ▶ SYSTEM.FTE.AUTHOPS1.*agent_name*
- ▶ SYSTEM.FTE.AUTHSCH1.*agent_name*
- ▶ SYSTEM.FTE.AUTHTRN1.*agent_name*

The agent process also publishes messages to the SYSTEM.FTE topic on the coordination queue manager using the SYSTEM.FTE queue. Depending on whether the agent process is in the role of the source agent or the destination agent, the agent process might require authority to read, write, update, and delete files.

You can create and modify authority records for WebSphere MQ objects using WebSphere MQ Explorer. Right-click the object and select **Object Authorities** → **Manage Authority Records**. You can also create authority records using the `setmqaut` command.

Instead of granting authority to individual users for all of the various objects that might be involved, configure two security groups for the purposes of administering WebSphere MQ File Transfer Edition access control:

- ▶ FTEUSER
- ▶ FTEAGENT

Additionally, WebSphere MQ File Transfer Edition has security features to protect the files and file systems from unauthorized access. These features allow you to control who can read and write files that are transferred and to protect the integrity of files. You can achieve this security using the following methods:

- ▶ Manage authorities to access file systems.

The user ID running the agent process, in this case the user ID running the broker, must have access to the local file system to read or write files during file transfer. This managing of authority is controlled through the local operating system.

- ▶ Use sandboxes.

You can restrict the access of an agent to the file system by defining the `sandboxRoot` property in an agent's properties file. This property restricts the agent's access to a certain directory or to a certain area of the file system, the so-called *sandbox*. For more information about sandboxing, see:

<http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/sandboxes.htm>

- ▶ Use the agent's `commandPath` property.

The `commandPath` property in an agent's property file restricts the locations from which an agent can run commands. By default, the `commandPath` is empty, so an agent cannot call any commands. Take extreme care when you set this property because any command in one of the specified `commandPath` settings can be called from a remote client system that can send commands to the agent. For more information about this property, see:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/command_path.htm

- ▶ Configure SSL encryption for WebSphere MQ File Transfer Edition.

When transferred between WebSphere MQ File Transfer Edition agents, you can protect the file data by establishing SSL on the WebSphere MQ channel connections.

- ▶ Secure the authority to publish log and status messages.

Agents issue various log, progress, and status messages to the coordination queue manager for publication. You can secure the publication of these messages using WebSphere MQ security.

- ▶ Use the MD5 checksum.

MD5 checksum is the default setting on file transfers in WebSphere MQ File Transfer Edition. This setting keeps files from being manipulated after the transfer is initiated or while the transfer is in progress.

- ▶ Authentication.

Client FTE agents must be authenticated. The authentication can be performed by WebSphere MQ using SSL or exits.

Firewall security

Firewall configuration plays an important role in securing connections to and from external partners and in protecting the internal network. The external firewall must allow incoming requests from all of the trading partner's source IP addresses or range of IP addresses. You can configure firewall security in the firewall rules configuration. The method that you use to configure firewall rules depends on the model and type of firewall. The DMZ is a termination point at the edge of the protected network and typically is used to house internet-facing systems.

Setting up tight rules on the inner firewall is important for protecting internal systems. Typically, inner firewall rules are set to allow only traffic from a mediation server in the DMZ that terminates the connection from the internet. The mediation server then re-establishes the connection through the inner firewall to a system for which the files are destined or to a system that moves the files to a back-end application.

For this scenario, we need to open an SFTP port, 10052, and an HTTPS port, 10050, in the outside firewall. The inside firewall requires the SFTP port, 8119, and the HTTPS port, 10000, to be open. These ports will vary based on your mediation server configuration in your DMZ and your organization's security policies and practices.

7.3 Configuring the solution components

To configure the solution components for this scenario, we completed the steps described in the following sections:

- ▶ 7.3.1, "Software prerequisites" on page 261
- ▶ 7.3.2, "Configuration prerequisites" on page 261
- ▶ 7.3.3, "Creating the protocol bridge agent" on page 261
- ▶ 7.3.4, "Configuring a broker and execution group" on page 266
- ▶ 7.3.5, "WebSphere MQ File Transfer Edition with WebSphere Message Broker" on page 271
- ▶ 7.3.6, "Creating message flows with WebSphere MQ File Transfer Edition nodes" on page 274
- ▶ 7.3.7, "Creating a community in Sterling File Gateway" on page 284
- ▶ 7.3.8, "Creating routing channel templates in Sterling File Gateway" on page 288
- ▶ 7.3.9, "Enabling WebSphere MQ File Transfer Edition in Sterling File Gateway" on page 288
- ▶ 7.3.10, "Modifying FirstCommunity in Sterling File Gateway" on page 288
- ▶ Figure 7.3.11 on page 288
- ▶ 7.3.12, "Setting up a trading partner for WebSphere Message Broker" on page 289
- ▶ 7.3.13, "Creating a trading partner for myFileGateway" on page 296
- ▶ 7.3.14, "Configuring Sterling B2B Integrator for SFTP communication" on page 299
- ▶ 7.3.15, "Creating the trading partner for SFTP" on page 310
- ▶ 7.3.16, "Creating an inbound routing channel" on page 314
- ▶ 7.3.17, "Creating an outbound routing channel" on page 317
- ▶ 7.3.18, "Importing key certificate files in Sterling B2B Integrator for HTTPS" on page 319
- ▶ 7.3.19, "Configuring Sterling B2B Integrator and myFileGateway for HTTPS" on page 324

To create the multi-enterprise file transfer scenario using Sterling File Gateway, WebSphere MQ File Transfer Edition, and WebSphere Message Broker, we installed and configured a database, configured Sterling File Gateway, created a WebSphere MQ File Transfer Edition protocol bridge agent, created and configured a broker, and built and deployed a message flow. We describe these steps in this section.

7.3.1 Software prerequisites

This scenario uses the following software:

- ▶ WebSphere MQ V7.0.1
- ▶ WebSphere Message Broker V7.0.0.1
- ▶ WebSphere MQ File Transfer Edition V7.0.3
- ▶ Sterling File Gateway 2.1
- ▶ Sterling B2B Integrator 5.1 Build 5101
- ▶ DB2 Version 9.5
- ▶ Web browser
- ▶ SFTP client

7.3.2 Configuration prerequisites

Before configuring this solution, ensure that the following configuration prerequisites are met:

- ▶ Create the WebSphere MQ queue manager FTPQMGR, as described in “Creating the queue managers” on page 349.
- ▶ Create a database for use with the WebSphere Message Broker Explorer message flow.
- ▶ Create the following ports that are opened in the external firewall:
 - 10050: Used for HTTPS
 - 10052: Used for SFTP
- ▶ Create the following ports that are opened in the internal firewall:
 - 8119: Used for SFTP
 - 10000: Used for HTTPS
- ▶ Set up and configure a mediation server in the DMZ to validate external partners and to authenticate users. Also, set up the mediation server to end the external session and to begin an internal session that directs data back to the protected network.
- ▶ There are no file size limitations for this scenario. You are limited only by the disk space that is available on the servers that are receiving the transfers.
- ▶ The SFTP client in this scenario passes a user ID and password to Sterling B2B Integrator, which allows Sterling B2B Integrator to communicate with the SFTP client without having to import the client’s public key. Review your security policy and practices to determine the appropriate ways for a client to authenticate in your production environment. You might need to import the client’s public key into Sterling B2B Integrator, which we do not describe in this book.

Security prerequisites: Review your local security policy and practices to determine what is appropriate for your production environment. While the scenarios in this book do not implement security, it is important that you take security into consideration when implementing these scenarios in your own environment.

7.3.3 Creating the protocol bridge agent

To create and configure the WMBBRIDGEAGT agent on the server in the protected network, we assume that the FTPQMGR queue manager is implemented already. The bridge agent requires a local agent queue manager to connect to in bindings mode. In our scenario, the WMBBRIDGEAGT agent is created on SysC, the system hosting FTPQMGR and Sterling File Gateway.

To create the protocol bridge agent:

1. Create the agent using the **fteCreateBridgeAgent** command. Open a command console, and run the following command from the command line:

```
fteCreateBridgeAgent -agentName WMBBRIDGEAGT -agentQMGr FTPQMGR -bt FTP -bh  
sysc -btz US/Eastern -bm UNIX -bs1 en_US -bfe UTF8 -bp 8112
```

This command creates the bridge agent in bindings mode to the FTPQMGR queue manager.

This command uses the following parameters:

-agentName	Name of the agent.
-agentQMGr	Name of the agent queue manager.
-bt	Protocol type (FTP or SFTP).
-bh	Host name or IP address of the FTP server machine.
-btz	(Optional) FTP server time zone.
-bm	FTP server platform.
-bs1	Location of the FTP server adapter.
-bfe	FTP server encoding.
-bp	Port of the FTP server adapter.

The FTP server adapter behaves as a UNIX FTP server.

Additional resource: For a detailed description of the **fteCreateBridgeAgent** command, see the WebSphere MQ File Transfer Edition 7.0.3 Information Center at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/index.jsp?topic=/com.ibm.wmqfte.home.doc/help_home_wmqfte.htmfirst

The **fteCreateBridgeAgent** command also creates three files. Two MQSC script files are created with the commands that are required to define and to delete the agent's system queues. It also creates a credential XML file that you must modify in a subsequent step. Information about these files is shown in the command output in Example 7-3.

Example 7-3 Results of the fteCreateBridgeAgent command

```
BFGCL0277I: A credential XML file has been created. This file must be  
completed with credential details for accessing the protocol file server before  
the bridge agent can be brought into service. The file can be found here:  
'C:\Documents and Settings\All Users\Application  
Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSDBRIDGEAGT\ProtocolBridgeCredentials.xml'.  
.....  
.....  
BFGCL0069I: A file has been created containing the MQSC definitions to create  
your agent. The file can be found here: 'C:\Documents and Settings\All  
Users\Application  
Data\IBM\WMQFTE\config\FTEQMGR\agents\WMBBRIDGEAGT\WMBBRIDGEAGT_create.mqsc'.  
BFGCL0070I: A file has been created containing the MQSC definitions to delete  
your agent. The file can be found here: 'C:\Documents and Settings\All  
Users\Application  
Data\IBM\WMQFTE\config\FTEQMGR\agents\WMBBRIDGEAGT\WMBBRIDGEAGT_delete.mqsc'.
```

BFGCL0053I: Agent configured and registered successfully.

Make sure that at the end of the output you see that the agent is successfully registered.

If you see a message that the agent was configured but could not be registered, the coordination queue manager could not be contacted because it is not available or because your configuration parameters are not correct.

In this case, the agent can be started and transfer files, but it is not listed by the **fteListAgents** command or in the WebSphere MQ File Transfer Edition Explorer. The status messages of this agent are also not shown in the WebSphere MQ File Transfer Edition Explorer Transfer Log view.

The WebSphere MQ reason code that is issued with the error provides more information about the reason for the problem. You can find explanations for reason codes in the WebSphere MQ V7 Information Center at:

<http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>

Creating bridge agents: The **fteCreateBridgeAgent** command creates a bridge agent for a specific FTP server. You have to create a bridge agent for each FTP server to which you want to connect.

2. Create the bridge agent's MQ objects on the queue manager, FTPQMGR, using the script that was generated in the previous step. Run the **WMBBRIDGEAGT_create.mqsc** script from the command line using the **runmqsc** utility:

```
runmqsc FTPQMGR < C:\Documents and Settings\All Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\WMBBRIDGEAGT\WMBBRIDGEAGT_create.mqsc
```

Example 7-4 shows the output of this command. Ensure that the command completes with no errors.

*Example 7-4 Results of the **SYSDBRIDGEAGT_create.mqsc** command*

```
C:\Documents and Settings\fteadmin>runmqsc FTPQMGR < "C:\Documents and Settings
All Users\Application Data\IBM\WMQFTE\config\FTEQMGR\agents\WMBBRIDGEAGT\WMBB
IDGEAGT_create.mqsc"
5724-H72 (C) Copyright IBM Corp. 1994, 2009. ALL RIGHTS RESERVED.
Starting MQSC for queue manager FTPQMGR.
```

```
11 MQSC commands read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. Configure the bridge agent credentials.

The bridge agent user must be authenticated when the **WMBBRIDGEAGT** agent connects to the FTP server adapter. The authentication of the bridge agent user at the FTP server can be done based on user ID and password credentials or by using a public/private key pair. The **ftecreateBridgeAgent** command creates the **ProtocolBridgeCredentials.xml** file, in which the credential mapping for this specific agent is defined. In our scenario, we use the user ID and password credentials for the authentication of our local user admin at the FTP server adapter.

To configure the credentials:

- a. Navigate to the bridge agent's home directory:

```
C:\Documents and Settings\All Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\WMBBRIDGEAGT
```

- b. Edit the **ProtocolBridgeCredentials.xml** file.

- c. Insert the following credentials:

```
<tns:user name="SYSTEM" serverUserId="admin"
serverPassword="<your_SI_password>" />
<tns:user name="fteadmin" serverUserId="admin"
serverPassword="<your_SI_password>" />
```

Note: Replace `<your_SI_password>` with your admin password for Sterling B2B Integrator.

Your `ProtocolBridgeCredentials.xml` file should look as shown in Example 7-5.

Example 7-5 The ProtocolBridgeCredentials.xml file

```
<tns:credentials xmlns:tns="http://wmqfte.ibm.com/ProtocolBridgeCredentials"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://wmqfte.ibm.com/ProtocolBridgeCredentials
ProtocolBridgeCredentials.xsd ">

  <tns:serverHost name="sysc">
    <!-- Insert user elements here -->
    <tns:user name="SYSTEM" serverUserId="admin"
serverPassword="<your_SI_password>" />
    <tns:user name="fteadmin" serverUserId="admin"
serverPassword="<your_SI_password>" />
  </tns:serverHost>
</tns:credentials>
```

4. Start the bridge agent from the command line. Open a command console and enter the following command:

```
fteStartAgent WMBBRIDGEAGT
```

Check the `fteListAgents` command (Example 7-6). If the agent is registered successfully, the agent name displays as a return result.

Example 7-6 The fteListAgents command output

```
>fteListAgents
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED
Agent Name:           Queue Manager Name:   Status:
BRKR.AGT              BRKRQMGR              NO_INFORMATION
SYSCAGT               FTPQMGR               READY
SYSDAGT              FTEQMGR               READY
SYSDBRIDGEAGT (FTP Bridge)  FTPQMGR              READY
WMBBRIDGEAGT (FTP Bridge)  FTPQMGR              READY
```


5. Verify that the agent is running successfully with no errors by checking the contents of its log file:
 - a. Navigate to the following directory:


```
C:\Documents and Settings\All Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\agents\WMBBRIDGEAGT\logs\
```
 - b. Open the output0.log file. The file should have an entry at the bottom of the file (Example 7-7).

Example 7-7 WMBBRIDGEAGT output0.log content

```
***** Start Display Current Environment *****
Build level: V7.0.3 f000-EDP6-20101111-1659
Java runtime version:
  JRE 1.6.0 IBM J9 2.4 Windows Server 2003 x86-32 jvmwi3260sr7-20091214_49398
(JIT enabled, AOT enabled)
  J9VM - 20091214_049398
  JIT - r9_20091123_13891
  GC - 20091111_AA
The maximum amount of memory that the Java virtual machine will attempt to use
is: '1639' MB
ICU4J version: 4.4.1.0
Properties:
  agentDesc=, agentName=WMBBRIDGEAGT, agentQMgr=FTPQMGR, agentType=BRIDGE,
  coordinationQMgr=FTEQMGR
  coordinationQMgrChannel=SYSTEM.DEF.SVRCONN,
  coordinationQMgrHost=9.42.170.129
  coordinationQMgrPort=1414, protocolServerFileEncoding=UTF8,
  protocolServerHost=sysc
  protocolServerLocale=en_GB, protocolServerPlatform=UNIX,
  protocolServerPort=8112
  protocolServerTimeZone=US/Eastern, protocolServerType=FTP
Install Locations:
  com.ibm.wmqfte.product.root=C:\Program Files\IBM\WMQFTE
  com.ibm.wmqfte.product.config=C:\Documents and Settings\All
Users\Application Data\IBM\WMQFTE\config
***** End Display Current Environment *****
[23/11/2010 16:54:45:703 EST] 00000001 Agent      I   BFGAG0090I: This agent
has been configured as a protocol bridge FTE agent.
[23/11/2010 16:54:45:703 EST] 00000001 AgentRuntime I   BFGAG0058I: The agent
has successfully initialized.
[23/11/2010 16:54:46:453 EST] 00000001 AgentRuntime I   BFGAG0059I: The agent
has been successfully started.
```

Alternatively, with WebSphere MQ File Transfer Edition V7.0.3, you can view the status of the agent in WebSphere MQ File Transfer Edition Explorer (Figure 7-4). If the agent shows a *Ready* status, it is running with no errors.

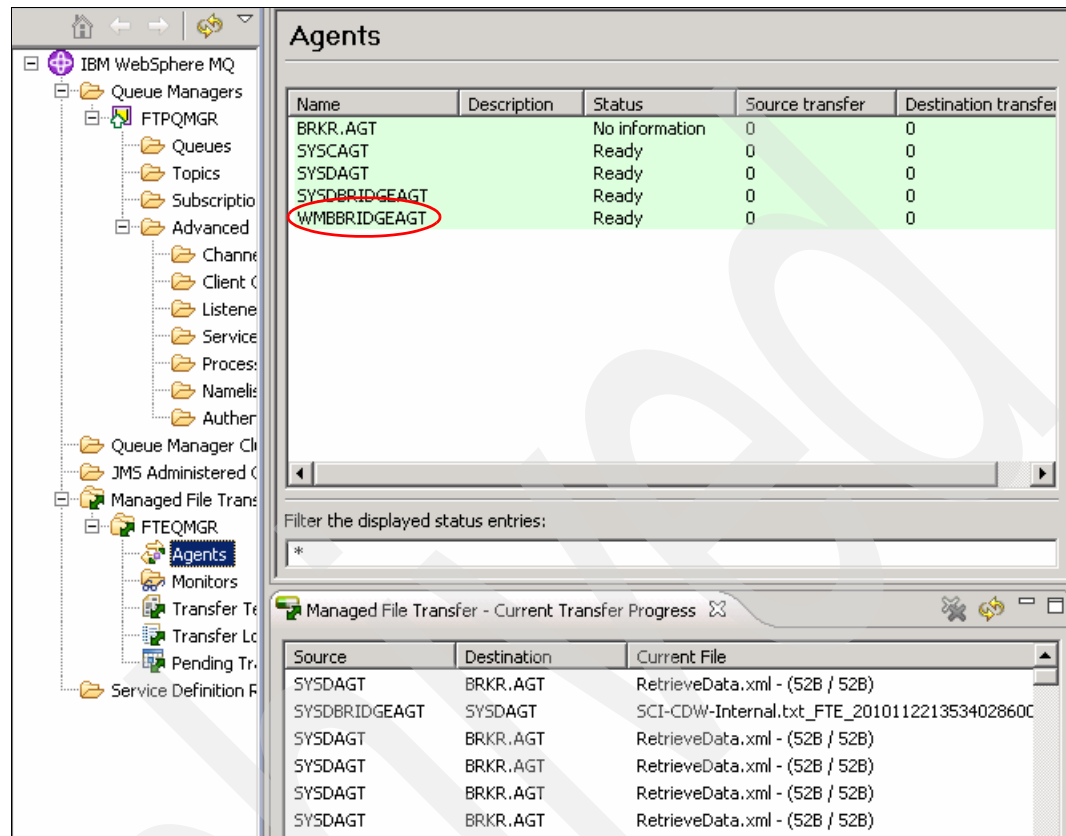


Figure 7-4 WMQFTE Explorer agent status

7.3.4 Configuring a broker and execution group

This section explains how to configure the broker and an execution group and how to set up the WebSphere Message Broker environment for WebSphere MQ File Transfer Edition. It is assumed that WebSphere Message Broker, WebSphere Message Broker Explorer, and WebSphere Message Broker Toolkit are already installed.

Installation documentation: You can find information about installing WebSphere Message Broker at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.doc/ax01445_.htm

This section assumes that you understand how to build and deploy message flows in WebSphere Message Broker Toolkit and how to use WebSphere Message Broker Explorer.

To create the WebSphere Message Broker environment:

1. Inside the WebSphere Message Broker Explorer, right-click, and select **Brokers** → **New** → **Local Broker** (Figure 7-5).



Figure 7-5 Broker menu

2. In the Create Broker Wizard (Figure 7-6), enter the new broker name. In our scenario, we use the name BRKR. Click **Next**.

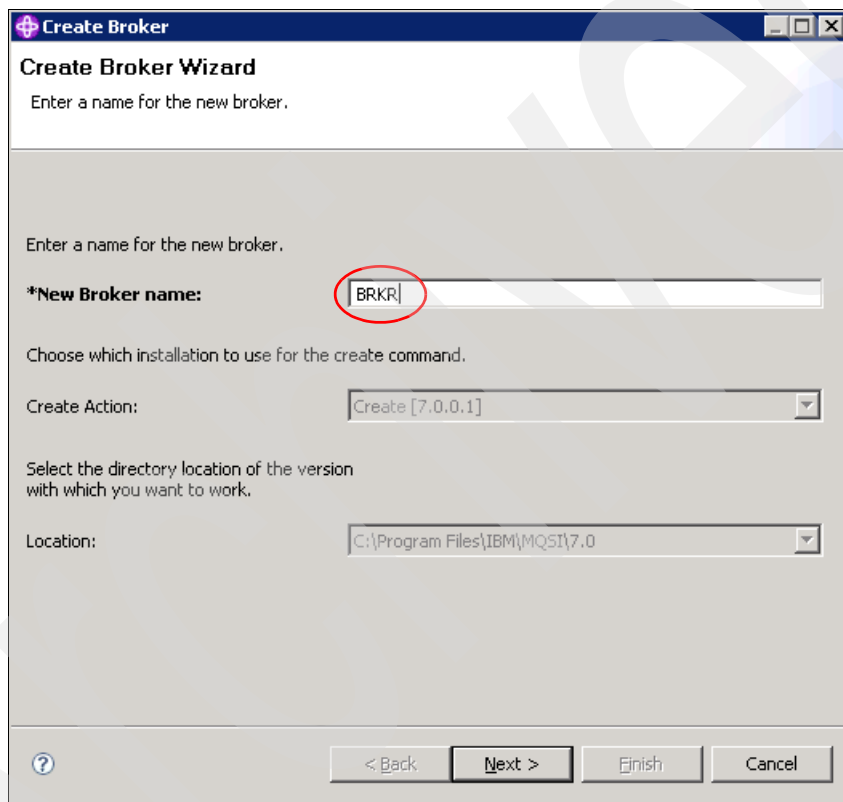


Figure 7-6 Create Broker Wizard - New Broker name

3. Enter the name of the queue manager to create for the broker (Figure 7-7). Also enter the location where the broker's queues will be defined and the name of the execution group in which to deploy flows. Leave the user name and password set to the defaults. Click **Next**.

In our scenario, we set the following queue manager and execution group:

- Queue manager: BRKRQMGR
- Execution group: AGT

User ID suggestion: The user ID that you use to log in while creating the broker is the user ID under which the broker runs. To avoid issues with WebSphere MQ publish/subscribe, the user ID needs to be 12 characters or less and should be a member of the mqbrkrs and mqm groups. Using a user ID of 12 characters or less adheres to MQ standards for user IDs on all other platforms. Failure to adhere to this suggestion can cause issues with sending files using the FTEOutput node and allowing the WebSphere MQ File Transfer Edition agent that is running in the broker execution group to publish status messages.

Figure 7-7 Create a new broker

The progress bar moves as the broker is defined and as the queue manager is created.

- When the broker creation is complete, a successful completion message displays (Figure 7-8). Click **Finish**.

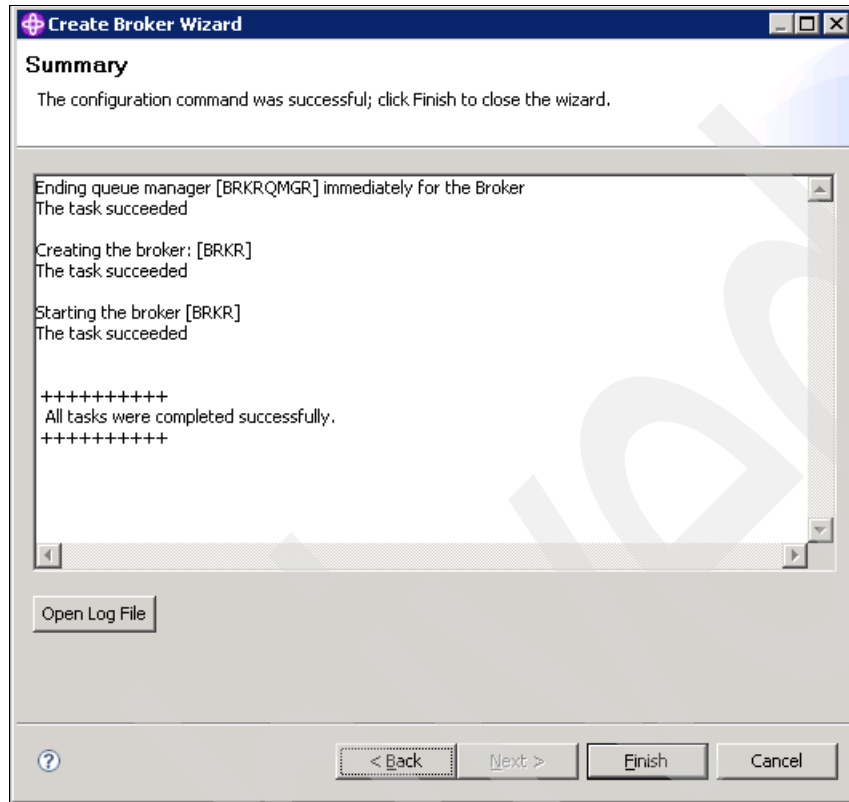


Figure 7-8 WebSphere MQ Explorer Summary

You can now see the new broker in WebSphere MQ Explorer in the list of brokers (Figure 7-9). You can also see that the execution group that you created for the new broker is started.

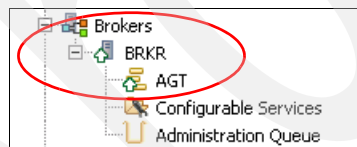


Figure 7-9 WebSphere MQ Explorer

5. BRKRQMGR is the second queue manager on SysD. The Create Broker Wizard did not allow us to set a port value different from the default TCP port 1414 value when it created BRKRQMGR. To keep from conflicting with the existing queue manager, BRKRQMGR needs a different TCP port value:
 - a. In WebSphere MQ Explorer, right-click **BRKRQMGR** and select **Properties** (Figure 7-10).

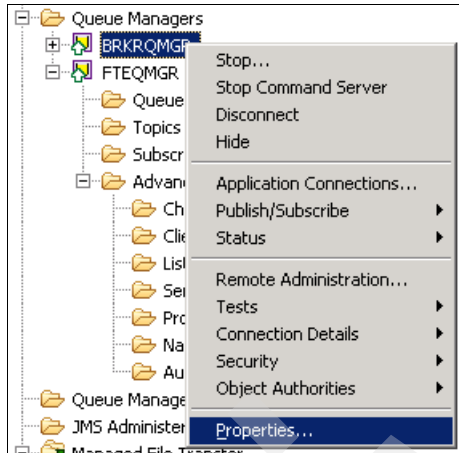


Figure 7-10 BRKRQMGR WebSphere MQ Explorer Menu

- b. In the Properties Menu, select **TCP** (Figure 7-11). Change the TCP Port value to a value that is not in use in your organization. We chose port 11414. Click **OK**.

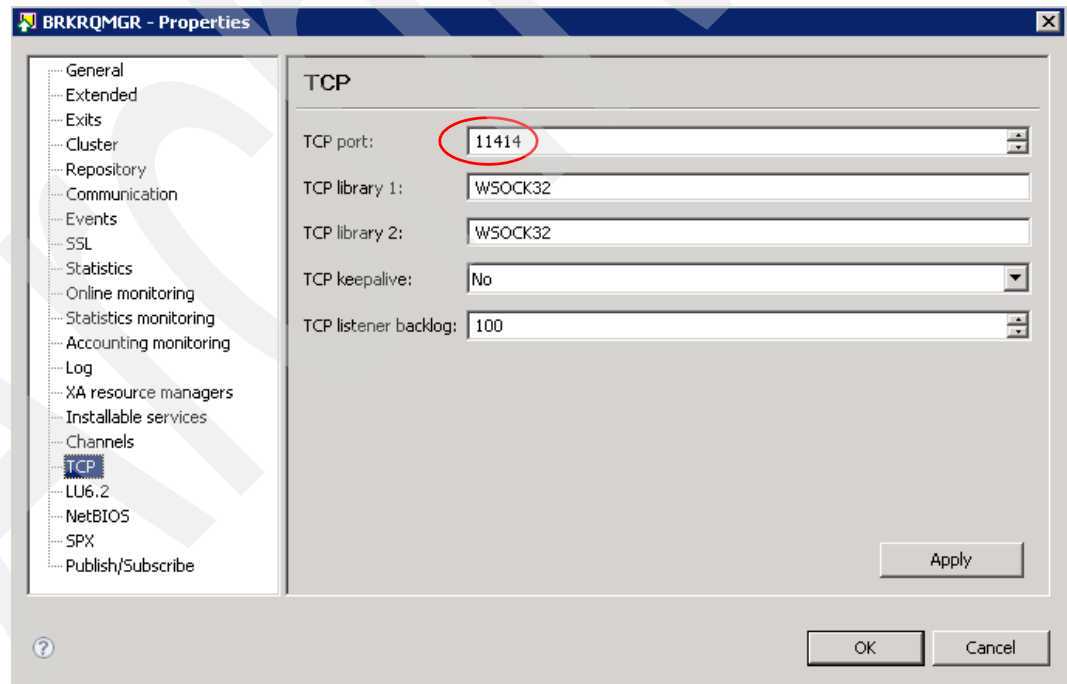


Figure 7-11 Modify BRKRQMGR TCP port

7.3.5 WebSphere MQ File Transfer Edition with WebSphere Message Broker

The FTEInput and FTEOutput nodes that are introduced in WebSphere Message Broker V7.0.0.1 provide seamless integration with an existing MQ FTE backbone network. Figure 7-12 shows a typical MQ FTE backbone network with a WebSphere Message Broker deployed with WebSphere MQ File Transfer Edition nodes.

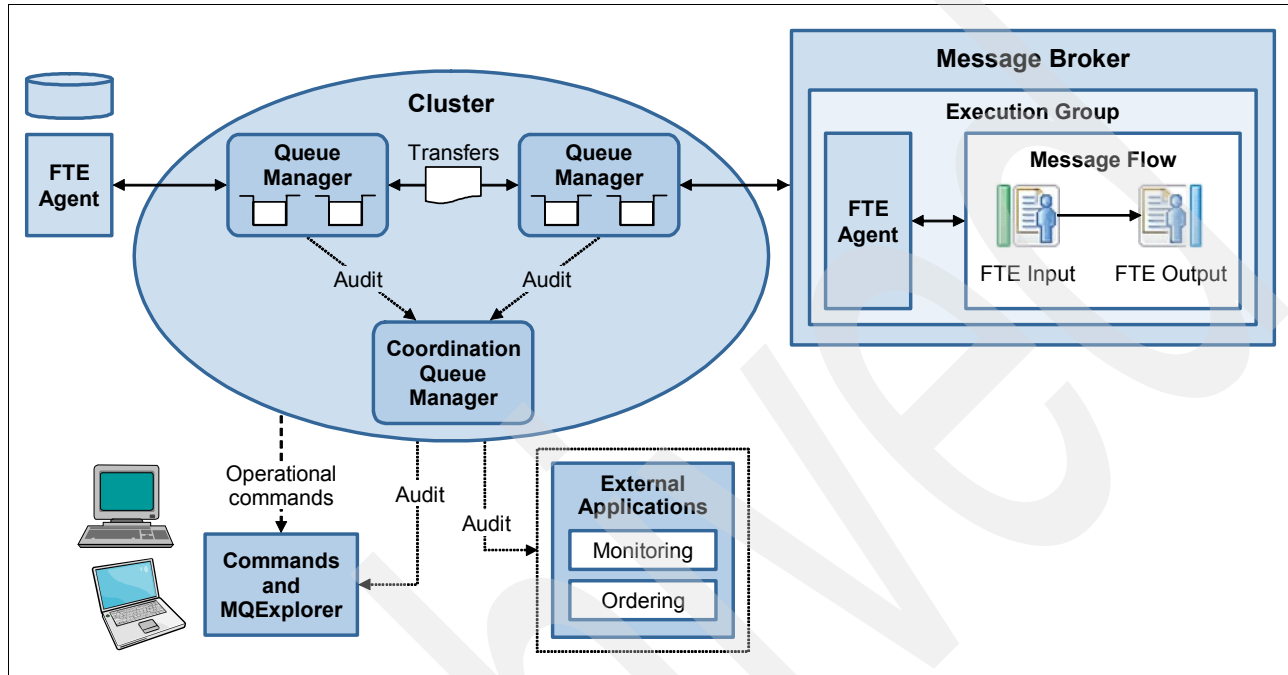


Figure 7-12 Integrating WebSphere MQ File Transfer Edition with WebSphere Message Broker overview

As depicted in Figure 7-12, a WebSphere MQ File Transfer Edition agent runs in each execution group that has deployed flows containing WebSphere MQ File Transfer Edition nodes. The agent is responsible for receiving and initiating all WebSphere MQ File Transfer Edition transfers.

You do not need to start or stop this agent. If a flow containing WebSphere MQ File Transfer Edition nodes is deployed, the agent is running. The agent is stopped only when the execution group is stopped. The broker queue manager is used as the queue manager for the agent.

Agent name

To send a file to a given execution group, users need to know the name of the agent that the broker creates. The agent name (BRKR.AGT in our case) is derived from *Broker.ExecutionGroup* and is not available for configuration after the broker and execution group are created. Ensure that the agent name meets the following criteria:

- ▶ The total name length is no more than 28 characters.
- ▶ The broker name is 12 characters or fewer (or at least unique in the first 12 characters). Broker names longer than this limit are truncated to form the agent name.
- ▶ The execution group names are 15 characters or fewer (or at least unique in the first 15 characters). Execution group names longer than this limit are truncated to form the agent name.
- ▶ The name must be in a valid format for generating the MQ Series queue name.

- ▶ The broker and execution groups do not contain any characters that are invalid for queue names.
- ▶ The *Broker.ExecutionGroup* tuples are all unique, even if the case is ignored.

Queue manager WebSphere MQ File Transfer Edition artifacts

When a broker is created, it creates all the required artifacts on the queue manager for the agent (Figure 7-13) and on the broker queue manager when it is the coordination queue manager. If the artifact creation fails due to the configuration of the system or permissions, the broker might not be able to create all artifacts. In this case, the user must create them in advance manually or create them using scripts. In our scenario, the BRKR broker created all the queues necessary for the BRKR.AGT agent. The coordination queue manager was created previously and needed no additional artifacts.

Database logger queues: When the broker creates the MQ artifacts, it also defines the queues for the database logger, because by default, the broker assumes that its queue manager is the coordination queue manager.

SYSTEM.FTE.AUTHADM1.BRKR.AGT	Local
SYSTEM.FTE.AUTHAGT1.BRKR.AGT	Local
SYSTEM.FTE.AUTHMON1.BRKR.AGT	Local
SYSTEM.FTE.AUTHOPS1.BRKR.AGT	Local
SYSTEM.FTE.AUTHSCH1.BRKR.AGT	Local
SYSTEM.FTE.AUTHTRN1.BRKR.AGT	Local
SYSTEM.FTE.COMMAND.BRKR.AGT	Local
SYSTEM.FTE.DATA.BRKR.AGT	Local
SYSTEM.FTE.EVENT.BRKR.AGT	Local
SYSTEM.FTE.REPLY.BRKR.AGT	Local
SYSTEM.FTE.STATE.BRKR.AGT	Local

Figure 7-13 WebSphere MQ queues for broker agent BRKR.AGT

Specifying a coordination queue manager

By default, the broker queue manager is the coordination queue manager. You can specify a different coordination queue manager using the WebSphere Message Broker Explorer or the `mqsichangeproperties` command.

In our scenarios, we define the coordination queue manager as FTEQMGR instead of the broker's queue manager, BRKRQMGR. Figure 7-14 shows the window for changing the coordination queue manager property. To find this window in WebSphere Message Broker Explorer, right-click the execution group and select **Properties** → **WebSphere MQ File Transfer Edition**.

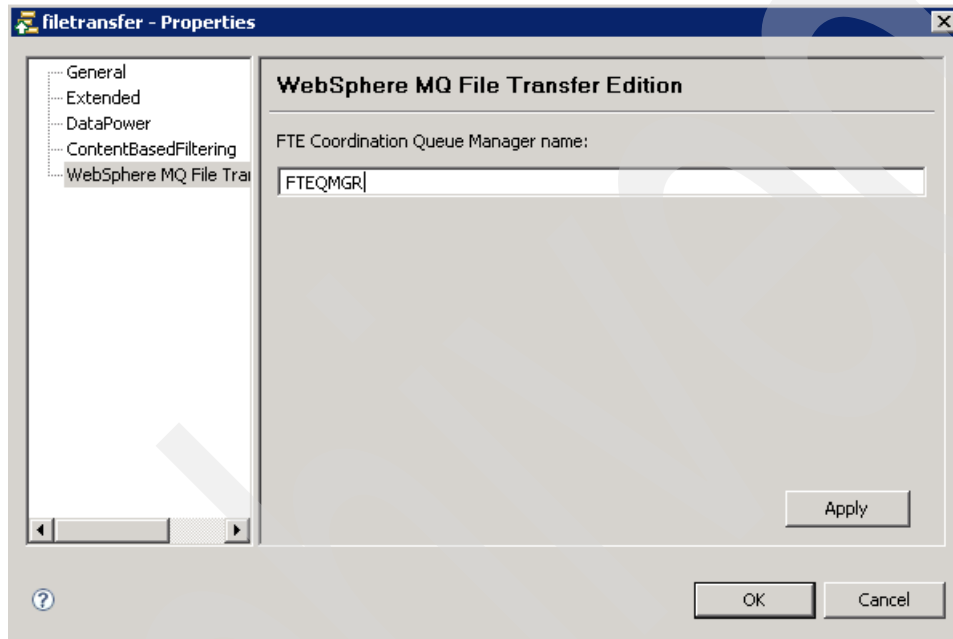


Figure 7-14 Execution group default properties window - Setting coordination queue manager

Directory for file transfers

WebSphere Message Broker uses a location in its work path to store transfers to remote agents. It uses another location as the default directory for received files. The high-level directory path for both locations is `workpath/common/FTE`.

On the broker system SYSD, we found the workpath to be:

```
C:\Documents and Settings\All Users\Application Data\IBM\MQSI\
```

7.3.6 Creating message flows with WebSphere MQ File Transfer Edition nodes

You can find the FTEInput and FTEOutput nodes in the WebSphere Message Broker Toolkit node palette in the File section (Figure 7-15).

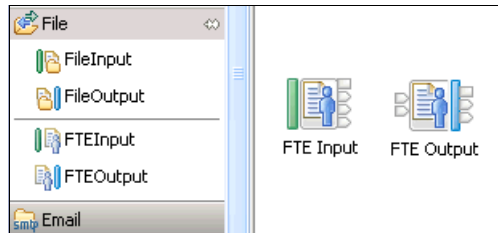


Figure 7-15 File drawer with WebSphere MQ File Transfer Edition nodes

You do not need to configure the WebSphere MQ File Transfer Edition code that runs in the broker. Operational tools in WebSphere Message Broker Explorer are provided to create transfers. At a high-level, the following steps use nodes to send or receive data across an existing WebSphere MQ File Transfer Edition network:

1. Create a flow that includes one of the WebSphere MQ File Transfer Edition nodes.
2. Configure the node.
3. For production purposes, change the coordination queue manager from the broker queue manager.
4. Deploy the flow.

Building a flow using the WebSphere MQ File Transfer Edition nodes: We describe the flow that we built for use in this scenario in Appendix B, “Building the WebSphere Message Broker flow” on page 365. The flow uses the configuration changes that we describe in this chapter.

Using the FTEInput node

The FTEInput node is used to extend WebSphere Message Broker Version 7.0 support for file processing through its integration with WebSphere MQ File Transfer Edition. Use this node in a flow that expects to receive files from a WebSphere MQ File Transfer Edition agent in the MQ FTE backbone network. The node is configured with properties like any other WebSphere Message Broker input node. The properties tell the node where to find the files to be processed, the file name pattern, and how to parse the data being received.

See the WebSphere Message Broker V7.0.0.1 Information Center for tables of the properties settings:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.doc/bc34034_.htm

Notice that the FTEInput node in the message flow in Figure 7-16 is waiting for files to arrive in the C:\FileTransfersInbound directory and is accepting any file name that is of a .xml file type. The properties page also indicates the disposition of the file after it is processed by the FTEInput node. Valid options are no action, add a time stamp, and delete.

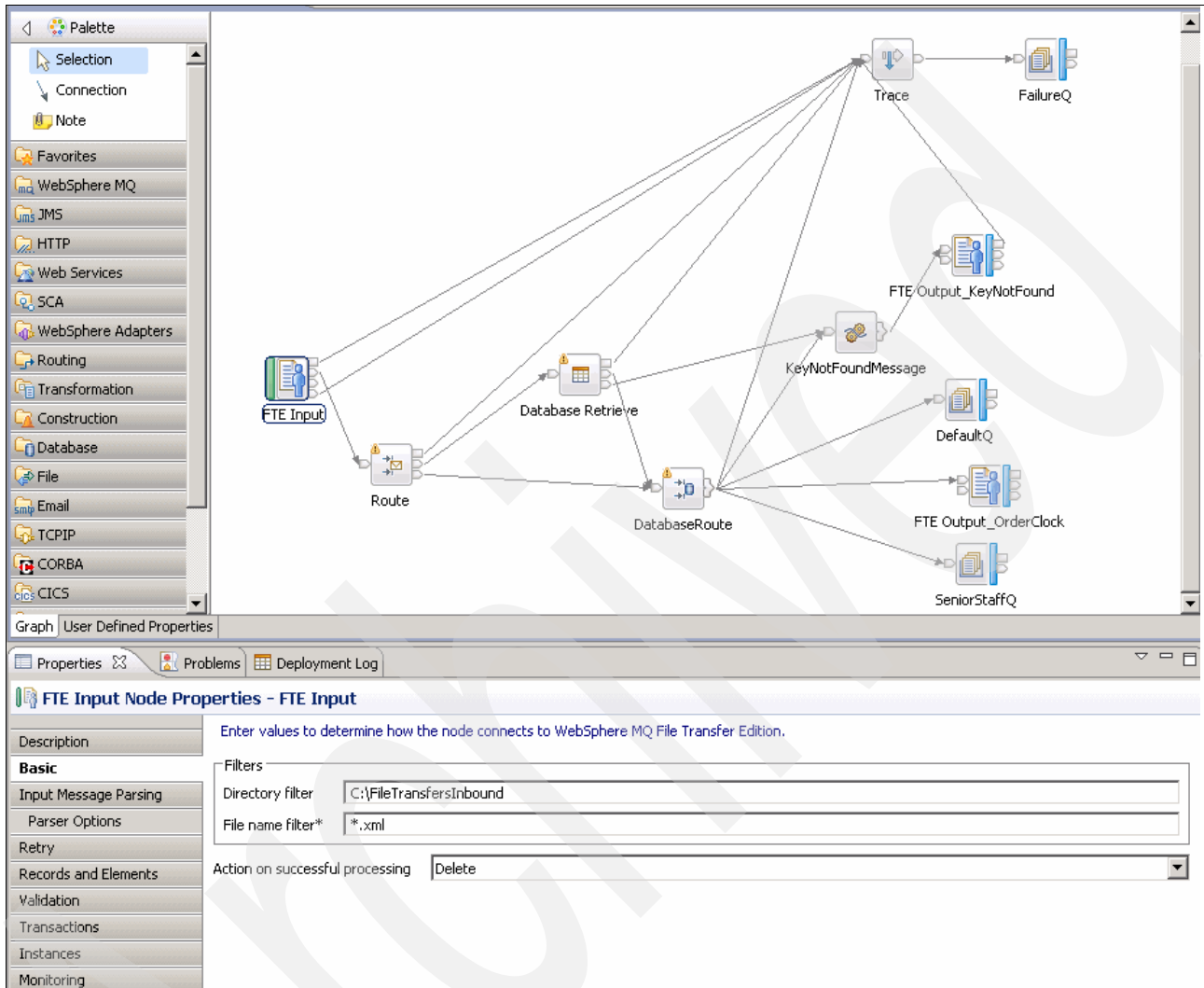


Figure 7-16 Inbound message flow with FTEInput node

The following steps were performed to build the message flow with an FTEInput node for this scenario. Figure 7-16 shows the complete flow. Detailed information about configuring the FTEInput node is given on the property pages for the node in the WebSphere Message Broker Toolkit.

1. Drag an FTEInput node onto a message flow and wire its out terminal to an output node of your choice. Notice that in our flow there is a trace node after the FTEInput node so that the inbound file can be examined.

To process the file based on details of the transfer, place a node such as the route node after the FTEInput node.

Details of the transfer are stored in the local environment, at LocalEnvironment.FTE.

2. Configure the properties of the FTEInput node in the Basic page (shown in the lower half of Figure 7-16). The Basic tab indicates the directory that contains the files to be processed by the node. To process only a defined subset of files sent to an agent,

configure the file name filter. This action allows multiple FTEInput nodes in the same execution group to receive specific files, depending on the directory or file filters that are specified.

When receiving files, you can apply filters. If an execution group has more than one FTEInput node, each node receives only the appropriate files. You can also determine what happens after the file is processed (the file is left in its existing destination directory, left with a time stamp added, or deleted). See the Basic tab on the node for details.

Note that the FTEInput node does not use a transit directory like the FileInput node. Each execution group has its own WebSphere MQ File Transfer Edition agent, and a node processes only files that are sent to the WebSphere MQ File Transfer Edition agent to which the node is deployed. The execution group ensures that only one node in the execution group processes each file.

You can also specify whether, after processing, the file is left in its directory, renamed, or deleted.

3. To change how the node handles a message flow failure, configure the Retry page.
4. To change how records are identified in the input file, configure the record detection property on the Records and Elements page (Figure 7-17). For example, you might want to specify that a record is fixed length and set the record length. Notice that we process the message as a whole file. Use the pull-down menu to select one of the other options:
 - Fixed length
 - Delimited
 - Parsed record sequence

If you set the record detection property to anything other than Whole File, drag an additional output node to the flow, such as the MQOutput node. Wire the End of data terminal on the FTEInput node to the in terminal of the MQOutput node. The node connected to the end of data terminal receives an empty message when the last record in the file is read.

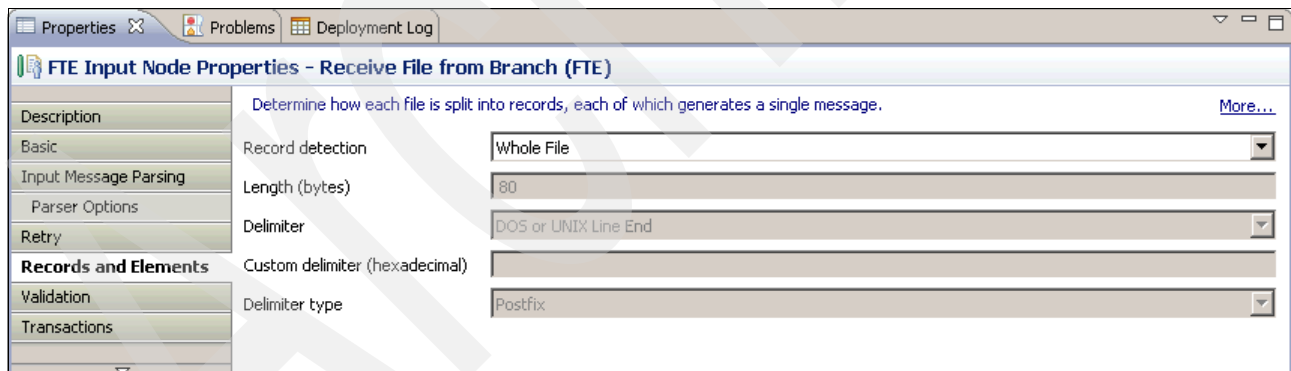


Figure 7-17 FTEInput node properties page for records and elements - Message is whole file

5. The Input Message Parsing property tells how to parse the data. We use the XMLNSC parser (Figure 7-18).

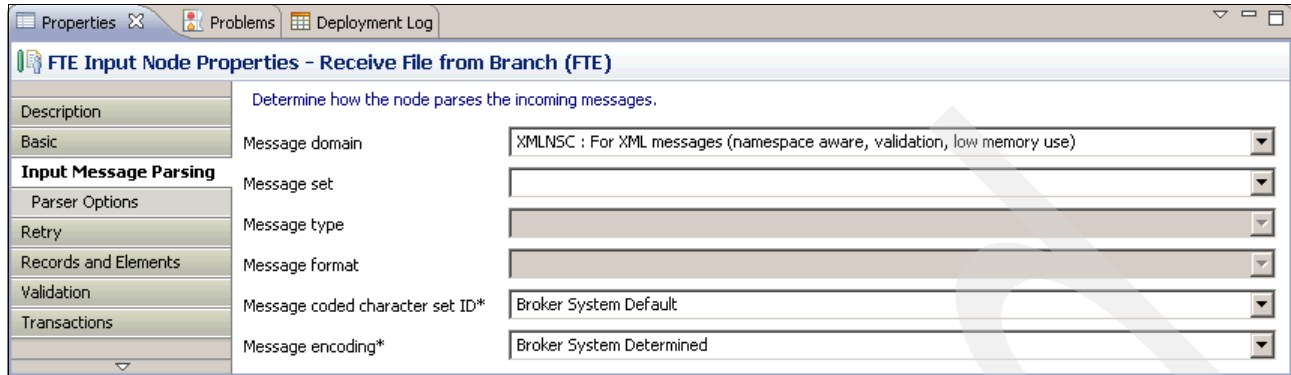


Figure 7-18 FTEInput node properties page for input message parsing - XMLNSC

6. Add the flow to a broker archive (bar) file and deploy the bar file.

Using the FTEOutput node

You can use the FTEOutput node in a message flow that needs to send a file using the MQ FTE backbone network. You can also deploy multiple FTEOutput nodes to the same execution group or to different execution groups in the same broker. However, an FTEOutput node can send only one file per transfer. Each file can have multiple records, and each record can have multiple elements.

Transfers from the FTEOutput node are non-blocking. An error occurs if another transfer is outstanding with the same file name, which suggests that flows with an FTEOutput node are single threaded if you use the same file name for each occurrence of the flow. If you code the properties to create unique file names, you should not encounter transfer errors because of duplicate file names.

When sending a file, you can dynamically set the following properties:

- ▶ Destination agent
- ▶ Destination file directory
- ▶ Destination file name
- ▶ Destination queue manager
- ▶ Job name
- ▶ Overwrite files on destination

Figure 7-19 shows properties for setting the file options. These parameters might be familiar if you have created WebSphere MQ File Transfer Edition transfers. They are equivalent to destination parameters in an `fteCreateTransfer` request:

- ▶ Destination agent name
- ▶ Destination agent queue manager
- ▶ Destination directory
- ▶ Destination file

Also notice the options for mode of transfer and overwriting the destination file.

The screenshot displays the IBM WebSphere MQ Message Designer interface. The top portion shows a message flow diagram for 'InboundFileTransferFlow.msgflow'. The flow starts with an 'FTE Input' node, which connects to a 'Route' node. From the 'Route' node, the flow branches to 'Database Retrieve' and 'DatabaseRoute'. 'Database Retrieve' connects to 'Trace1', which then connects to 'KeyNotFoundMessage'. 'DatabaseRoute' connects to 'FTE Output_KeyNotFound', 'DefaultQ', 'FTE Output_OrderClock', and 'SeniorStaffQ'. The bottom portion of the screenshot shows the 'Properties' pane for the 'FTE Output Node Properties - FTE Output_KeyNotFound'. The 'Basic' tab is selected, showing the following configuration:

- Description:** Settings for working with WebSphere MQ File Transfer Edition.
- Metadata:** Job name: InvalidWorkDepartment
- Destination:**
 - Agent: WMBBRIDGEAGT
 - Queue manager: FTPQMGR
 - File directory: /SysD_WMB_Partner/To_MyFG_Partner
 - File name*: KeyNotFound.txt
- Options:**
 - Mode: Text transfer (ASCII/EBCDIC and CR/LF automated)
 - Disable computation of MD5 check sum:
 - Overwrite files on destination system:

Figure 7-19 Message with FTEOutput node and corresponding properties

You need the following information to configure the message flow with a FTEOutput node:

- ▶ The name of the remote WebSphere MQ File Transfer Edition agent to which the file is to be sent
- ▶ The name of the destination queue manager
- ▶ The name of the output file

Using Figure 7-19 on page 278 as reference, we performed the following steps to build the message with an FTEOutput node for use in the outbound part of the scenario. Detailed information about configuring the FTEOutput node is given on the property pages for the node in the WebSphere Message Broker Toolkit. Create a message flow that contains an input node.

1. Drag an FTEOutput node onto the message flow, and wire its in terminal to the input node.
2. Configure the Basic page (shown in the lower half of Figure 7-19 on page 278):
 - a. Set values for the destination agent and destination file name properties. Configuring just these two properties is enough if you want to send all of the input message tree as a single record in the output file.
 - b. Set a value for the destination queue manager property. The default destination queue manager is the queue manager for the broker.
3. To specify a location in the input message tree for the data to be sent, configure the data location property on the Request properties page.
4. To change how records are placed in the output file, configure the record definition property on the Records and Elements page. For example, you might want to specify that a record is fixed length, and set the record length.

If you set the record definition property to anything other than Record is Whole File, drag a node such as the MQOutput node to the flow and wire its out terminal to the finish file terminal on the FTEOutput node. The node that is connected to the finish file terminal must have logic to determine the last record in the file.

5. If required, configure the node to write the overrides to the LocalEnvironment.Destination.FTE subtree (Table 7-6 on page 283). To set properties for the transfer dynamically, place a node such as the compute node or the mapping node, before the FTEOutput node. You can override the following properties:
 - Destination agent
 - Destination file directory
 - Destination file name
 - Destination queue manager
 - Job name
 - Overwrite files on destination

We did not use any overrides in this scenario.

6. Use a node such as the compute node or the mapping node before the FTEOutput node to alter any of the properties or to add or change headers for the WebSphere MQ File Transfer Edition transfer.

In the flow, notice that we chose to use the compute node to add a line of text to the input message tree before sending to the FTEOutput_KeyNotFound node.

7. Add the flow to a broker archive (bar) file and deploy the bar file.

Local environment tree structure for WebSphere MQ File Transfer Edition nodes

The local environment tree is a part of the logical message tree, in which you can store information while the message flow processes the message. The root of the local environment tree is called *LocalEnvironment*. This tree is always present in the input message. It is created when a message is received by the input node. Certain input nodes create local environment fields. Others leave it empty.

Use the local environment tree to store variables that can be referred to and updated by message processing nodes that occur later in the message flow. You can also use the local environment tree to define destinations (both internal and external to the message flow) to which a message is sent. WebSphere Message Broker also stores information in *LocalEnvironment* in certain circumstances and references it to access values that you might have set for destinations. (Compare this behavior with the environment tree structure, which the broker uses only in specific situations.)

Figure 7-20 shows an example of the local environment tree structure. The children of the destination are protocol-dependent.

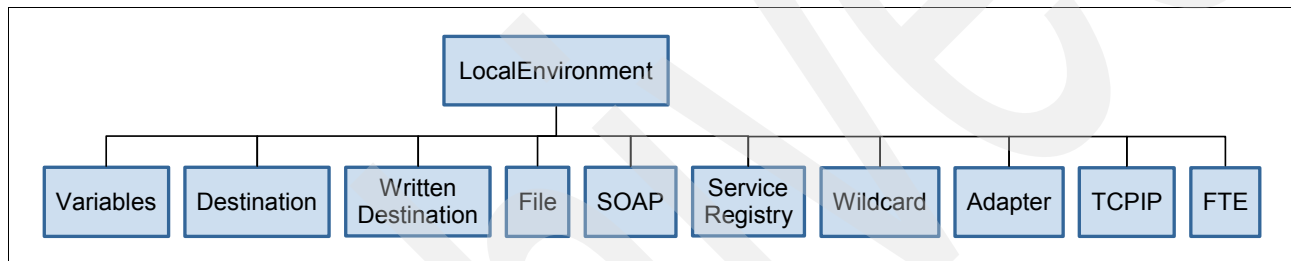


Figure 7-20 *Local.Environment tree structure*

In the tree structure shown in Figure 7-20, the local environment has several children. This subtree is optional. If you create local environment variables, store them in a subtree called *Variables*. This subtree provides a work area that you can use to pass information between nodes. This subtree is never inspected or modified by any supplied node.

Variables in the local environment can be changed by any subsequent message processing node, and the variables persist until the node that created them goes out of scope. The variables in this subtree are persistent only within a single instance of a message flow. If you have multiple instances of a message passing through the message flow and need to pass information between them, you must use an external database.

You can use fields in the local environment to dynamically alter the behavior of the *FTEInput* and *FTEOutput* nodes. You can also determine the values that the output nodes used to process the file.

LocalEnvironment.Wildcard.WildcardMatch tree structure for WebSphere MQ File Transfer Edition nodes

On the FileInput and FTEInput nodes, you can specify a file name pattern that contains wildcard characters. The input nodes copy the characters in the file name that is matched by wildcards, together with any intermediate characters, to LocalEnvironment.Wildcard.WildcardMatch (Table 7-2).

Table 7-2 LocalEnvironment.Wildcard.WildcardMatch field

Element name	Element data type	Description
Wildcard match	CHARACTER	You can use fields in the local environment to dynamically alter the behavior of the FileInput, FileOutput, FTEInput, and FTEOutput nodes. You can also find which values the output nodes used to process the file.

LocalEnvironment.FTE tree structure for WebSphere MQ File Transfer Edition nodes

When you use the FTEInput node, it stores information that you can access in the LocalEnvironment.FTE and LocalEnvironment.FTE.Transfer message trees. The LocalEnvironment.FTE subtree stores information relating to the current record and is populated by the broker. Table 7-3 describes the fields in this structure.

Table 7-3 LocalEnvironment.FTE fields

Element name	Element data type	Description
TimeStamp	CHARACTER	You can use fields in the local environment to dynamically alter the behavior of the FileInput, FileOutput, FTEInput, and FTEOutput nodes. You can also determine the values that the output nodes used to process the file.
Offset	INTEGER	Start of the record within the file. The first record starts at offset 0 bytes. When offset is part of the end of data message tree, this value is the length of the input file.
Record	INTEGER	Number of the record within the file. The first record is record number 1. When the record is part of the end of data message tree, this value is the number of records.
Delimiter	CHARACTER	The characters used to separate this record from the preceding record, if delimited is specified in record detection. The first record has a null delimiter. When delimiter is part of the end of data message tree, this value is the delimiter that follows the last record, if any.
IsEmpty	BOOLEAN	Whether the record propagated by the message flow is empty. IsEmpty is set to TRUE if the current record is empty. When IsEmpty is part of the end of data message tree, this value is always set to TRUE.

Local Environment.FTE.Transfer tree structure for WebSphere MQ File Transfer Edition nodes

The LocalEnvironment.FTE.Transfer subtree contains information received from WebSphere MQ File Transfer Edition regarding the transfer or file. Table 7-4 describes the fields in this structure.

Table 7-4 LocalEnvironment.FTE.Transfer message tree

Element name	Element data type	Description
Directory	CHARACTER	The absolute directory path of the input directory.
JobName	CHARACTER	The name of the transfer.
Name	CHARACTER	The file name and extension (per file).
LastModified	TimeStamp	Date and time the file was last modified (per file).
SourceAgent	CHARACTER	The name of the agent sending the file.
DestinationAgent	CHARACTER	The name of the agent to which to send the file.
OriginatingHost	CHARACTER	The name of the host from which the transfer was submitted.
TransferId	CHARACTER	The unique name of the transfer.
MQMDUser	CHARACTER	The MQ user ID in the MQMD of the transfer message.
OriginatingUser	CHARACTER	The user ID of the user that submitted the transfer request.
TransferMode	CHARACTER	The mode of the transfer. Valid values are binary and text.
TransferStatus	CHARACTER	The status of the transfer of the file.
FileSize	INTEGER	The size of the file being transferred.
ChecksumMethod	CHARACTER	The only allowed value is MD5.
Checksum	CHARACTER	If the ChecksumMethod element is set to MD5, this element is the actual checksum in hex string format.
DestinationAgentQmgr	CHARACTER	The name of the destination agent's queue manager to which to send the file.
SourceAgentQmgr	CHARACTER	The name of the source agent's queue manager that sent the file.
OverallTransferStatus	CHARACTER	The overall status of the transfer.
TotalTransfers	INTEGER	The total number of files successfully transferred.
TransferNumber	INTEGER	The number of the current file in the transfer.

The LocalEnvironment.FTE and LocalEnvironment.FTE.Transfer structures are propagated with each message written to the out terminal of the FTEInput node and with the empty message written to the end of data terminal.

Local Environment.WrittenDestination tree structure for WebSphere MQ File Transfer Edition nodes

When you use the FTEOutput node, it stores information that you can access in the LocalEnvironment.WrittenDestination.FTE message tree. Table 7-5 describes the fields in this structure.

Table 7-5 LocalEnvironment.WrittenDestination.FTE fields

Element name	Element data type	Description
DestinationAgent	CHARACTER	The name of the agent to which to send the file.
DestinationAgentQmgr	CHARACTER	The name of the destination queue manager.
JobName	CHARACTER	The name for the transfer.
Directory	CHARACTER	The absolute directory path of the output directory in the form used by the file system of the broker. For example, on Windows systems, this starts with the drive letter prefix (such as C:).
Name	CHARACTER	The file name of the output file.
Overwrite	BOOLEAN	The absolute directory path of the output directory in the form used by the file system of the broker. For example, on Windows systems, this starts with the drive letter prefix (such as C:).

Local Environment.Destination.FTE tree structure for WebSphere MQ File Transfer Edition nodes

This subtree consists of a number of children that indicate the transport types to which the message is directed (the transport identifiers) or the target label nodes that are used by a RouteToLabel node. Transport information is used by the following input and output nodes:

- ▶ WebSphere MQ File Transfer Edition (FTE)
- ▶ HTTP
- ▶ WebSphere MQ (MQ)
- ▶ Java Message Service (JMS)
- ▶ SOAP
- ▶ File
- ▶ Email
- ▶ TCP/IP

When you use the FTEOutput node, you can override its destination agent, destination queue manager, job name, destination file directory, destination file name, and overwrite files on destination system properties with elements in the message tree. The default location for these overrides is LocalEnvironment.Destination.FTE. Table 7-6 describes the fields of this structure.

Table 7-6 LocalEnvironment.Destination.FTE fields

Element name	Element data type	Description
DestinationAgent	CHARACTER	The name of the agent to which to send the file.
DestinationAgentQmgr	CHARACTER	The name of the destination queue manager.
Jobname	CHARACTER	The name of the transfer.

Element name	Element data type	Description
Directory	CHARACTER	The absolute directory path of the output directory in the form used by the file system of the broker. For example, on a Windows system, this starts with the drive letter prefix (such as C:).
Name	CHARACTER	The file name of the output file.
Overwrite	BOOLEAN	This specifies whether files on the destination system can be overwritten when the destination agent moves files of the same name there. If the destination agent fails to overwrite the file, the transfer fails and the transfer logs report the failure. The FTEOutput node does not throw or log any errors.
TransferId	CHARACTER	The unique name of the transfer initiated by the FTEOutput node.

7.3.7 Creating a community in Sterling File Gateway

This section assumes that you understand how to start Sterling File Gateway and understand the basic administration and operation of Sterling File Gateway. To create a community in Sterling File Gateway, you need access to the Sterling File Gateway user interface.

Logging in to Sterling File Gateway

To log in to Sterling File Gateway:

1. Direct a web browser to the following URL to access the Sterling File Gateway Administration Console:

`http://<servername>:<port>/filegateway/`

Where `<servername>` is the name of your server and `<port>` is the port that Sterling File Gateway uses.

2. Log in using `<userid>/<password>` (Figure 7-21). The default admin user ID for Sterling File Gateway is `fg_sysadmin`.



Figure 7-21 Log in page for Sterling File Gateway

The main window for Sterling File Gateway opens (Figure 7-22).

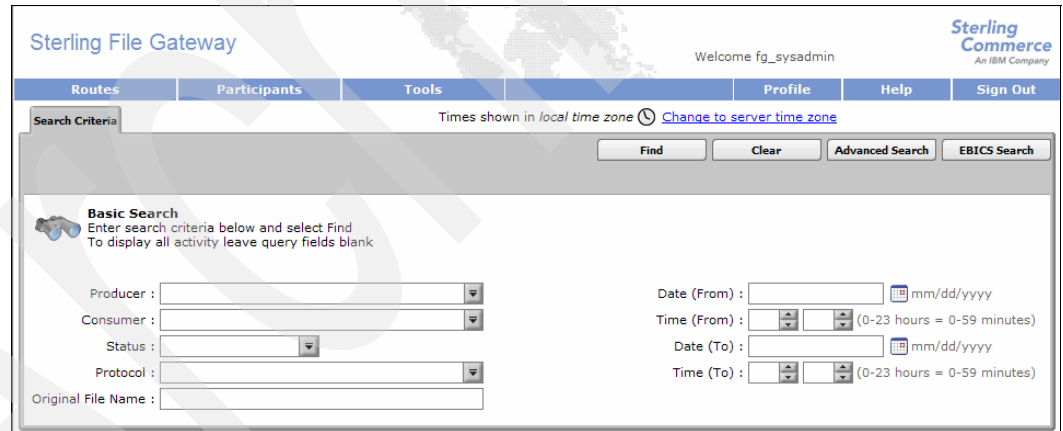


Figure 7-22 Sterling File Gateway Administration Console opening page

Creating a community

A *community* defines the protocols that partners within this community can use. If you have not already created a community according to the instructions in “Creating a community” on page 131, complete the following steps in Sterling File Gateway:

1. Navigate to the top menu bar option in the Sterling File Gateway Administration console and select **Participants** → **Communities** (Figure 7-23).

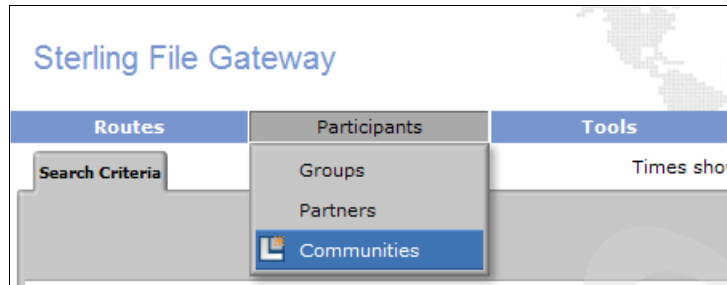


Figure 7-23 Participants menu options

2. Select **add** in the Communities window (Figure 7-24).



Figure 7-24 Add communities

3. Enter a name for your community (Figure 7-25). We named our community FirstCommunity. Click **Next**.

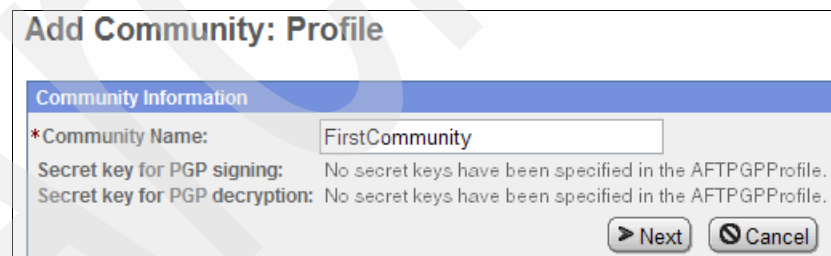


Figure 7-25 Name the community

The community for this book: For simplicity, we create only one community for this book and enable all protocols within that community.

4. Select all the options to allow all partners in this community to communicate using all available protocols (FTP or FTPS, Sterling Connect:Direct or SSH/SFTP) (Figure 7-26). Click **Next**.




Figure 7-26 Select all protocols

5. On the Notifications window (Figure 7-27), accept the default setting of **No**. Click **Next**.

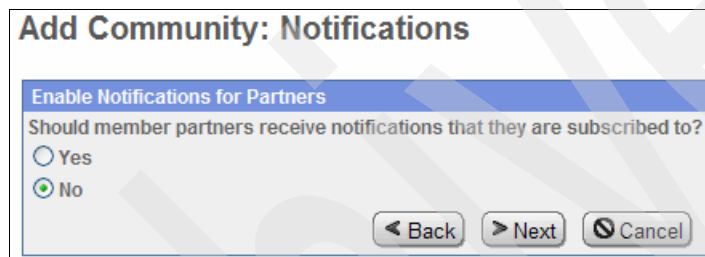


Figure 7-27 Accept the default setting for notifications

6. Review the selections on the confirmation window (Figure 7-28). Click **Finish**. Close the browser window to go back to the main Sterling File Gateway Administration Console page.



Figure 7-28 Add community confirmation page

7.3.8 Creating routing channel templates in Sterling File Gateway

This scenario uses two channel templates. If you do not already have the channel templates set up, the configuration is detailed in the following locations:

1. Follow the steps to create the PassThrough_RouteByMailbox template described in “Creating the PassThrough_RouteByMailbox routing channel template” on page 139.
2. Follow the steps to create the second template, PassThrough, as described in “Creating the PassThrough routing channel template” on page 159.

7.3.9 Enabling WebSphere MQ File Transfer Edition in Sterling File Gateway

In 6.4.1, “Inbound scenario” on page 229, we describe how to enable WebSphere MQ File Transfer Edition as an available transfer protocol in Sterling File Gateway. Refer to this section and follow the instructions given if you have not enabled WebSphere MQ File Transfer Edition in Sterling File Gateway.

7.3.10 Modifying FirstCommunity in Sterling File Gateway

The steps for adding the WebSphere MQ File Transfer Edition protocol to a Sterling File Gateway community are included in Chapter 6, “External Transfers with Protocol Switching between IBM Sterling Connect:Direct and WebSphere MQ File Transfer Edition via Sterling File Gateway” on page 167. If you have not modified the community as shown in “Creating the FirstCommunity community in Sterling File Gateway” on page 212, then follow the steps to add WebSphere MQ File Transfer Edition as an available protocol in the FirstCommunity community.

7.3.11 Creating a listener queue for Sterling B2B Integrator

Through its WebSphere MQ adapter, Sterling B2B Integrator listens to a reply queue to receive replies back from WebSphere MQ File Transfer Edition. These replies contain information regarding the status of the transfer and the transfer command Sterling B2B Integrator placed on the WMBBRIDGEAGT's command queue. To capture these responses, a local queue must be created on the WebSphere MQ queue manager local to WMBBRIDGEAGT and Sterling File Gateway, FTPQMGR.

You can find instructions for creating a local queue at:

http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp?topic=/com.ibm.mq.explorer.tutorials.doc/bi00257_.htm

We named our local queue *REPLYMSGQ*. You need this queue to complete the next section.

7.3.12 Setting up a trading partner for WebSphere Message Broker

For Sterling File Gateway to send a file to WebSphere Message Broker, you create a trading partner in Sterling File Gateway to send files to WebSphere Message Broker on SysD using the WebSphere MQ File Transfer Edition protocol. To create a trading partner named SysD_WMB_Partner:

1. The trading partner is created in the Sterling File Gateway Administration Console. From the navigational menu bar on the main page of the console, select **Participants** → **Partners** (Figure 7-29).



Figure 7-29 Participants menu options, Partners

Sterling File Gateway Administration Console: We include directions about how to access the console in “Logging in to Sterling File Gateway” on page 284.

2. Under the Partners tab, click **Create** to add a new trading partner to Sterling File Gateway (Figure 7-30).

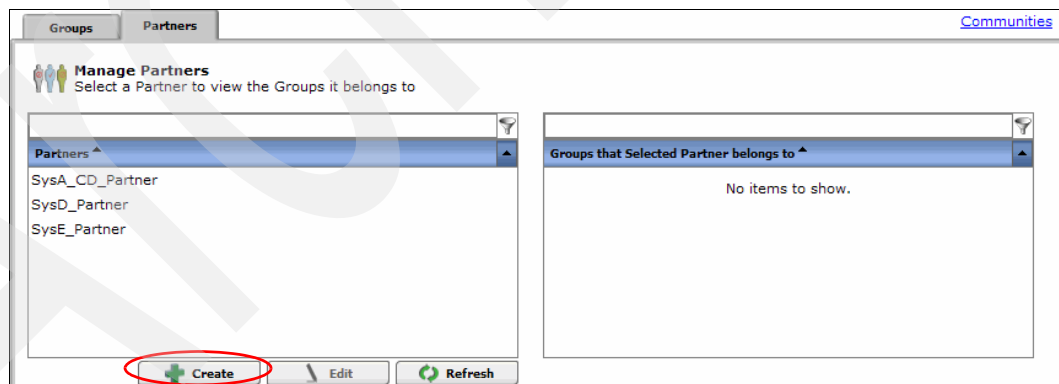


Figure 7-30 Create new partner

3. Select the community that you created (Figure 7-31). In our scenario, the community is **FirstCommunity**. Click **Next**.




Figure 7-31 Select the community for the new partner to join

4. Create the WebSphere Message Broker business partner. This partner sends files from Sterling File Gateway over WebSphere MQ File Transfer Edition. Enter a name for the partner. We chose to use the name SysD_WMB_Partner.

The phone number and email address fields are mandatory even though we have notifications disabled in this example. We used a false telephone number and email address for this example (Figure 7-32).

Click **Next**.



Figure 7-32 Enter contact details for the SysD_WMB_Partner internal trading partner

5. Enter a user name and password for the SysD_WMB_Partner internal trading partner. We used SysD_WMB_Partner as the user name (Figure 7-33). Click **Next**.

Add Partner: User Account

User Account

Create a user account by filling the following information. This user account is to be used by SysD_WMB_Partner when logging into myFileGateway.

*User Name: SysD_WMB_Partner

*Password: ●●●●●●

*Confirm Password: ●●●●●●

*First Name: SysD_WMB_Firstname

*Last Name: SysD_WMB_Lastname

< Back > Next Cancel

Figure 7-33 Create a user account for the SysD_WMB_Partner internal trading partner

6. In our scenario, the created partner, SysD_WMB_Partner, puts files into a mailbox on Sterling File Gateway (Producer of Data) and retrieves files residing in its mailbox (Consumer of Data). In other words, this partner is used for both inbound and outbound file transfers.

From the Select Partner Role page (Figure 7-34), configure a partner that is used for both inbound and outbound file transfers:

- a. Select that the partner be both a **Consumer of Data** and a **Producer of Data**.
- b. Under the Consumer of Data section, select the **Partner will listen for a connection to consume data** option. This means that SysD_WMB_Partner will host a server that listens for connections that can be used to transfer files. Click **Next**.

Add Partner: Partner Role

Select Partner Role

There are no consumers configured in the system right now. It is recommended that the first partner that is configured is one that is a consumer or both a consumer and a producer.

SysD_WMB_Partner is a Consumer of Data

Partner will initiate the connection to consume data.

Partner will listen for a connection to consume data.

SysD_WMB_Partner is a Producer of Data

< Back > Next Cancel

Figure 7-34 Select Partner Role page

- This partner communicates only using WebSphere MQ File Transfer Edition and not with SSH/SFTP or SSH/SCP protocols. On the Initiate Connection Settings page (Figure 7-35), the default selection is No. Leave the default selected. Click **Next**.

Figure 7-35 Initiate Connections Settings page

- Choose how the partner, SysD_WMB_Partner, communicates by selecting **Listen for WebSphere MQ FTE Connections** (Figure 7-36). Select **Next**.

Figure 7-36 Select a Protocol Published by the community

- On the WebSphere MQ FTE page (Figure 7-37 on page 293), configure the Sterling File Gateway partner to communicate with a WebSphere MQ File Transfer Edition agent. Enter the details as shown in Table 7-7. The values shown in Table 7-7 are the values that we created for this scenario. After you enter the values, click **Next**.

Table 7-7 WebSphere MQ File Transfer Edition protocol values for SysD_WMB_Partner

Parameter	Value
Source Agent Name (-sa)	WMBBRIDGEAGT
Source Agent Queue Manager (-sm)	FTPQMGR
Destination Agent Name (-da)	BRKR.AGT
Destination Agent Queue Manager (-dm)	BRKRQMGR
Destination Agent's Directory (-dd)	c:\FileTransfersInbound\
Queue for Transfer Status Reply Messages	REPLYMSGQ
Destination File Already Exists (-de)	overwrite
Transfer Timeout (seconds)	600

WebSphere MQ FTE

WebSphere MQ FTE Parameters	
* Source Agent Name (-sa)	WMBBRIDGEAGT
* Source Agent Queue Manager (-sm)	FTPQMGR
Source Agent Queue Manager Host Name	
Source Agent Queue Manager Port	
Source Agent Queue Manager User Id	
Source Agent Queue Manager Password	
Source Agent Queue Manager Password Confirm	
* Destination Agent Name (-da)	BRKR.AGT
* Destination Agent Queue Manager (-dm)	BRKRQMGR
* Destination Agent's Directory (-dd)	c:\FileTransfersInbound\
* Destination File Already Exists (-de)	overwrite
Queue For Transfer Status Reply Messages	REPLYMSGQ
* Priority (-pr)	0
* Conversion (-t)	binary
* Checksum Method (-cs)	MD5
Transfer Timeout (seconds)	600

Figure 7-37 Enter the WebSphere MQ File Transfer Edition details for SysD_WMB_Partner

Reply queue: A queue is defined for the transfer status reply messages. Although this field is optional, we suggest using a queue for transfer status messages.

If no queue is specified here, the business process that initiates the WebSphere MQ File Transfer Edition cannot monitor the status of the file transfer. Provided that the business process can successfully write the transfer request message to the command queue, the business process reports a successful file transfer even when the file transfer fails because the business process has completed because no method is specified for Sterling B2B Integrator to monitor the status.

If a queue is specified, WebSphere MQ File Transfer Edition places status messages on that queue. The business process listens on that queue and can more accurately report transfer successes or failures.

10. On the PGP Settings page, accept the default values of No (Figure 7-38). Click **Next**.

PGP Settings

PGP Settings

When SysD_WMB_Partner sends PGP packaged files
... the files are processed in accordance with the Routing Channels (and their templates) that SysD_WMB_Partner is a producer for.

Note:

- *If SysD_WMB_Partner sends data that is encrypted, it will be decrypted using the Router's secret PGP key.
- *If SysD_WMB_Partner sends data that is signed, it will be verified using SysD_WMB_Partner's public PGP key if that key is present in the public key ring.

Does SysD_WMB_Partner require data to be signed by the Router ?

Yes No

Does SysD_WMB_Partner require data to be encrypted by the Router ?

Yes No

< Back > Next Cancel

Figure 7-38 PGP Settings page

11. Review your choices on the confirmation page. Click **Finish** to complete the creation of your partner, SysD_WMB_Partner (Figure 7-39).

Add Trading Partner: Confirm

Profile	
Partner Name:	SysD_WMB_Partner
Address:	*
Phone:	3333
Email Address:	sysdwmb@itso_redbooks.com
User Account	
User Name:	SysD_WMB_Partner
Password:	*****
First Name:	SysD_WMB_Firstname
Last Name:	SysD_WMB_Lastname
Protocol	
Partner Role:	Producer of Data, Consumer of Data
Connection Direction:	Listen Connection
Transport Method:	WebSphere MQ FTE
<ul style="list-style-type: none"> • Source Agent Name (-sa) • Source Agent Queue Manager (-sm) • Source Agent Queue Manager Host Name • Source Agent Queue Manager Port • Source Agent Queue Manager User Id • Source Agent Queue Manager Password • Destination Agent Name (-da) • Destination Agent Queue Manager (-dm) • Destination Agent's Directory (-dd) • Destination File Already Exists (-de) • Queue For Transfer Status Reply Messages • Priority (-pr) • Conversion (-t) • Checksum Method (-cs) • Transfer Timeout (seconds) 	WMBBRIDGEAGT FTPQMGR ***** BRKR.AGT BRKRQMGR c:\FileTransfersInbound\ overwrite REPLYMSGQ 0 binary MD5 600
Does SysD_WMB_Partner require data to be signed by the Router :	No
Does SysD_WMB_Partner require data to be encrypted by the Router :	No
Community Membership	
Community Name:	FirstCommunity
Joined Date:	Today

Figure 7-39 Confirmation page

12. When the partner is added successfully, close the browser window and return to the main Sterling File Gateway Administration Console page.

7.3.13 Creating a trading partner for myFileGateway

This section describes how to create a business partner to represent an organization that sends files into the internal network of another company using myFileGateway:

1. From the Sterling File Gateway Administration Console main page (Figure 7-40), use the top navigational menu to select **Participants** → **Partners**.



Figure 7-40 Participants Partners menu options

2. Click **Create** under the Partners tab (Figure 7-41) to add a new partner to Sterling File Gateway.

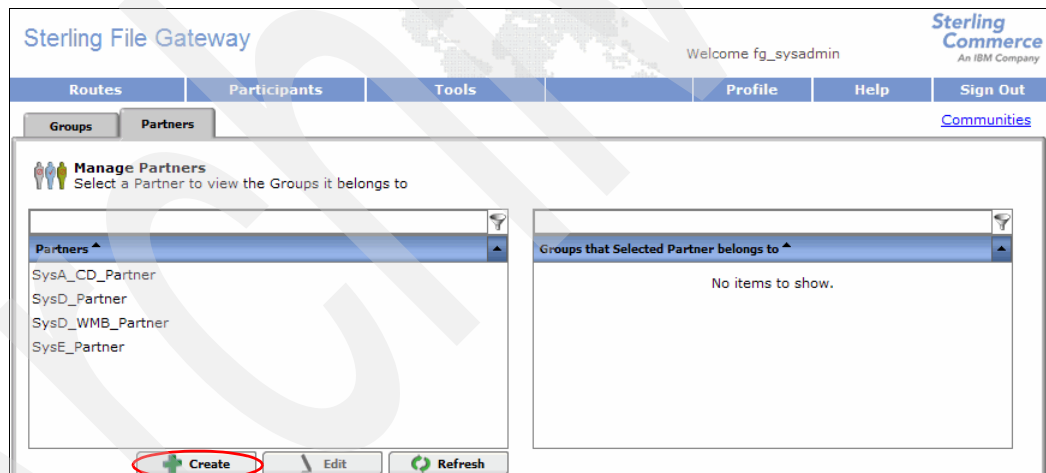


Figure 7-41 Create a new partner

3. Select your community (Figure 7-42). Our community is FirstCommunity. Click **Next**.

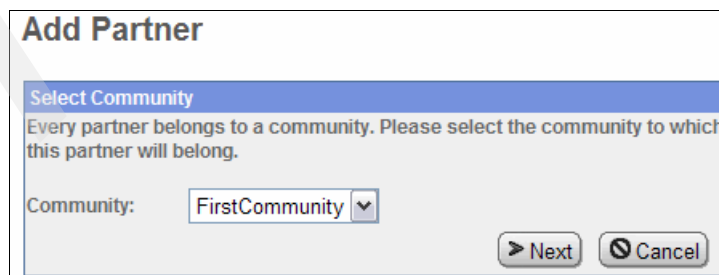


Figure 7-42 Select the community for the new partner to join

4. Create the myFileGateway business partner. This partner uploads and downloads files through myFileGateway. Enter a name for the partner. We chose to use the name of MyFG_Partner.

The phone number and email address fields are mandatory even though we have notifications disabled in this example. We used a false telephone number and email address for this example (Figure 7-43).

Click **Next**.

Add Partner: Information

Contact Information

* Partner Name: MyFG_Partner

Address:

City:

State:

Postal Code:

* Phone: 8888

Country: UNITED STATES

Time Zone: (GMT-05:00) Eastern Time (US & Canada)

* Email Address: myfg@itso_redbooks.com

< Back > Next Cancel

Figure 7-43 Enter contact information for the MyFG_Partner external business partner

5. Create a user name and password for the MyFG_Partner external trading partner (Figure 7-44). These values are used to log in to myFileGateway. This credential allows an external trading partner to place files into or retrieve files from the MyFG_Partner mailboxes. In our scenario, we used MyFG_Partner as the user name. After you enter the user name and password, click **Next**.

Add Partner: User Account

User Account

Create a user account by filling the following information. This user account is to be used by MyFG_Partner when logging into myFileGateway.

* User Name: MyFG_Partner

* Password: ●●●●●●

* Confirm Password: ●●●●●●

* First Name: MyFG_Firstname

* Last Name: MyFG_Lastname

< Back > Next Cancel

Figure 7-44 Enter user account details for the MyFG_Partner external business partner

- In our scenario, the created partner, MyFG_Partner, puts files into a mailbox (Producer of Data) and retrieves files residing in its mailbox (Consumer of Data). That is, this partner is used for both inbound and outbound file transfers.

From the Select Partner Role page (Figure 7-45), configure a partner that is used for both inbound and outbound file transfers:

- Select that the partner be both a **Consumer of Data** and a **Producer of Data**.
- Under the Consumer of Data section, select the **Partner will initiate the connection to consume data** option. This means that MyFG_Partner will connect to Sterling File Gateway to consume files placed into the mailbox. Click **Next**.

Add Partner: Partner Role

Select Partner Role

There are no consumers configured in the system right now. It is recommended that the first partner that is configured is one that is a consumer or both a consumer and a producer.

MyFG_Partner is a Consumer of Data

Partner will initiate the connection to consume data.

Partner will listen for a connection to consume data.

MyFG_Partner is a Producer of Data

< Back > Next Cancel

Figure 7-45 Partner Role page

- This partner communicates only using myFileGateway and not with the SSH/SFTP or SSH/SCP protocols. On the Initiate Connection Settings page (Figure 7-46), the default selection is No. Leave the default selected. Click **Next**.

Add Partner: Initiate Connections Settings

Initiate Connections Settings

Will MyFG_Partner use either SSH/SFTP or SSH/SCP protocol to initiate connections?

Yes No

< Back > Next Cancel

Figure 7-46 Initiate Connections Settings

- On the PGP Settings page, accept the default values of No (Figure 7-47). Click **Next**.

PGP Settings

PGP Settings

When MyFG_Partner sends PGP packaged files
 ... the files are processed in accordance with the Routing Channels (and their templates) that MyFG_Partner is a producer for.
 Note:
 *If MyFG_Partner sends data that is encrypted, it will be decrypted using the Router's secret PGP key.
 *If MyFG_Partner sends data that is signed, it will be verified using MyFG_Partner's public PGP key if that key is present in the public key ring.

Does MyFG_Partner require data to be signed by the Router ?
 Yes No

Does MyFG_Partner require data to be encrypted by the Router ?
 Yes No

< Back Next > Cancel

Figure 7-47 PGP Settings

- Review your choices on the confirmation page. Click **Finish** to complete the creation of your partner, MyFG_Partner (Figure 7-48).

Add Trading Partner: Confirm

Profile	
Partner Name:	MyFG_Partner
Address:	.
Phone:	8888
Email Address:	myfg@itso_redbooks.com
User Account	
User Name:	MyFG_Partner
Password:	*****
First Name:	MyFG_Firstname
Last Name:	MyFG_Lastname
Protocol	
Partner Role:	Producer of Data, Consumer of Data
Connection Direction:	Initiate Connection
Will MyFG_Partner use either SSH/SFTP or SSH/SCP protocol to initiate connections?	No
Transport Method:	MAILBOX
Does MyFG_Partner require data to be signed by the Router :	No
Does MyFG_Partner require data to be encrypted by the Router :	No
Community Membership	
Community Name:	FirstCommunity
Joined Date:	Today

< Back Cancel **Finish**

Figure 7-48 Confirmation page

- When the partner is added successfully, close the window and return to the main Sterling File Gateway Administration Console page.

7.3.14 Configuring Sterling B2B Integrator for SFTP communication

This section describes how to configure an SFTP adapter in Sterling B2B Integrator. This SFTP adapter allows Sterling B2B Integrator to act as an SFTP server for the external trading partner to make a client connection for file transmission. To configure the adapter and

associated components, you need access to the Sterling B2B Integrator Administration Console. This section assumes that you understand how to start Sterling B2B Integrator and understand the basic administration and operation of Sterling B2B Integrator.

Logging in to Sterling B2B Integrator

To log in to Sterling B2B Integrator:

1. Start Internet Explorer and go to:

`http://<servername>:<port>/filegateway/`

Where *<servername>* is the name of your server and *<port>* is the port that Sterling B2B Integrator uses.

2. Log in using the Sterling File Gateway administrator user ID and password (Figure 7-49). The default administrator user ID is `fg_sysadmin`.



Figure 7-49 Sterling File Gateway login page

3. In Sterling File Gateway, go to **Tools** → **B2B Console** (Figure 7-50) to open the Sterling B2B Integrator console in a new browser window.



Figure 7-50 Open the Sterling B2B Integrator browser window

The Sterling B2B Integrator console displays in a new browser similar to Figure 7-51.

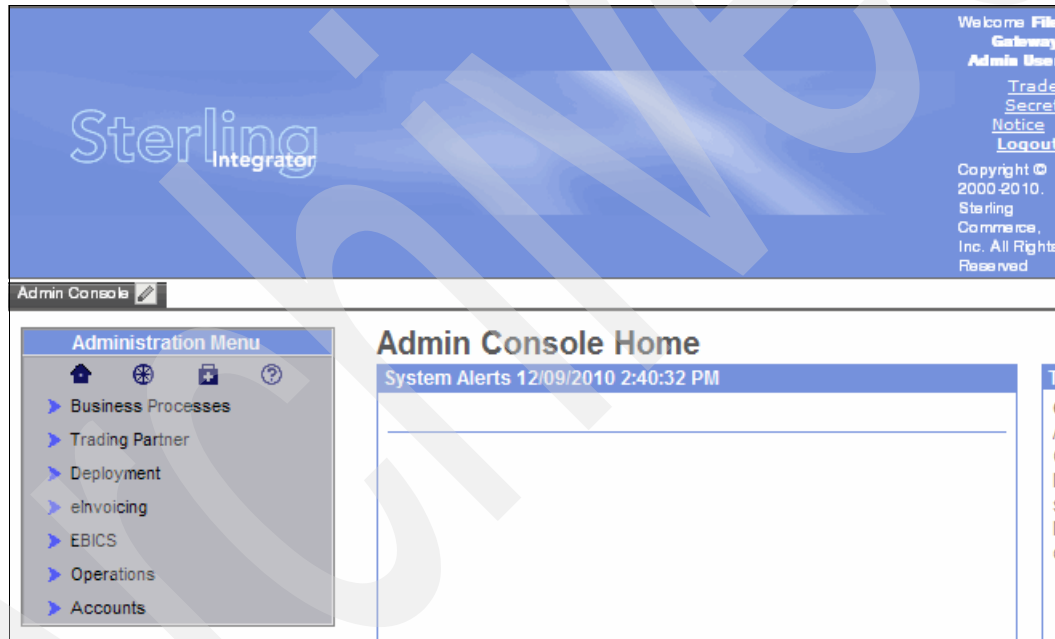


Figure 7-51 Sterling B2B Integrator Admin Console Home page

Generating a new SSH Host Identity Key

SSH provides protection against password sniffing and third-party session monitoring to better protect an organization's privacy during data transmissions. SSH offers the connecting client session a variety of options for authentication. The type of authentication that is used for the session is determined by the SSH server. The client then attempts to authenticate using the determined authentication type.

During authentication, the SSH server has a unique identifying code called the *host identity key*. This key prevents another server from being forged by another server. This key is designed to keep the transmission session from being attacked and from directing the external trading partner client to a foreign, unknown server. The client can decide whether to trust the host identity key.

To generate a new SSH Host Identity Key in Sterling B2B Integrator:

1. From the Sterling B2B Integrator Administration Console's Administration Menu, select **Deployment** → **SSH Host Identity Key**. After you select the SSH Host Identity Key, the right pane looks similar to Figure 7-52. Select **Go!** next to New Host Identity Key.

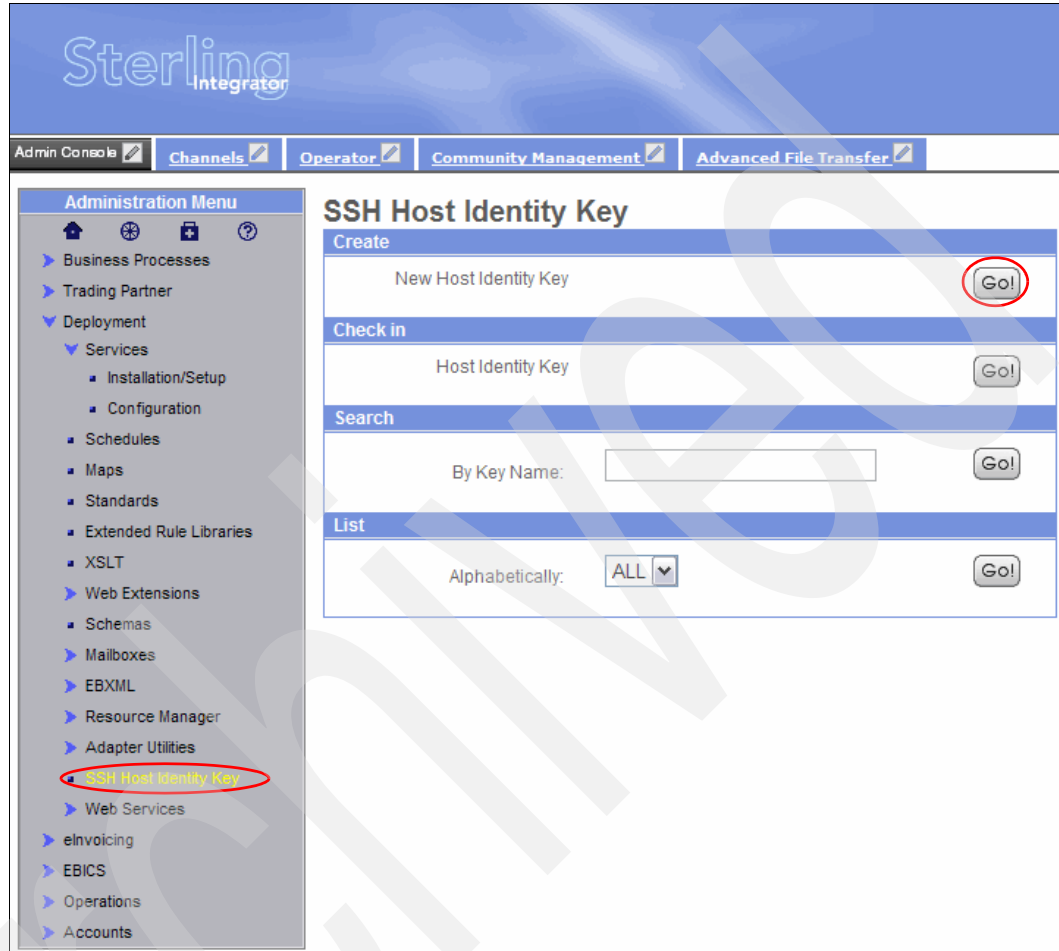


Figure 7-52 Create new host identity key

- From the SSH Host Identity Key page, enter a host name, select the key type, select the key length, enter a comment describing the key, and verify that the box next to Enabled is checked (Figure 7-53). For our scenario, we used a host name of sysc and added the comment of SFTP Server Adapter key. Leave the defaults for key type and key length. Click **Next**.

SSH Host Identity Key

New Host Identity Key

Host Name:

Key Type:

Key Length:

Key Comment:

Enabled

Figure 7-53 Enter values for the SSH host identity key

- Review the confirmation page for the SSH Host Identity Key and click **Finish** (Figure 7-54).

SSH Host Identity Key

Confirm

Key Specification	
Host Name	sysc
Key Type	ssh-rsa
Key Length	1024
Key Comment	SFTP Server Adapter key
Host Identity Key Status	Enabled

Figure 7-54 Confirmation page

Configuring the SFTP Server Adapter

A disabled SFTP Server Adapter is installed by default into Sterling B2B Integrator. The SFTP Server Adapter requires configuration before it can be started. To configure the adapter:

1. From the Sterling B2B Integrator Administration Console under the Administration Menu, select **Deployment** → **Services** → **Configuration**. From the Services Configuration window (Figure 7-55), on the right side of the browser, search for a service by entering SFTP Server Adapter in the search box.

Click **Go!** to the right of the search box to start the search.

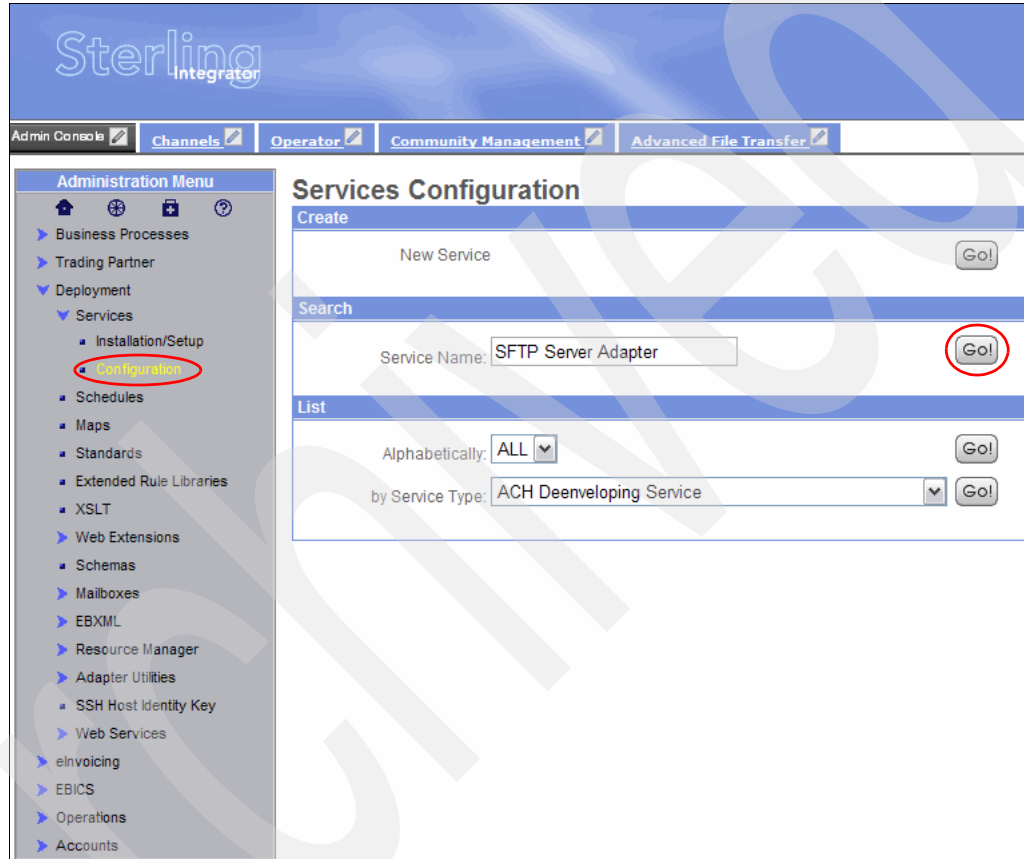


Figure 7-55 Search for the SFTP Server Adapter service

2. Figure 7-56 shows the search results that display. The service shown is the default SFTP Adapter. Click **edit**.



Figure 7-56 Services Configuration search results

3. The first page available for editing is the SFTP Server Adapter:Name page. Keep the defaults shown in Figure 7-57. Click **Next**.

Services Configuration

SFTP Server Adapter: Name

Name: SFTP Server Adapter

Description: SFTP Server Adapter

Select a group:

None

Create New Group

Select Group:

< Back > Next Cancel Save

Figure 7-57 SFTP Server Adapter:Name configuration

4. The SFTP Server Adapter: Configuration page displays all of the properties that the SFTP Server Adapter uses (Figure 7-58). For our scenario, we modified the Enabled Protocols and Host Identity Key as follows:
 - a. Select **SFTP** as the Enabled Protocols.
 - b. Select **sysc** as the Host Identity Key. (You created the sysc host identity key in “Generating a new SSH Host Identity Key” on page 301.)
 - c. Accept the default values for all other fields and options on this page. Click **Next**.

Services Configuration

SFTP Server Adapter: Configuration

Perimeter Server: node1 & local

Enabled Protocols: SFTP

Host Identity Key: sysc

SFTP Server Listen Port: 8119

Minimum Number of Threads: 3

Maximum Number of Threads: 6

Transfer Thread Pool Size: 2

Channels Per Transfer Thread: 400

Maximum Authentications: 3

Session Timeout (secs): 120000

Idle Connection Timeout (minutes):

Resumption Timeout (hours): 48

Compression: none

Preferred Cipher: blowfish-cbc

Preferred MAC: hmac-sha1

Required Authentication: Password or Public Key

Maximum Logins:

Maximum Logins Per User:

Payload Repository

Mailbox

File System

< Back Next > Cancel Save

Figure 7-58 Adapter configuration page

5. The SFTP Server Adapter:Configuration:Document Storage page (Figure 7-59) shows a default document storage type of file system. Accept the default setting and click **Next**.

Services Configuration

SFTP Server Adapter: Configuration: Document Storage

Document Storage Type

File System

Database

System Default

< Back > Next Cancel Save

Figure 7-59 SFTP Server Adapter: Configuration: Document Storage

6. For the SFTP Server Adapter: Add Policies page, no additional policies are necessary (Figure 7-60). Click **Next**.

Services Configuration

SFTP Server Adapter: Add Policies

Policy Type
+add Policy Type

< Back > Next Cancel Save

Figure 7-60 SFTP Server Adapter: Add Policies

7. The SFTP Server Adapter: Configuration page has two options (Figure 7-61). For this scenario, select the following answers:
 - a. For the “Should the adapter be restricted to a certain group of users?” option, leave the default setting of No. You might consider changing this option in a more secure system.
 - b. For the “Should users start in the directory that matches their user name upon login?” option, change the answer to **Yes** so that upon login, the user is placed automatically in a directory that matches the login user ID.

Click **Next**.

Services Configuration

SFTP Server Adapter: Configuration

Should the adapter be restricted to a certain group of users?

No

Yes

Should users start in the directory that matches their user name upon login?

No

Yes

< Back > Next Cancel Save

Figure 7-61 Choose to start the users in the directory that matches their user name

- Accept the defaults on the SFTP Server Adapter: Extractability page (Figure 7-62). Click **Next**.

Services Configuration

SFTP Server Adapter: Extractability

Extractable Count:
 Extractable For: Days Hours Minutes
 Extractable:

>

Figure 7-62 Accept the extractability defaults

- Review the SFTP Server Adapter:Confirm page (Figure 7-63) to verify the configuration changes.

SFTP Server Listen Port: Make note of the value given to the SFTP Server Listen Port shown on this confirmation page. This port is the value that the mediation server in the DMZ needs to direct SFTP connections to the SFTP Server Adapter in Sterling B2B Integrator. In our environment, SFTP communications come into Sterling B2B Integrator over port 8119.

Click **Finish**.

Services Configuration

SFTP Server Adapter: Confirm

Enable Service for Business Processes

Service Settings

Service Name	SFTP Server Adapter
Service Type	SFTP Server Adapter
Description	SFTP Server Adapter
System Name	SFTP_SERVER_ADAPTER
Group Name	None provided
Perimeter Server	node1 & local
Enabled Protocols	SFTP
Host Identity Key	sysc
SFTP Server Listen Port	8119

Figure 7-63 Take note of the port number

- When returned to the Services Configuration page, select **Enabled** to start the SFTP Server Adapter (Figure 7-64).

Enabled	Select	Name ▲▼	Type ▲▼	Advanced State	
<input checked="" type="checkbox"/>			SFTP Server Adapter	SFTP Server Adapter	Enabled

Figure 7-64 Enable the SFTP Server Adapter

7.3.15 Creating the trading partner for SFTP

In this scenario, WebSphere Message Broker sends a file out to an external business partner over SFTP. To create the SFTP trading partner:

- From the Sterling File Gateway Administration Console, select **Participants** → **Partners** from the top navigational menu (Figure 7-65).



Figure 7-65 Participants menu options - Partners

- Add a new partner to Sterling File Gateway by clicking **Create** on the Partners tab (Figure 7-66).

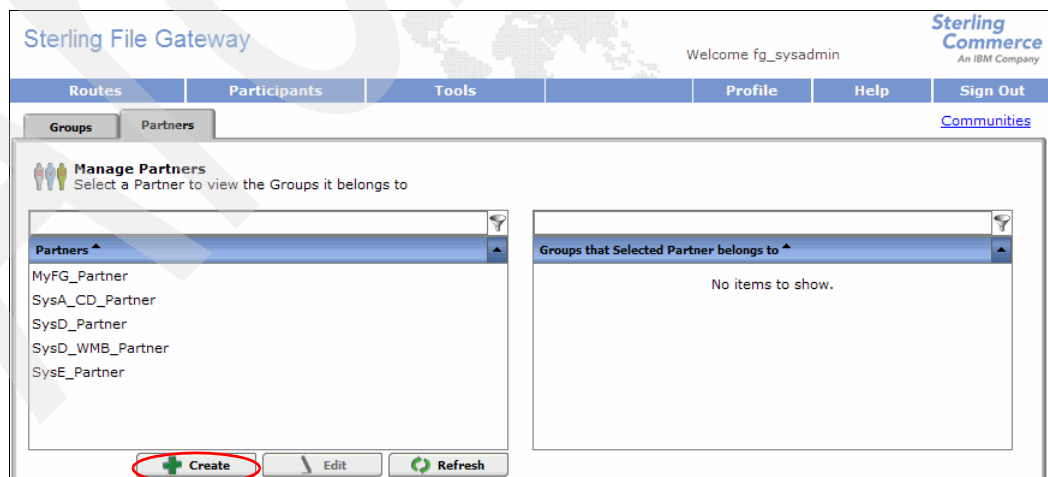


Figure 7-66 Create a new partner

3. Select your community (Figure 7-67). In our scenario, the community is **FirstCommunity**. Click **Next**.



The screenshot shows a dialog box titled "Add Partner". Inside, there is a section titled "Select Community" with a blue header. Below the header, the text reads: "Every partner belongs to a community. Please select the community to which this partner will belong." There is a label "Community:" followed by a dropdown menu showing "FirstCommunity". At the bottom right, there are two buttons: "Next" and "Cancel".

Figure 7-67 Select the community for the new partner to join

4. Create the SFTP business partner. This partner uploads and downloads files from Sterling File Gateway using only SFTP. Enter a name for the partner. We chose to use the name of SFTP_Partner.

The phone number and email address fields are mandatory even though we have notifications disabled in this example. We used a false telephone number and email address for this example (Figure 7-68).



The screenshot shows a dialog box titled "Add Partner: Information". Inside, there is a section titled "Contact Information" with a blue header. Below the header, there are several fields: "*Partner Name:" with the value "SFTP_Partner"; "Address:" with two empty text boxes; "City:" with one empty text box; "State:" with one empty text box; "Postal Code:" with one empty text box; "*Phone:" with the value "9999"; "Country:" with a dropdown menu showing "UNITED STATES"; "Time Zone:" with a dropdown menu showing "(GMT-05:00) Eastern Time (US & Canada)"; and "*Email Address:" with the value "sftp@itso_redbooks.com". At the bottom right, there are three buttons: "Back", "Next", and "Cancel".

Figure 7-68 Enter contact information for the SFTP_Partner external business partner

5. Enter a user name and password for the SFTP_Partner external trading partner (Figure 7-69). These values are used for the SFTP client to authenticate with Sterling B2B Integrator. This credential allows an external trading partner to place files into or retrieve files from SFTP_Partner's mailboxes. We used SFTP_Partner as the user name. Click **Next**.

Add Partner: User Account

User Account

Create a user account by filling the following information. This user account is to be used by SFTP_Partner when logging into myFileGateway.

* User Name: SFTP_Partner

* Password: ●●●●●●

* Confirm Password: ●●●●●●

* First Name: SFTP_Firstname

* Last Name: SFTP_Lastname

< Back > Next Cancel

Figure 7-69 Create a user account for the SFTP_Partner external business partner

6. In our scenario, the created partner, SFTP_Partner, retrieves files residing in its mailbox (Consumer of Data). That is, this partner is used for only outbound file transfers. From the Select Partner Role page (Figure 7-70), configure a partner that is used for only outbound file transfers:
 - a. Select only **Consumer of Data**.
 - b. Under the Consumer of Data section, select the **Partner will initiate the connection to consume data** option. This means that SFTP_Partner connects to its designated mailbox using SFTP to consume any files residing there. Click **Next**.

Add Partner: Partner Role

Select Partner Role

There are no consumers configured in the system right now. It is recommended that the first partner that is configured is one that is a consumer or both a consumer and a producer.

SFTP_Partner is a Consumer of Data

- Partner will initiate the connection to consume data.
- Partner will listen for a connection to consume data.

SFTP_Partner is a Producer of Data

< Back > Next Cancel

Figure 7-70 Partner role page

7. On the Initiate Connection Settings page (Figure 7-71), answer the following questions:
- For the “Will SFTP_Partner use either SSH/SFTP or SSH/SCP protocol to initiate connections” option, select **Yes**. The SFTP_Partner uses SFTP to transmit data from Sterling File Gateway.
 - For the “Will SFTP_Partner use an Authorized User Key to authenticate?” option, select **No**. The SFTP_Partner authenticates in a batch job with the user ID and password configured in Sterling File Gateway. In this case, Sterling B2B Integrator does not require the use of an Authorized User Key. Refer to your security policies to determine whether this practice is appropriate for your organization’s production environment.

Click **Next**.

Add Partner: Initiate Connections Settings

Initiate Connections Settings

Will SFTP_Partner use either SSH/SFTP or SSH/SCP protocol to initiate connections?

Yes No

Will SFTP_Partner use an Authorized User Key to authenticate?

Yes No

< Back > Next Cancel

Figure 7-71 Initiate Connections Settings page

8. On the PGP Settings page, accept the default values of No (Figure 7-72). Click **Next**.

PGP Settings

PGP Settings

Does SFTP_Partner require data to be signed by the Router ?

Yes No

Does SFTP_Partner require data to be encrypted by the Router ?

Yes No

< Back > Next Cancel

Figure 7-72 PGP Settings

- Review your choices on the confirmation page (Figure 7-73). Click **Finish** to complete the creation of your partner, SFTP_Partner.

Add Trading Partner: Confirm	
Profile	
Partner Name:	SFTP_Partner
Address:	.
Phone:	9999
Email Address:	sftp@itso_redbooks.com
User Account	
User Name:	SFTP_Partner
Password:	*****
First Name:	SFTP_Firstname
Last Name:	SFTP_Lastname
Protocol	
Partner Role:	Consumer of Data
Connection Direction:	Initiate Connection
Will SFTP_Partner use either SSH/SFTP or SSH/SCP protocol to initiate connections?	Yes
Will SFTP_Partner use an Authorized User Key to authenticate?	no
Selected Authorized User Key	SFTP_Partner_PublicKey
Transport Method:	MAILBOX
Does SFTP_Partner require data to be signed by the Router :	no
Does SFTP_Partner require data to be encrypted by the Router :	no
Community Membership	
Community Name:	FirstCommunity
Joined Date:	Today
<input type="button" value="Back"/> <input type="button" value="Cancel"/> <input type="button" value="Finish"/>	

Figure 7-73 Confirmation page

- When the partner is added successfully, close the window and return to the main Sterling File Gateway Administration Console page.

7.3.16 Creating an inbound routing channel

In this section, we explain how to create the inbound routing channel to define the route from MyFG_Partner to SysD_WMB_Partner. This routing channel allows the external user MyFG_Partner to log into myFileGateway and put files in their root mailbox. The routing channel forwards the files on to the SysD_WMB_Partner internal partner. SysD_WMB_Partner is configured to send the file to SysD using WebSphere MQ File Transfer Edition and place it in a directory on the file system where it is processed by WebSphere Message Broker.

To create an inbound routing channel:

1. In the Sterling File Gateway Administration Console browser, from the top navigational menu select **Routes** → **Channels**. Click **Create** (Figure 7-74).

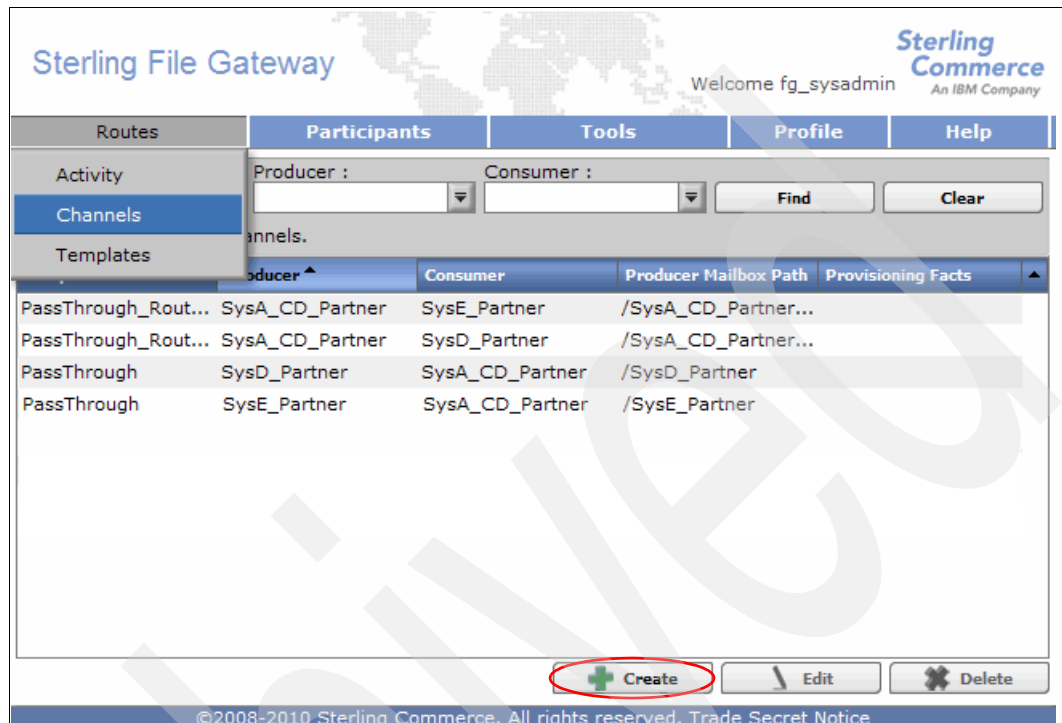


Figure 7-74 Create a channel in Sterling File Gateway

- Use the values shown in Table 7-8 to create a new routing channel (Figure 7-75). The values for producer and consumer are populated by the business partners that are created. Click **Save**.

Table 7-8 Values for routing channel

Parameter	Value
Template	PassThrough
Producer	MyFG_Partner
Consumer	SysD_WMB_Partner

Figure 7-75 Create a new routing channel

- A success message opens when the channel is created (Figure 7-76). Click **OK**.

Figure 7-76 Successfully created Routing Channel message

7.3.17 Creating an outbound routing channel

In this section, we describe how to create two outbound routing channels to define the route from SysD_WMB_Partner to MyFG_Partner and the route from SysD_WMB_Partner to SFTP_Partner. When WebSphere Message Broker has a file to be delivered to MyFG_Partner, WebSphere MQ File Transfer Edition gets the file and places it in SysD_WMB_Partner's mailbox, /To_MyFG_Partner/, where it is picked up by Sterling File Gateway and routed to the mailbox /MyFG_Partner/Inbox. The file remains in this mailbox until MyFG_Partner logs in to myFileGateway and downloads the file.

When WebSphere Message Broker has a file to be delivered to SFTP_Partner, WebSphere MQ File Transfer Edition gets the file and places it in SysD_WMB_Partner's mailbox, /To_SFTP_Partner/, where it is picked up by Sterling File Gateway and routed to the mailbox /SFTP_Partner/Inbox. The file remains in this mailbox until SFTP_Partner runs a batch processing job overnight to log into the mailbox and retrieve files.

To create an outbound routing channel:

1. In the Sterling File Gateway Administration Console, use the top navigational menu to select **Routes** → **Channels** (Figure 7-77). Click **Create**.

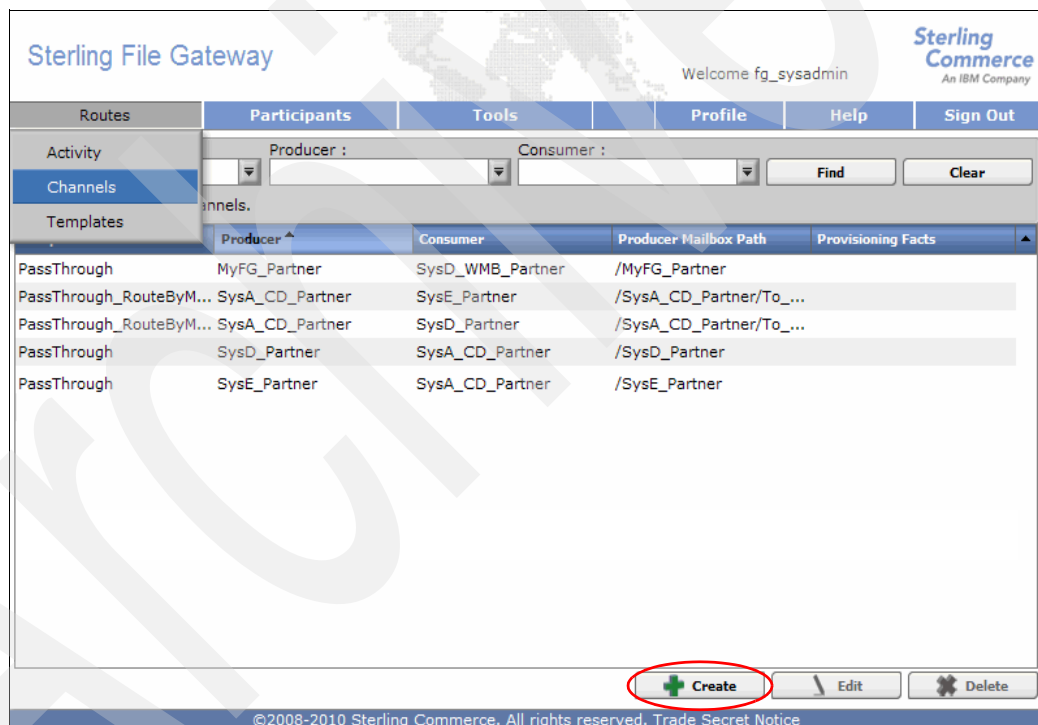


Figure 7-77 Create a channel in Sterling File Gateway

- Use the values shown in Table 7-9 to create a new routing channel (Figure 7-78). Click **Save**.

Table 7-9 Values for routing channel

Parameter	Value
Template	PassThrough_RouteByMailbox
Producer	SysD_WMB_Partner
Consumer	MyFG_Partner

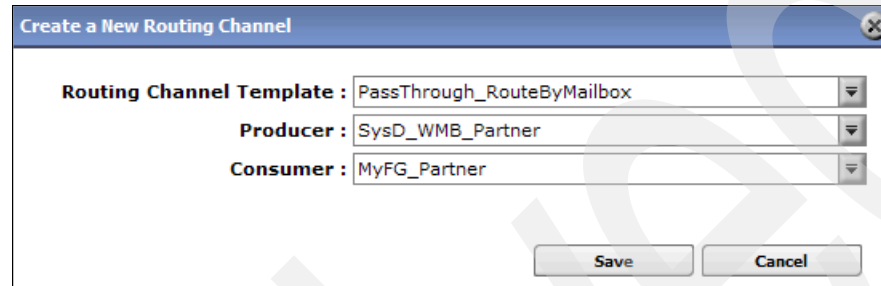


Figure 7-78 Create a new routing channel

- A success box opens when the channel is created (Figure 7-76 on page 316). Click **OK**.

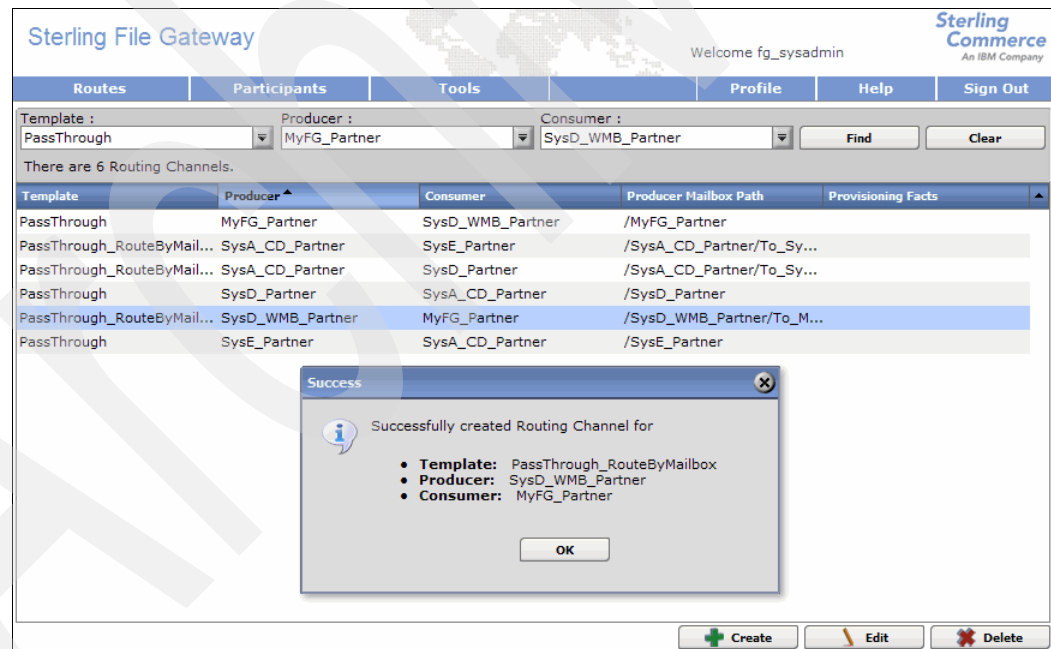


Figure 7-79 Successfully created Routing Channel message

- Repeat step 1 to select the menu options to create a new routing channel. Use the values listed in Table 7-10 for the routing channel (Figure 7-80). Click **Save**.

Table 7-10 Values for routing channel

Parameter	Value
Template	PassThrough_RouteByMailbox
Producer	SysD_WMB_Partner
Consumer	SFTP_Partner

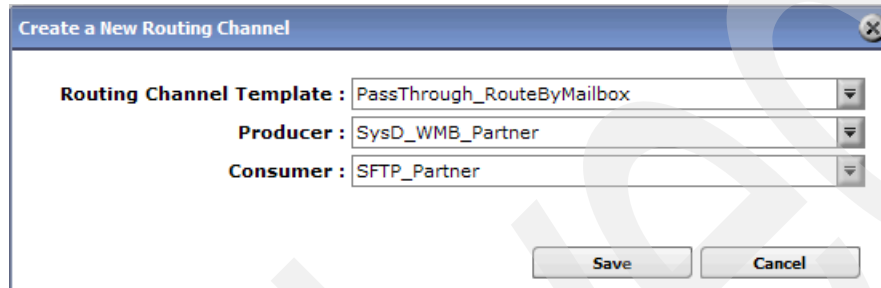


Figure 7-80 Create a new routing channel

- A success box opens when the channel is created (Figure 7-81). Click **OK**.

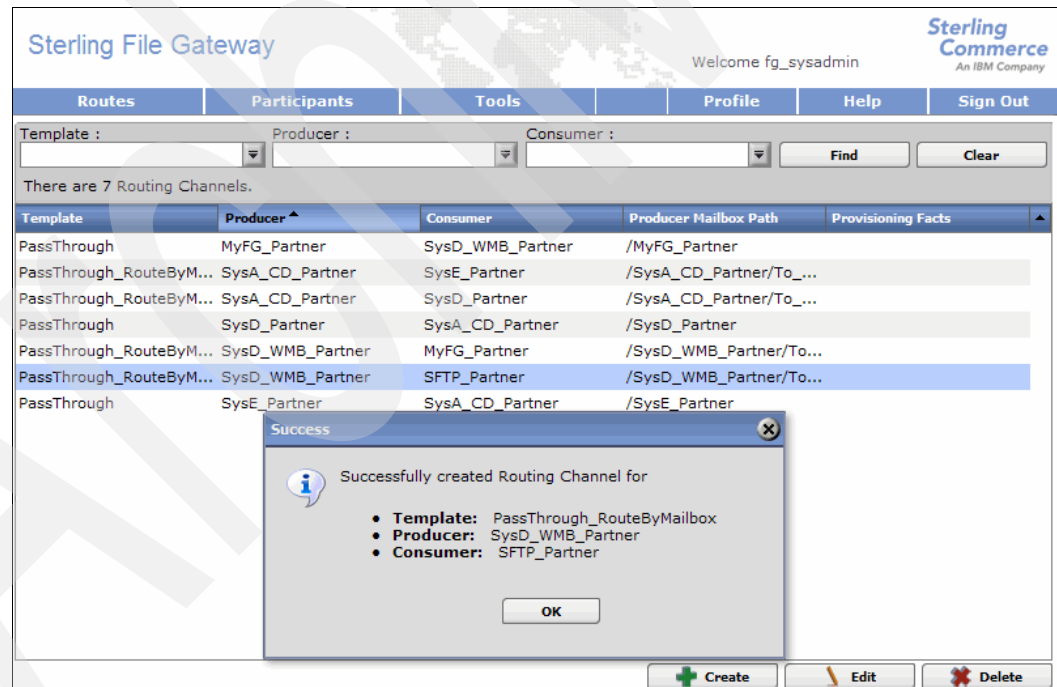


Figure 7-81 Creation of the routing channel from SysD_WMB_Partner to SFTP_Partner

7.3.18 Importing key certificate files in Sterling B2B Integrator for HTTPS

To communicate with Sterling B2B Integrator over HTTPS, you need to import key certificates into Sterling B2B Integrator. HTTPS is used to create a more secure network connection over an insecure, public network. The use of HTTP over an SSL encrypted connection provides

protection from a data transmission session being attacked or snooped. Most commonly, HTTPS trust is based on the exchange of major certificate authorities that come installed with many browsers.

The HTTP Server provides a certificate, made and maintained by individual organizations or provided and maintained for an organization by one of the major certificate authorities. The transmission of the certificate from the HTTP server to the web browser allows the trusted certificate authority to provide automated verification that the server is who it claims to be. For individually made certificates, the client can view the certificate in the web browser and choose to add and accept the certificate. The acceptance of this certificate indicates that the client and server know and trust the identity provided, which establishes the secure session.

We used a widely available tool for generating system certificates for the purposes of this book. We created one certificate file (trusted.txt), which contains trusted certificate information for all nodes in our environment. Each node also has its own system certificate (keycert_<machine_name>.txt). These are the keys imported into Sterling B2B Integrator in this section.

Sterling B2B Integrator Administration Console is used to import the certificates. Refer to “Logging in to Sterling B2B Integrator” on page 300 for information about how to access the console.

Follow your organization’s guidelines to create your system certificates. Then complete the following steps to import the certificates into Sterling B2B Integrator:

1. From the Sterling B2B Integrator Administration Console, use the left navigational menu to select **Trading Partner** → **Digital Certificates** → **Trusted**.

When Trusted is selected, the right pane of the browser displays (Figure 7-82). Click **Go!** next to New Certificate.

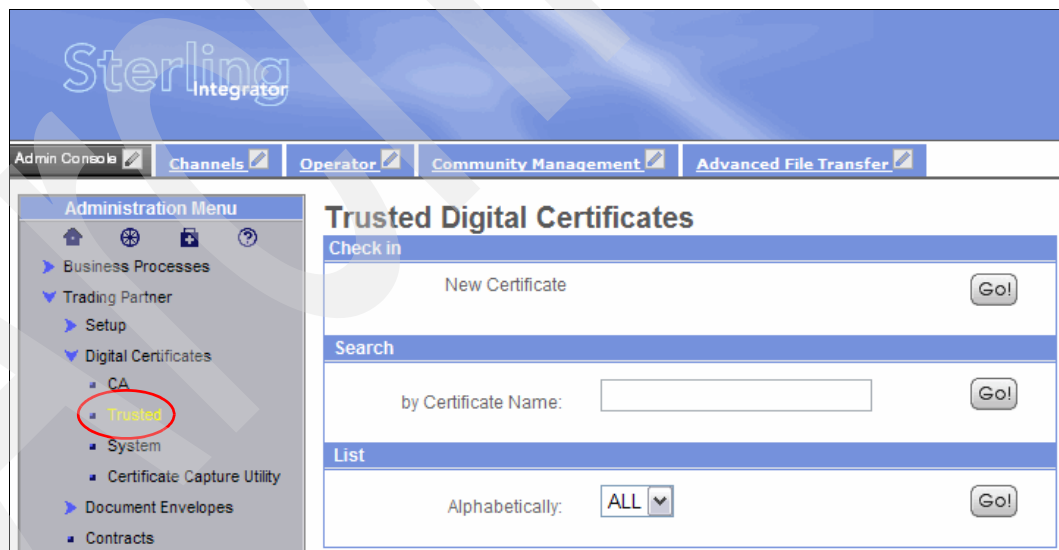


Figure 7-82 Sterling B2B Integrator Trusted Digital Certificates page

2. On the Get Certificate page (Figure 7-83), browse to the location of your trusted key certificate file and select your digital certificate. Click **Next**.

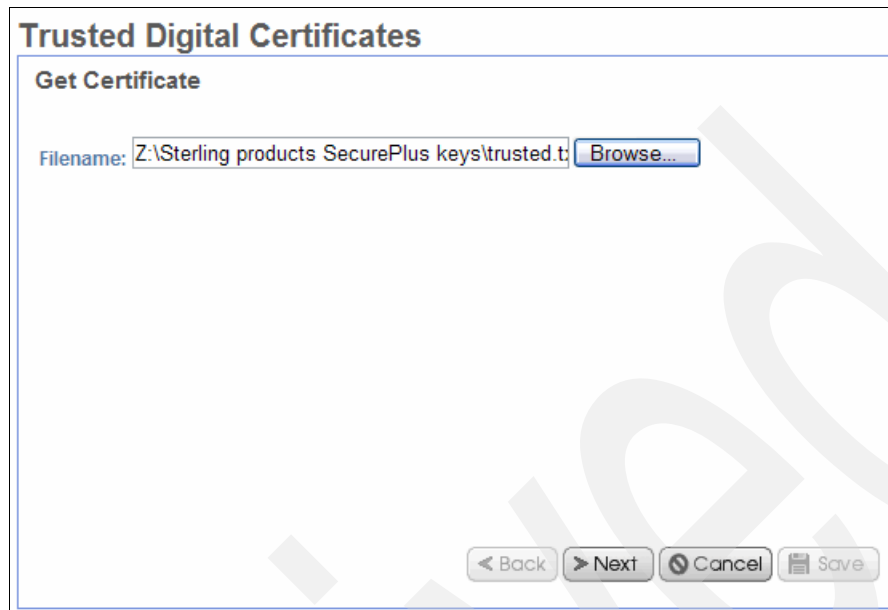


Figure 7-83 Add trusted digital certificate

Certificate note: In this scenario, we created one key file for all nodes in our environment. Follow your organization's policy for creating and obtaining keys.

3. The Naming page shows the contents of the certificate, which indicates the systems that are included. Scroll to the bottom and click **Next**.
4. Review the Confirmation page. Click **Finish**.

5. From the Sterling B2B Integrator Administration Console, use the left navigational menu to select **Trading Partner** → **Digital Certificates** → **System**. After you select System, the right pane of the browser displays (Figure 7-84). Click **Go!** next to Key Certificate.

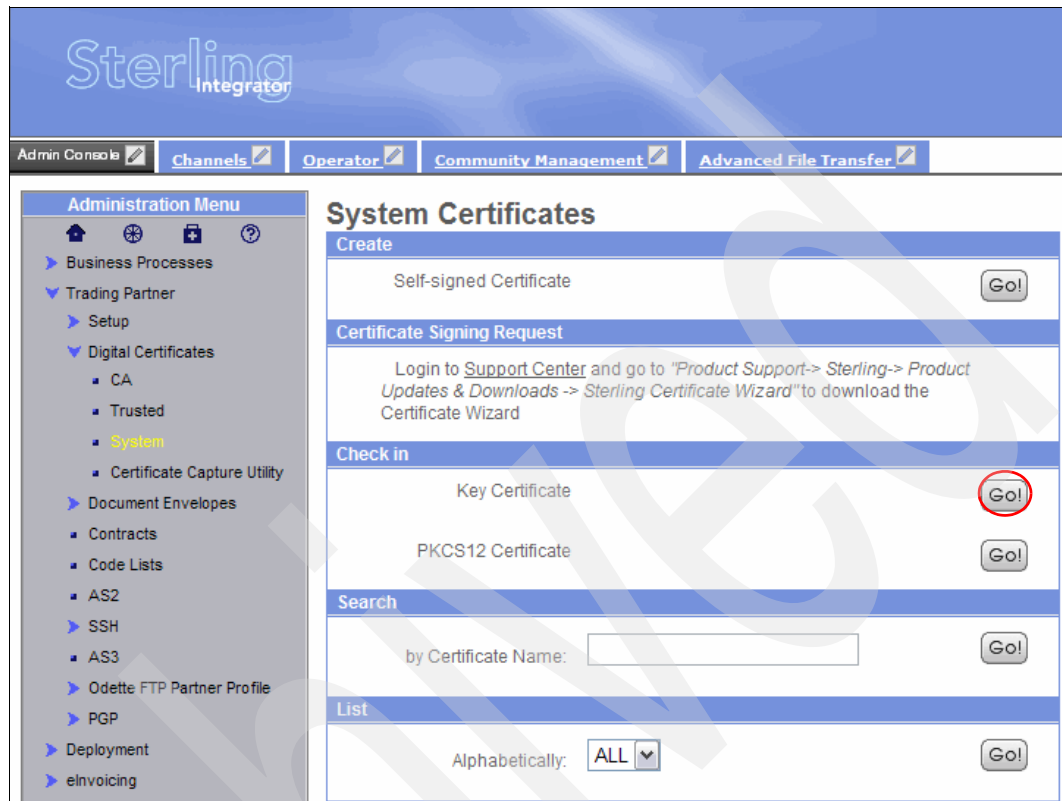


Figure 7-84 Sterling B2B Integrator System Certificates

- The Key Certificate Data page allows you to enter a name for the certificate for your Sterling B2B Integrator node, create a private key password, and import the key certificate that you made. As shown in Figure 7-85, we named our certificate SysC_Certificate, entered a private key password, and uploaded the key certificate that we created for SysC. Click **Next**.

Key Certificates

Key Certificate Data

Certificate Name:

Private Key Password:

Filename (.txt):

Figure 7-85 Save key certificate in Sterling B2B Integrator

Key certificate note: We created our own key certificate for the Sterling B2B Integrator. Follow your organization’s policy for obtaining or generating keys.

7. The imported certificate can be validated. Figure 7-86 shows the validation page for the key certificate. By default, no options are selected. Leave the default choice of no selections. Click **Next**.

Key Certificates

SysC_Certificate: Validate When Used

Validate When Used: Validity Auth Chain

< Back > Next Cancel Save

Figure 7-86 Validate key certificates

8. Review the Confirmation page (Figure 7-87) and click **Finish**.

Key Certificates

SysC_Certificate: Confirm

Certificate Specification	
Certificate Name	SysC_Certificate
Filename	Z:\Sterling products SecurePlus keys\keycert_sysc.bt
Validate When Used	None
Status	Date valid Self-signed Verified

< Back > Next Cancel Finish

Figure 7-87 Key certificate confirmation page

7.3.19 Configuring Sterling B2B Integrator and myFileGateway for HTTPS

The default installation of Sterling File Gateway and myFileGateway use the same HTTP port value as the Sterling B2B Integrator Administration Console. The default for this port is 8080. These consoles can also be accessed over HTTPS by the default HTTPS port value of 8081.

myFileGateway is included with Sterling File Gateway. When myFileGateway is configured, it is configured by default to use the HTTP Server adapter that is included in the installation of Sterling File Gateway. When using the default installation of myFileGateway, the default HTTP Server adapter cannot be configured for specific applications or to use custom key certificates. This configuration can create a security concern because the usage of the default ports allows the mediation server in the DMZ setup to direct trading partners to myFileGateway to forward requests onto the default port from external users. This redirection makes the Sterling File Gateway and Sterling B2B Integrator administration console available publicly unless other configuration is performed. The administration consoles still require a user to have a valid user ID and password to gain admission to administrative functions for either Sterling File Gateway or Sterling B2B Integrator.

To avoid making the administration consoles publicly available in our scenario, we installed a new HTTP Server Adapter on a different port in Sterling B2B Integrator, configured the adapter to use HTTPS, and defined custom key certificates. This setup also allows us to configure our proxy server in the DMZ to forward HTTPS requests to a port specifically designated for HTTPS communication with myFileGateway.

You use the Sterling B2B Integrator Administration Console to install and configure the HTTP Server adapter. For information about how to access the Sterling B2B Integrator Administration Console, refer to “Logging in to Sterling B2B Integrator” on page 300.

To configure the new HTTP Server adapter:

1. In the Sterling B2B Integrator console, in the left navigational menu select **Deployment** → **Services** → **Configuration**. Figure 7-88 shows the page that displays the Services Configuration after Configuration is selected. Click **Go!** next to New Service.

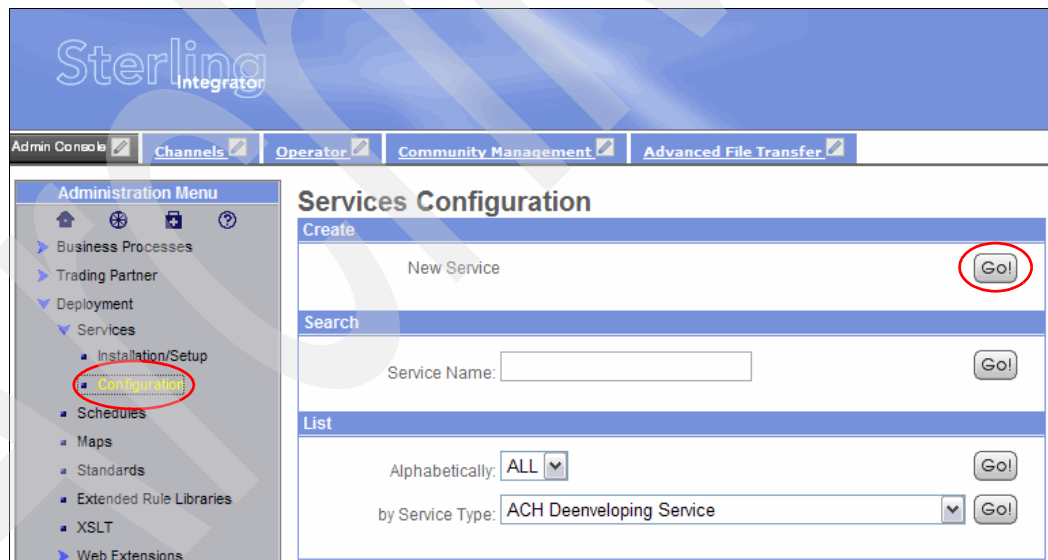


Figure 7-88 Create a new service

- Using the directory tree shown in Figure 7-89, select a service type. Expand the **Communications** directory and select **HTTP Server Adapter**. Click **Save**.

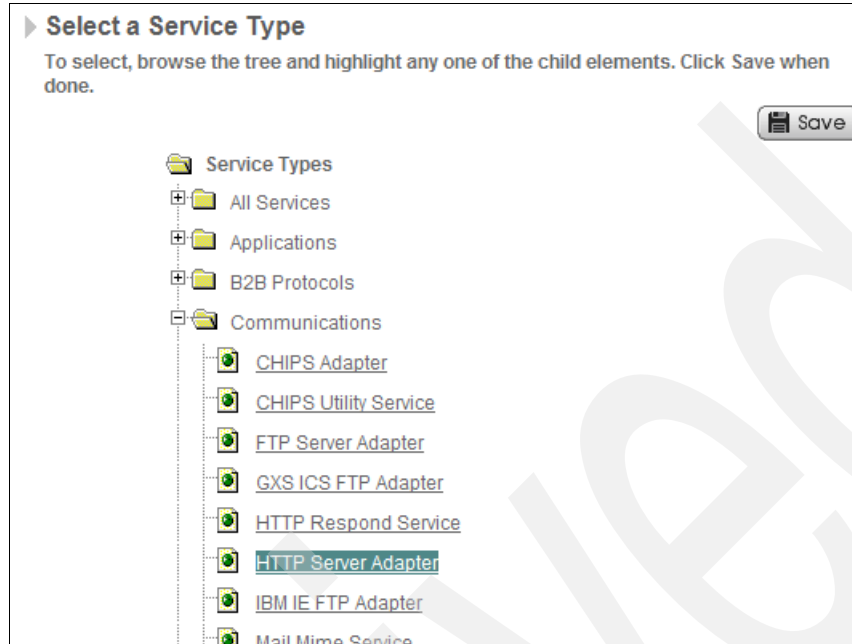


Figure 7-89 Select HTTP Server Adapter from the service type directory tree

- The selection of HTTP Server Adapter populates the service type shown on the Services Configuration: Select Service Type page (Figure 7-90). Click **Next**.

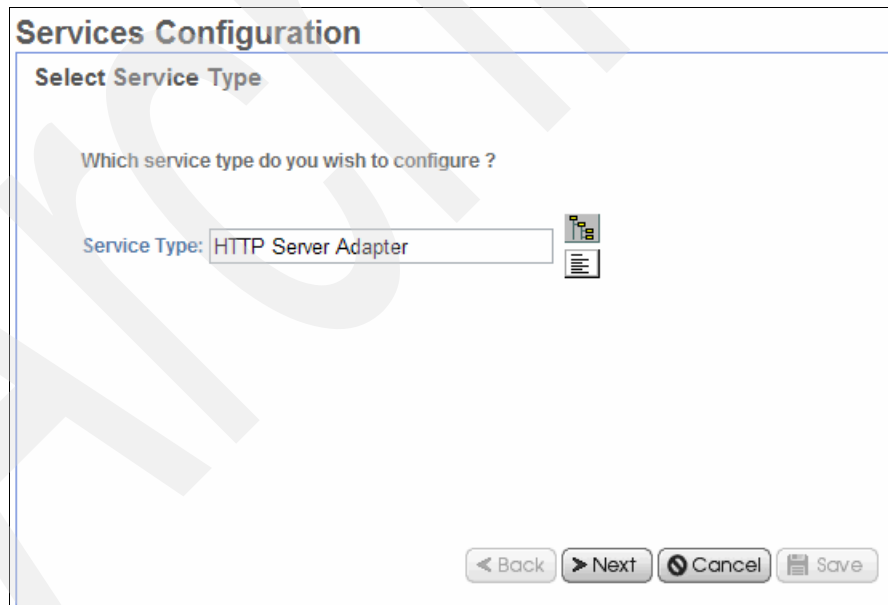


Figure 7-90 Select a service type

4. On the HTTP Server Adapter:Name page (Figure 7-91), create a name for the new HTTP Server Adapter. In our scenario we used the following values:
- Name: myFG_HTTP_Server
 - Description: Secure HTTP Server for myFileGateway
 - Leave the default selection of None to not select a group for the adapter.

Click **Next**.

The screenshot shows a configuration window titled "Services Configuration" with a sub-header "HTTP Server Adapter: Name". It contains three text input fields: "Name" with the value "myFG_HTTP_Server", "Description" with the value "Secure HTTP Server for myFileGateway", and "Select a group:" with the "None" radio button selected. Below these are three radio buttons: "None" (selected), "Create New Group" (with an empty text box), and "Select Group:" (with a dropdown arrow). At the bottom right are four buttons: "Back", "Next", "Cancel", and "Save".

Figure 7-91 Enter a name and description for the HTTP Server Adapter

5. On the HTTP Connection Properties page (Figure 7-92), enter configuration values specifically for the new HTTP Server Adapter. Determine and enter a value for the HTTP Listen Port that is used to connect to myFileGateway, the Total Business Process queue depth threshold.

Services Configuration

myFG_HTTP_Server: HTTP Connection Properties

HTTP Listen Port:

Perimeter Server Name:

Total Business Process queue depth threshold:

Document Storage

System Default

Database

File System

User Authentication Required

Yes

No

Use SSL (Note: User Authentication without SSL will result in a weak security configuration)

None

Must

Figure 7-92 Values for the new HTTP Server Adapter

We used the following values for our scenario:

- HTTP Listen Port: 10000
- Total Business Process queue depth threshold: 10

We accepted the default value of 10. This value has no effect on this server adapter because myFileGateway is the only application using this adapter. myFileGateway does not initiate business processes.

- Document Storage: System Default
This option is the default selected option.

- User Authentication Required: No

This option prompts the user for a user ID and password before loading to the first page of myFileGateway, which also contains a prompt page for a user ID and password. Entering the same values twice is unnecessary, so we disable this setting.

- Use SSL: Must

Click **Next**.

6. On the SSL Settings page (Figure 7-93), configure certificates for HTTPS according to your organization's security policy and practices. If your organization uses CA certificate, you can select them from this page.

For our scenario, we selected the certificate for the Sterling B2B Integrator system that was imported earlier, SysC_Certificate.

Click **Next**.

The screenshot shows a web-based configuration interface titled "Services Configuration". The main heading is "myFG_HTTP_Server: HTTP Connection Properties: SSL Settings". Below this, there are several configuration options:

- System Certificate:** A dropdown menu currently showing "SysC_Certificate".
- Cipher Strength:** A dropdown menu currently showing "STRONG".
- CA Certificate:** A section with a "Filter:" input field and a search icon. Below it is a large empty box containing the text "[No CA Certificates Available]". To the right of this box are four arrow buttons: two pointing right and two pointing left.

At the bottom of the configuration area, there are four buttons: "< Back", "> Next", "Cancel", and "Save".

Figure 7-93 Select the system certificate

- To add a new URI, click **add** (Figure 7-94). The URI is the extension to the address that will be used to access applications using this HTTP Server Adapter.

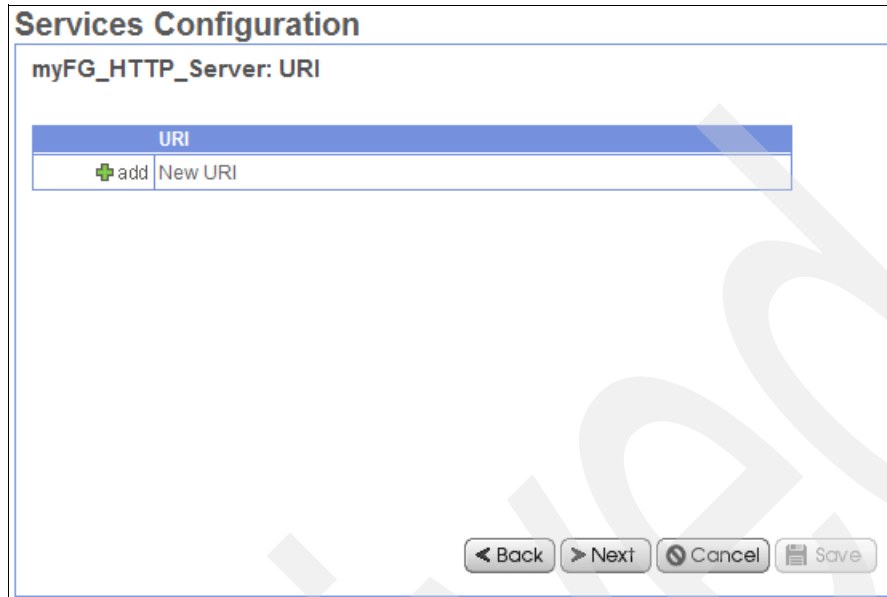


Figure 7-94 Add a new URI to the adapter

- On the URI Config (Figure 7-95), enter a URI for the HTTP Server Adapter and choose to launch a WAR file.

In our scenario, we used the following values:

- A URI of /myfilegateway
- Selected **War File** to launch the myFileGateway application

Click **Next**.

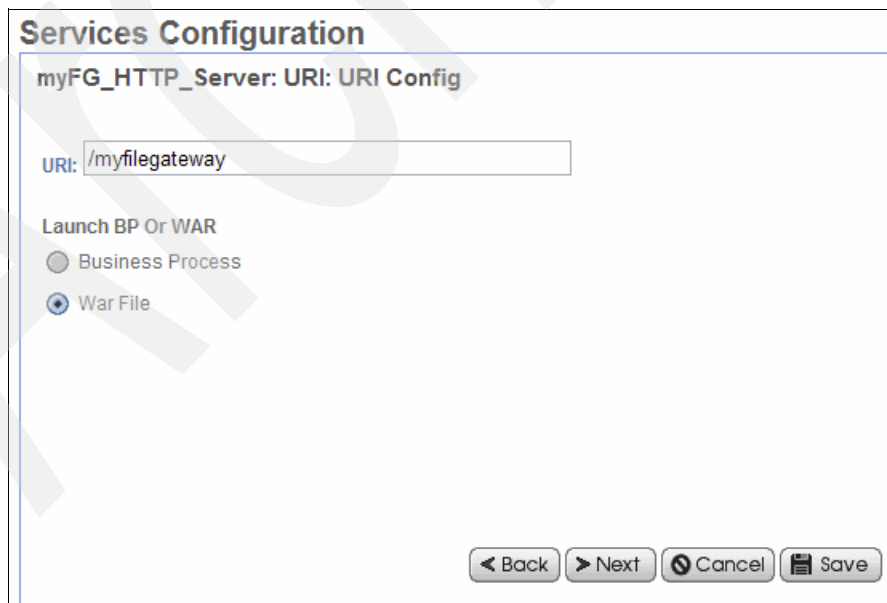


Figure 7-95 Enter a URI and select a process to launch from the HTTP Server Adapter

9. On the WAR Config page (Figure 7-96), specify the location of the WAR file. In our scenario, the WAR file is the `myfilegateway.war` file. This WAR file is in the directory structure `<SI_install_root>\container\Applications\myfilegateway.war`. Enter the path name for your WAR file. Click **Next**.

Services Configuration

myFG_HTTP_Server: WAR Config

Enter War File Path:

OR

Load a System Generated War File: No Generated War Files




< Back > Next Cancel Save

Figure 7-96 Enter the path for the `myfilegateway.war` file

10. The URI is added and configured for the HTTP Server Adapter. It can be seen as the URI name `/myfilegateway` (Figure 7-97). Click **Next**.

Services Configuration

myFG_HTTP_Server: URI

URI	
 add	New URI
 edit	 delete /myfilegateway

< Back > Next Cancel Save

Figure 7-97 `/myfilegateway` URI successfully added to the HTTP Server Adapter

11. Review the Confirmation page. Verify that the Enable Service for Business Processes option is selected (Figure 7-98). Click **Finish**.

Services Configuration

myFG_HTTP_Server: Confirm

Enable Service for Business Processes

Service Settings

Service Name	myFG_HTTP_Server
Service Type	HTTP Server Adapter
Description	Secure HTTP Server for myFileGateway
System Name	myFG_HTTP_Server
Group Name	None provided
HTTP Listen Port	10000
Perimeter Server Name	node1 & local
Total Business Process queue depth threshold	10
Document Storage	System Default
User Authentication Required	No
Use SSL (Note: User Authentication without SSL will result in a weak security configuration)	Must
System Certificate	SysC_Certificate
Cipher Strength	STRONG
CA Certificate	NONE
URI	URI: /myfilegateway War File Path: C:\SCS\install\container\Applications\myfilegateway.war

Figure 7-98 HTTP Server Adapter Confirmation page

12. To test that the HTTP Server Adapter is working:
 - a. Open a browser and navigate to `https://<your_server_name>:10000/myfilegateway`.
 - b. Accept any browser-issued security warnings.
 - c. Verify that you see the myFileGateway login page.
 - d. Close the browser.

Note: You might need to exchange key certificates between Sterling File Gateway and the systems that you are using to access the Sterling File Gateway node. This exchange should be trivial by accepting security prompts in the browser window or SFTP client. Contact your security administrator for assistance if needed.

7.4 Testing the flows

This file transfer scenario is initiated when the external trading partner from Company A, MyFG_Partner, logs in to myFileGateway and uploads a file. WebSphere Message Broker is configured to look for an XML file to be placed in the directory specified on the Basics tab of the FTEInput node. MyFG_Partner has no knowledge of where the file will be sent within Company B. MyFG_Partner must know if a specific type of file is necessary. In this case, WebSphere Message Broker is looking for a file ending with a .xml file type extension. If MyFG_Partner sends another file type extension, the file will not be processed by WebSphere Message Broker Explorer.

We describe the movement of the file after Company A's MyFG_Partner has uploaded the file in 7.2.2, "Inbound file transfer flow" on page 253. When the file is consumed by WebSphere Message Broker, it is processed by the flow and sent as output to a WebSphere MQ queue, to myFileGateway, or to Sterling File Gateway for a SFTP client to retrieve. For detailed information about how the WebSphere Message Broker Explorer flow works and is created, refer to Appendix B, "Building the WebSphere Message Broker flow" on page 365.

WebSphere Message Broker uses two different FTEOutput nodes to send the output file to either myFileGateway or Sterling File Gateway for SFTP. The only difference between these nodes is the location for WMBBRIDGEAGT to write the file. We discuss the different locations and how the files are directed in 7.2.3, "Outbound file transfer flow" on page 254.

7.4.1 Testing the flow sending an output file over SFTP

To test this scenario:

1. Figure 7-99 shows how to log in to the myFileGateway page as Company A's external trading partner, MyFG_Partner.

For our scenario, we use a URL built off of the following sample URL:

`https://<your_server_name>:<port>/<context_root>/`

The URL's port number and context root is determined by how you configure your mediation server in the DMZ. Refer to your organization's security policy for more information.

Click **Sign In**.



Figure 7-99 myFileGateway secure login page

- From the Upload tab, select the root mailbox of / and browse to the location of the sample file (Figure 7-100).

For this scenario, we are using a file named RetrieveData.xml. This file contains contents that retrieve data from the database that will be processed by WebSphere Message Broker and that send an output file to Sterling File Gateway for SFTP transmission.

Click **Send**.

The screenshot shows the myFileGateway web interface. At the top left is the logo 'myFileGateway'. At the top right is the Sterling Commerce logo with the text 'Welcome MyFG_Partner' and 'An IBM Company'. Below the logo is a navigation bar with buttons for 'Home', 'Profile', 'Help', and 'Sign Out'. Underneath is a secondary navigation bar with buttons for 'File Activity', 'Upload Files', 'Download Files', and 'Reports'. The main content area is titled 'Upload a file' and includes the instruction 'Specify mailbox file and renaming pattern'. It contains three input fields: 'Mailbox Path' with a dropdown menu showing '/', 'File' with a text box containing 'C:\XMLFilesSendToWMB\RetrieveData.xml' and a 'Browse...' button, and 'Rename File to' with an empty text box. A 'Send' button is located below the input fields. At the bottom of the page is a footer with the text '©2008-2010 Sterling Commerce. All rights reserved. Trade Secret Notice'.

Figure 7-100 Upload file to send to Company B

- A successful upload shows in myFileGateway under the File Activity tab. When the file upload completes successfully, the upload information automatically displays (Figure 7-101).



Figure 7-101 Successful file upload

You can further verify that the transfer completed successfully by using WebSphere MQ File Transfer Edition Explorer (Figure 7-102) on SysD to view the transfer log.

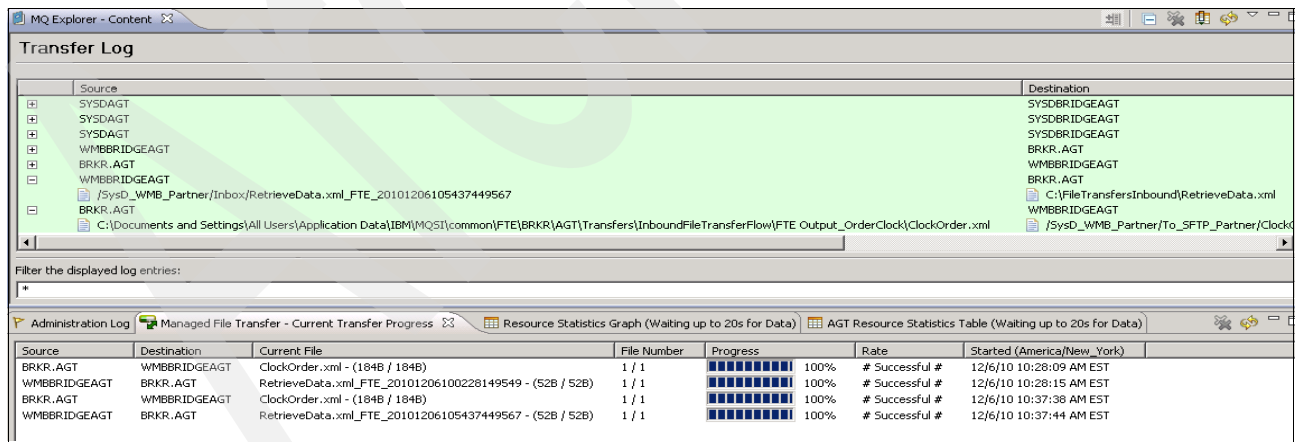


Figure 7-102 Successful file transfer in WebSphere MQ File Transfer Edition Explorer

4. To verify that the file is ready for SFTP_Partner to receive use the following steps:
 - a. Log in to the myFileGateway Console. Figure 7-103 shows how to log into the myFileGateway page as Company A's external trading partner, SFTP_Partner.

For our scenario, we use a URL built off of the following sample URL:

```
https://<your_server_name>:<port>/<context_root>/
```

The URL's port number and context root are determined by how you configure your mediation server in the DMZ. Refer to your organization's security policy for more information.

Click **Sign In**.



Figure 7-103 myFileGateway SFTP_Partner log in page

- b. Click the **Download Files** tab to see whether the ClockOrder.xml file resides in the /SFTP_Partner/Inbox/ mailbox (Figure 7-104). You might need to select **Refresh** at the bottom of the page for the file to appear. When the file appears, it verifies that SFTP_Partner has a file available to be pulled down using SFTP.

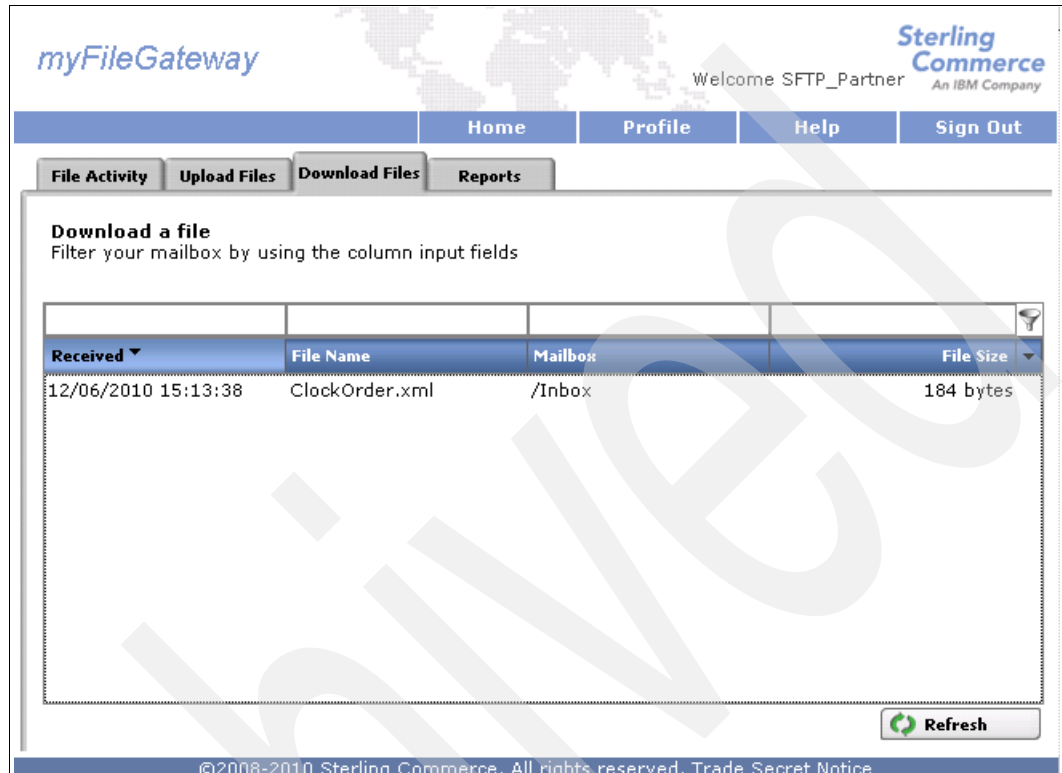


Figure 7-104 myFileGateway file available for SFTP_Partner to download

5. For Company A to receive the file from Company B's Sterling File Gateway, Company A will run a nightly batch job. The contents of the batch job are shown in Example 7-1 on page 256.
6. For Company A to verify that the batch job has successfully run and removed the file from Sterling File Gateway, view your location file system to see whether the file is where you downloaded it.

7. Additionally, you can log into myFileGateway using Company A's SFTP_Partner user ID and password:
- a. Figure 7-105 shows how to log into the myFileGateway page as Company A's external trading partner, SFTP_Partner.

For our scenario, we use a URL built of the following sample URL:

`https://<your_server_name>:<port>/<context_root>/`

The URL's port number and context root are determined by how you configure your mediation server in the DMZ. Refer to your organization's security policy for more information.

Click **Sign In**.



Figure 7-105 myFileGateway SFTP_Partner log in page

- b. When logged in, select the **Download Files** tab. If there are no files listed to download, the SFTP batch job has successfully pulled down all files available to SFTP_Partner through Sterling File Gateway (Figure 7-106).



Figure 7-106 myFileGateway Download Files, no files available

7.4.2 Testing the scenario downloading a file using myFileGateway

To test this scenario:

1. Figure 7-107 shows how to log in to the myFileGateway page as Company A's external trading partner, MyFG_Partner.

For our scenario, we use a URL built off of the following sample URL:

`https://<your_server_name>:<port>/<context_root>/`

The URL's port number and context root are determined by how you configure your mediation server in the DMZ. Refer to your organization's security policy for more information.

Click **Sign In**.



Figure 7-107 myFileGateway secure login page

- From the Upload tab, select the root mailbox of / and browse to the location of the sample file (Figure 7-108).

For this scenario, we use a file named UseRouteNode-KeyNotFound.xml. This file contains contents that contain a work department and other employee information. The work department sent is an invalid department number when compared to departments listed in the Department table of the database. This error results in the file being sent back to myFileGateway for Company A to review.

Click **Send**.

The screenshot shows the myFileGateway web interface. At the top left is the logo 'myFileGateway'. At the top right, it says 'Welcome MyFG_Partner' and 'Sterling Commerce An IBM'. Below this is a navigation bar with 'Home', 'Profile', 'Help', and 'Sign'. A secondary navigation bar has 'File Activity', 'Upload Files', 'Download Files', and 'Reports'. The main content area is titled 'Upload a file' with the instruction 'Specify mailbox file and renaming pattern'. It contains three input fields: 'Mailbox Path' with a dropdown menu showing '/', 'File' with a text box containing 'C:\XMLFilesSendToWMB\UseRouteNode-KeyNotFound.xml' and a 'Browse...' button, and 'Rename File to' with an empty text box. A 'Send' button is located below the input fields. At the bottom of the page, there is a copyright notice: '©2008-2010 Sterling Commerce. All rights reserved. Trade Secret Notice'.

Figure 7-108 Upload file to send to Company B

- A successful upload shows in myFileGateway under the File Activity tab. After the file is successfully uploaded, this information displays automatically (Figure 7-101 on page 336).



Figure 7-109 Successful file upload

You can further verify that the transfer completed successfully by using WebSphere MQ File Transfer Edition Explorer (Figure 7-110) on SysD to view the transfer log.

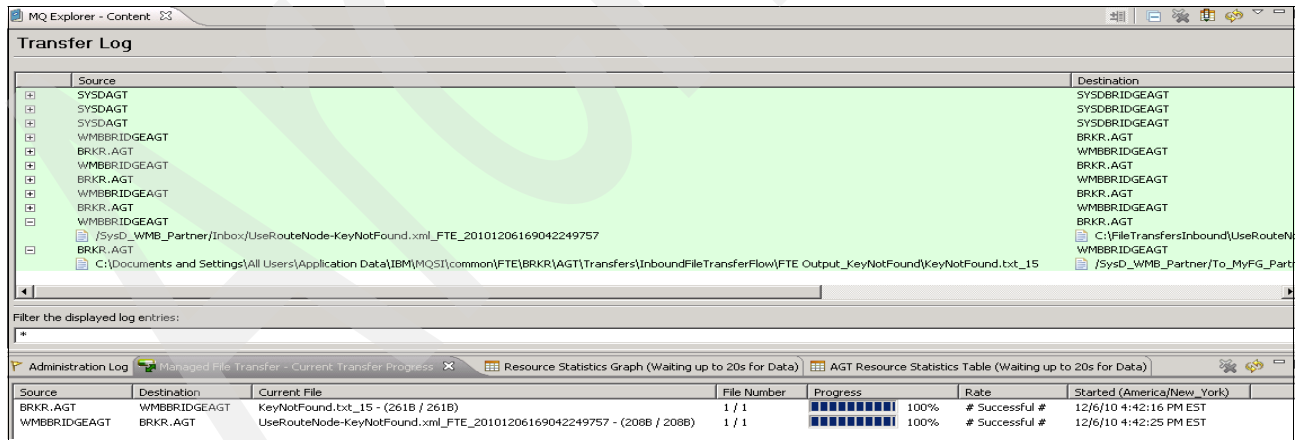


Figure 7-110 Successful file transfer in WebSphere MQ File Transfer Edition Explorer

4. To verify that the error message is available for MyFG_Partner and to download the file, select the **Download File** tab in the myFileGateway console (Figure 7-111).

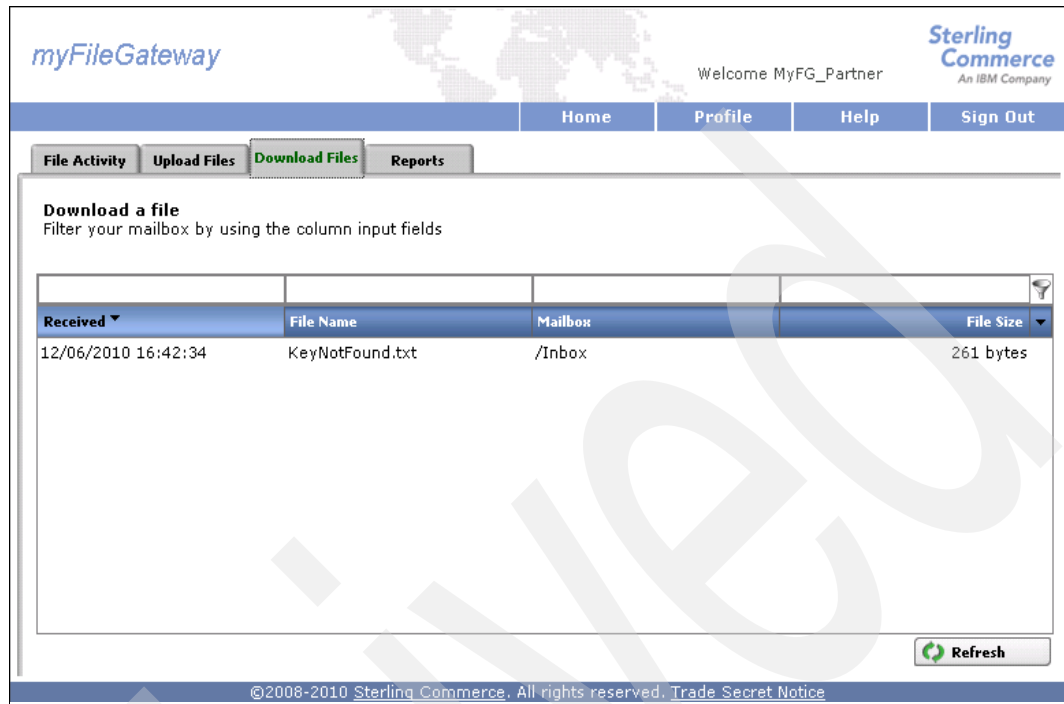


Figure 7-111 myFileGateway Download Files KeyNotFound.txt

5. Click the listing for **KeyNotFound.txt**. Confirm that you want to download the file by clicking **OK** (Figure 7-112).

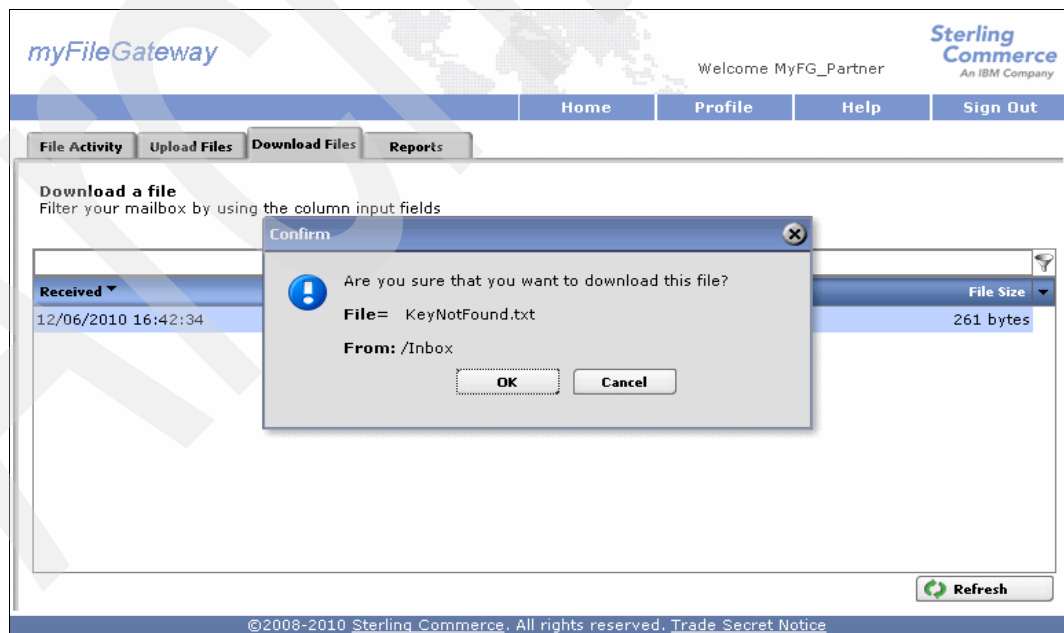


Figure 7-112 Download KeyNotFound.txt

6. A prompt from your browser guides you through opening or saving the file. Choose your preference and save the file to your local file system (Figure 7-113).

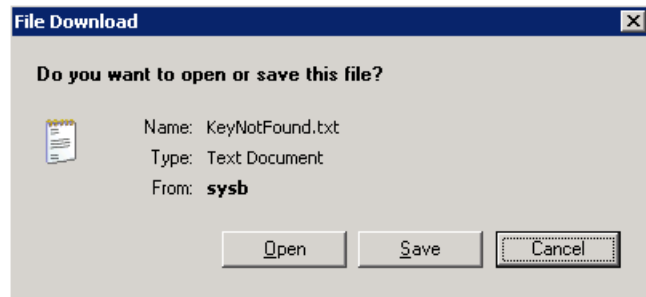


Figure 7-113 Prompt to save file to local file system.

7.5 Troubleshooting tips

The best way to get this scenario working, and the way we went about building the scenario, is to start small and build. For example:

- ▶ Install WebSphere MQ File Transfer Edition and configure it.
- ▶ Test just WebSphere MQ File Transfer Edition to see if you can send files successfully between all of the agents.
- ▶ Install and configure Sterling File Gateway.
- ▶ Test to make sure that the routing channels and mailboxes work by using myFileGateway to upload files and download files, which might require logging in with different user IDs.
- ▶ Install WebSphere Message Broker and build a message flow.
- ▶ Test the message flow by sending a file to the broker agent from a local server agent. Check to see whether you can write a file to the local file system from WebSphere Message Broker using the FTEOutput node.
- ▶ Use WebSphere MQ File Transfer Edition to send files and receive files from Sterling File Gateway.
- ▶ Integrate WebSphere MQ File Transfer Edition, Sterling File Gateway, and WebSphere Message Broker to fully test the scenario.
- ▶ Add security, including the mediation server in the DMZ, and retest components before testing the full integration.

For more general troubleshooting information, refer to the following sections:

- ▶ “Sterling File Gateway and Sterling B2B Integrator” on page 380
- ▶ “WebSphere MQ File Transfer Edition” on page 384
- ▶ “WebSphere Message Broker tips” on page 396

Archived



Configuration of WebSphere MQ File Transfer Edition

This appendix describes how WebSphere MQ File Transfer Edition was configured for use with the scenarios presented.

This chapter contains the following sections:

- ▶ “Overview” on page 348
- ▶ “Configuring WebSphere MQ” on page 349
- ▶ “Configuring WebSphere MQ File Transfer Edition” on page 356

Overview

This appendix describes how we configured WebSphere MQ File Transfer Edition V7.0.3. Figure A-1 shows the topology that we used.

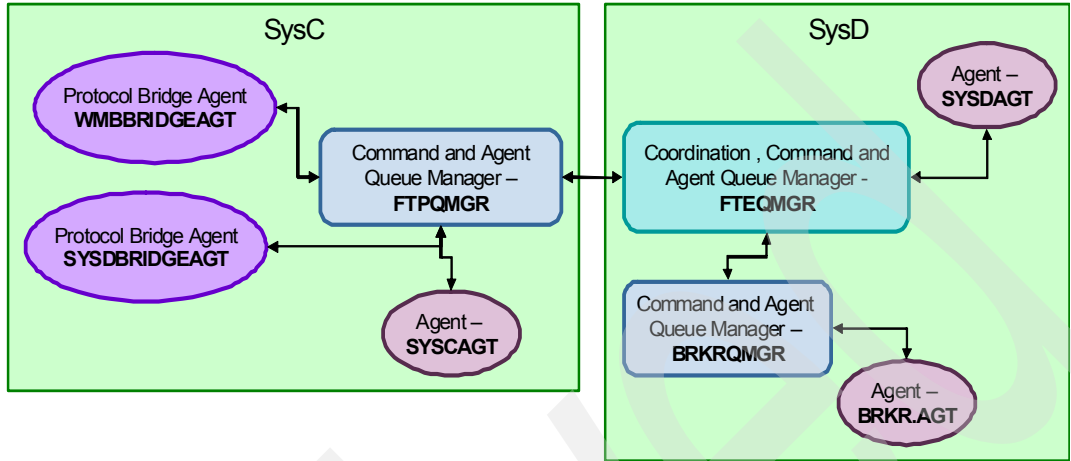


Figure A-1 WebSphere MQ File Transfer Edition Topology

We show how to set up FTPQMGR and FTEQMGR. Refer to 6.3.9, “Configuring WebSphere MQ File Transfer Edition bridge agent” on page 196, for information about setting up SYSDBRIDGEAT and 7.3, “Configuring the solution components” on page 260, for information about setting up WMBBBRIDGEAT, SYSDBRIDGEAT, BRKQMGR and BRKR.AGT.

Our configuration process proceeds as follows:

1. Create queue manager FTPQMGR on SysC and FTEQMGR on SysD.
2. Create WebSphere MQ objects on queue managers FTPQMGR and FTEQMGR.
3. Create a WebSphere MQ File Transfer Edition network using the installation procedures:
 - a. Define FTEQMGR as the coordination, command, and agent queue manager.
 - b. Create WebSphere MQ File Transfer Edition Server agent SYSDAGT on FTPQMGR.
 - c. Define FTPQMGR as the command and agent queue manager.
 - d. Create WebSphere MQ File Transfer Edition Server agent SYSCAGT on FTEQMGR.

Security prerequisites: Review your local security policy and practices to determine what is appropriate for your production environment. While the scenarios in this book do not implement security, it is important that you take security into consideration when implementing these scenarios in your own environment.

Configuring WebSphere MQ

In this section, we create two queue managers:

- ▶ FTPQMGR
- ▶ FTEQMGR

Once that is accomplished, we create WebSphere MQ objects on them.

Before creating queue managers, create OS Local user fteadmin and put it in the mqm group (Figure A-2). We use fteadmin to run the queue managers and WebSphere MQ File Transfer Edition agents.

Security issue: Note that the scenarios we build for this book are done in a lab environment. Part of the planning for a production file transfer environment includes defining the accounts and groups to be used and determining the appropriate level of authority that each should have.

By putting fteadmin in the mqm group, we have given the agents full administrative authority over WebSphere MQ. This allowed us to focus on the file transfer aspects of the configuration but *should never be done in a production environment*.

Your user ID scheme should incorporate isolation between WebSphere MQ and the file transfer agents by using an account in an ordinary group to run the file transfer agent, and then authorizing the agent to the appropriate queues using `setmqaut` commands. In the case of client agents, SSL or exits are usually used to authenticate the connection. Be sure to apply your locally approved security controls to limit FTE agents to non-administrative access to WebSphere MQ.

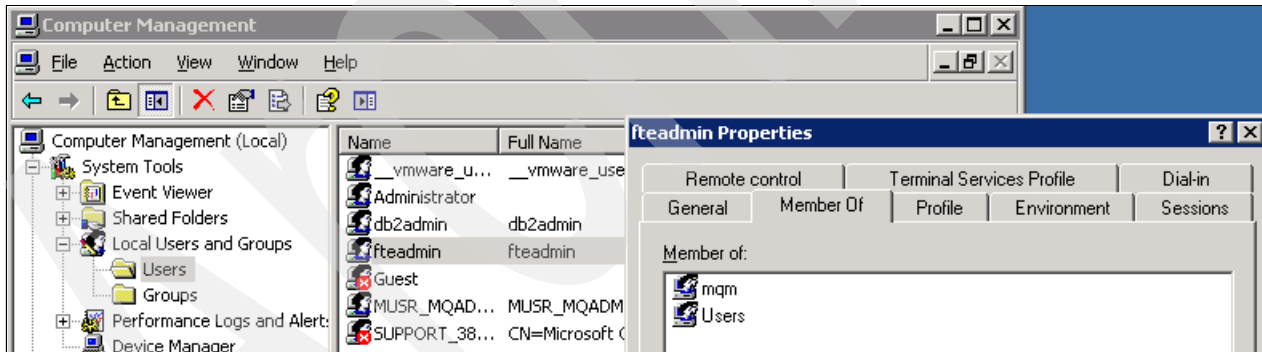


Figure A-2 Create fteadmin user

Creating the queue managers

Table A-1 summarizes the queue managers topology.

Table A-1 Queue managers topology

Queue manager name	System	Port
FTPQMGR	SysC	1414
FTEQMGR	SysD	1414

The queue managers can be created using the WebSphere MQ Explorer (on Windows and Linux) or using the command-line interface (CLI). On both systems, we created the queue managers using WebSphere MQ Explorer based on information in Table A-1 on page 349.

To create a queue manager in the WebSphere MQ Explorer, in the left pane, right-click **Queue Manager** and then click **New** → **Queue manager**. Enter the name of the queue manager as shown in Figure A-3. Click **Finish**. The queue manager is created and we can now define WebSphere MQ objects.

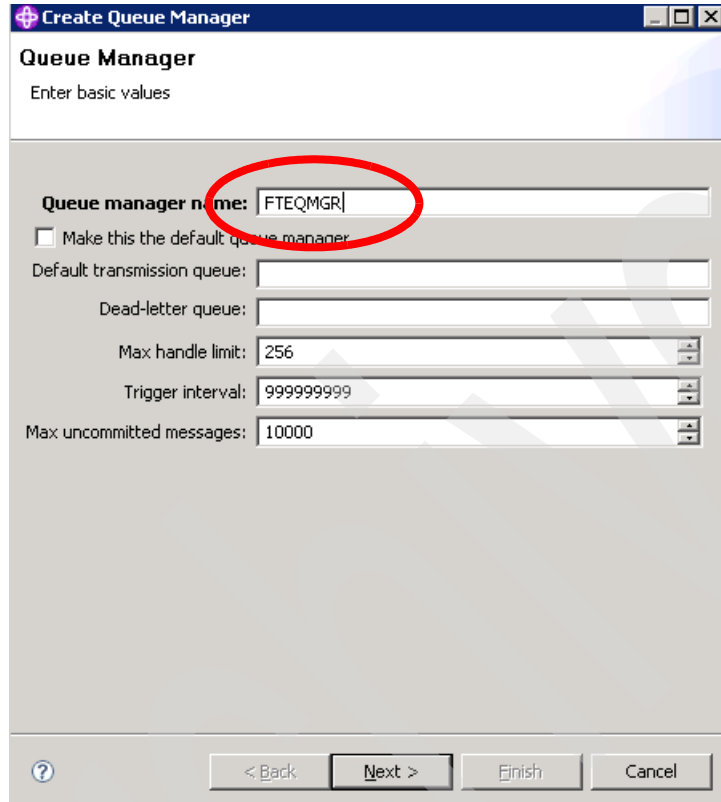


Figure A-3 Create Queue Manger

Creating a queue manager using CLI

We can also create a queue manager using CLI. Example A-1 shows the commands that create the queue manager FTEQMGR and the listener on port 1414, and defines that the listener starts at queue manager startup.

Example A-1 Create queue manager with CLI

```
crtmqm FTEQMGR

strmqm FTEQMGR
echo DEF LISTENER(LISTENER.TCP) TRPTYPE(TCP) PORT(1414) CONTROL(QMGR) | runmqsc
FTEQMGR
echo START LISTENER(LISTENER.TCP) | runmqsc FTEQMGR
```

Creating the queue manager objects

Table A-2 summarizes the WebSphere MQ channel objects on queue manager FTPQMGR.

Table A-2 Channel objects on FTPQMGR (SYSC)

Channel name	Channel type	Connection names	Transmission queue
FTPQMGR.TO.FTEQMGR	SENDER	sysd(1414)	FTEQMGR
FTEQMGR.TO.FTPQMGR	RECEIVER	-	-

Table A-3 summarizes the local queue object on queue manager FTPQMGR.

Table A-3 Local queue objects on FTPQMGR (SYSC)

Queue name	Usage
FTEQMGR	XMITQ

Table A-4 summarizes the channel objects on queue manager FTEQMGR.

Table A-4 Channel objects on FTEQMGR (SYSD)

Channel name	Channel type	Connection names	Transmission queue
FTEQMGR.TO.FTPQMGR	SENDER	sysc(1414)	FTPQMGR
FTPQMGR.TO.FTEQMGR	RECEIVER	-	-

Table A-5 summarizes the local queue object on queue manager FTEQMGR.

Table A-5 Local queue objects on FTEQMGR (SYSD)

Queue name	Usage
FTPQMGR	XMITQ

On both systems, we created these WebSphere MQ objects using WebSphere MQ Explorer based on the information in the above tables.

To create the sender channel for FTPQMGR on SYSC and corresponding receiver channel for FTEQMGR on SYSD, follow these steps:

1. In WebSphere MQ Explorer, in the left pane, right-click **Channels**, then click **New** → **Sender channel**. Enter the name of the Sender channel as shown in Figure A-4.

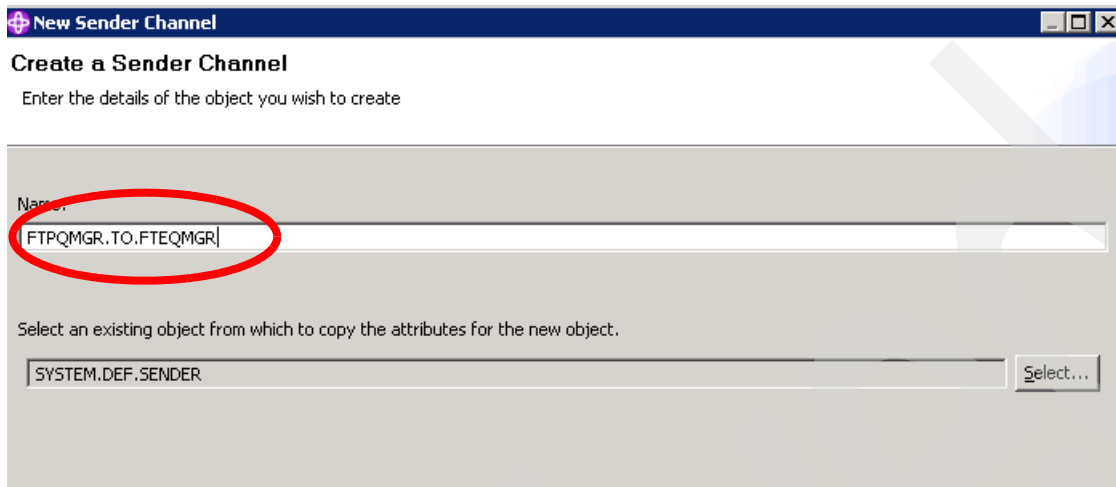


Figure A-4 Create a Sender Channel

Click **Next**.

2. On the Change properties panel, enter the connection name and transmission queue name as shown in Figure A-5.

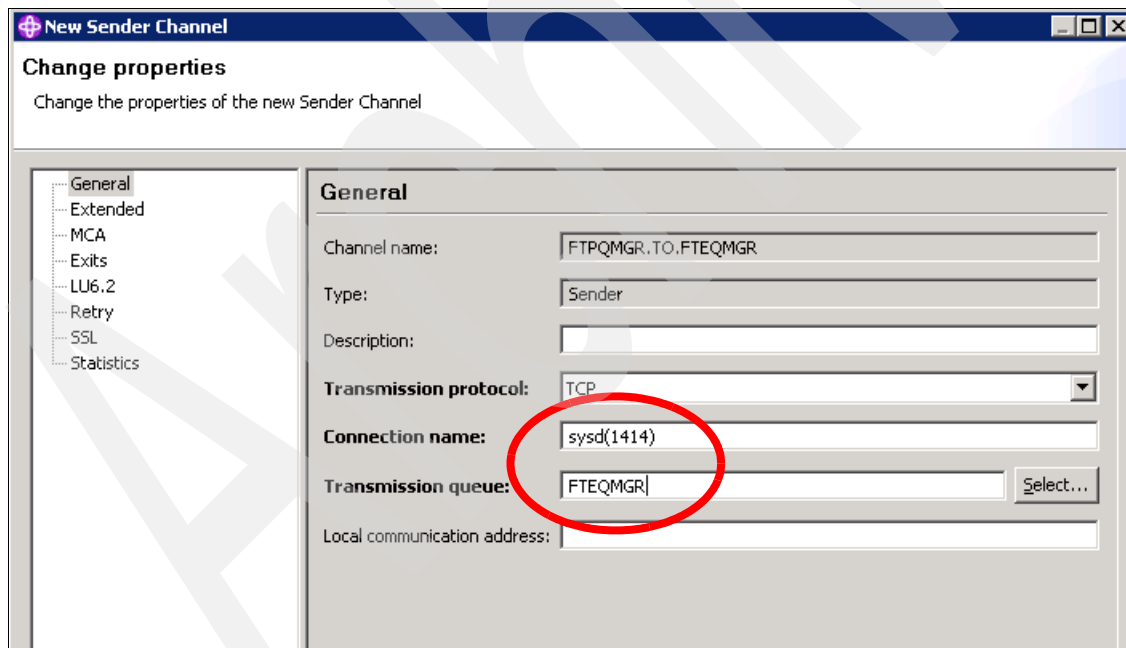


Figure A-5 Change Sender Channel properties

3. Click **Finish**. The sender channel is created.
4. The next step is to create the corresponding receiver channel on SYSD. Note that the name matches the sender channel on SYSC.

In the WebSphere MQ Explorer in the left pane, expand **FTEQMGR**, right-click **Channels**, and then click **New** → **Receiver channel**. Enter the name of the receiver channel (Figure A-6).

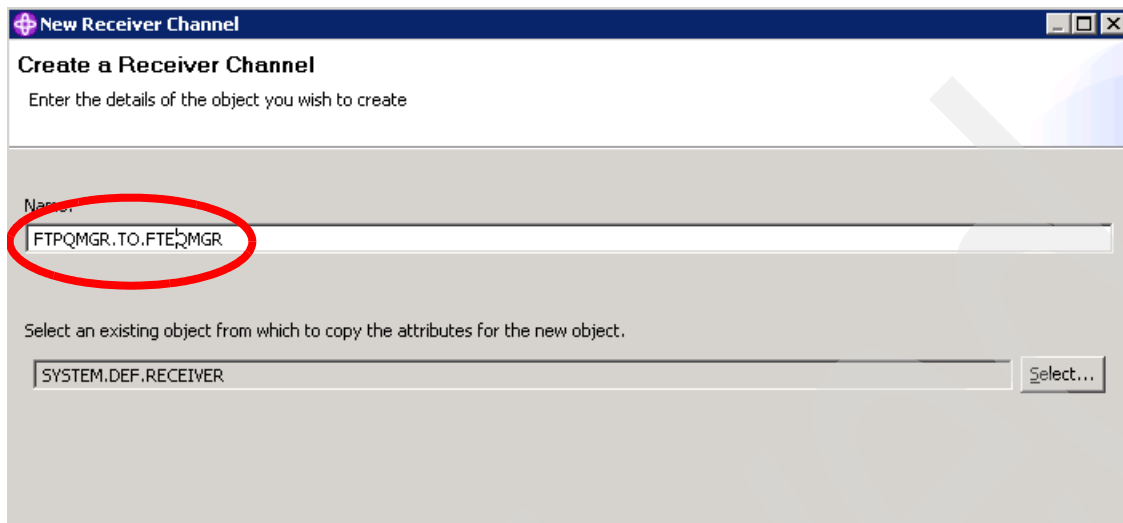


Figure A-6 Create a Receiver Channel

5. Click **Finish**. The receiver channel is created.
6. Repeat this set of instructions to create the sender channel on FTEQMGR for SYSD and the corresponding receiver channel for FTPQMGR on SYSC.

To create the local queues listed in Table A-3 on page 351 and Table A-5 on page 351, follow these steps for each queue:

1. In WebSphere MQ Explorer, in the left pane, right-click **Queue** and then click **New** → **Local Queue**. Enter the name of the local queue name (Figure A-7).

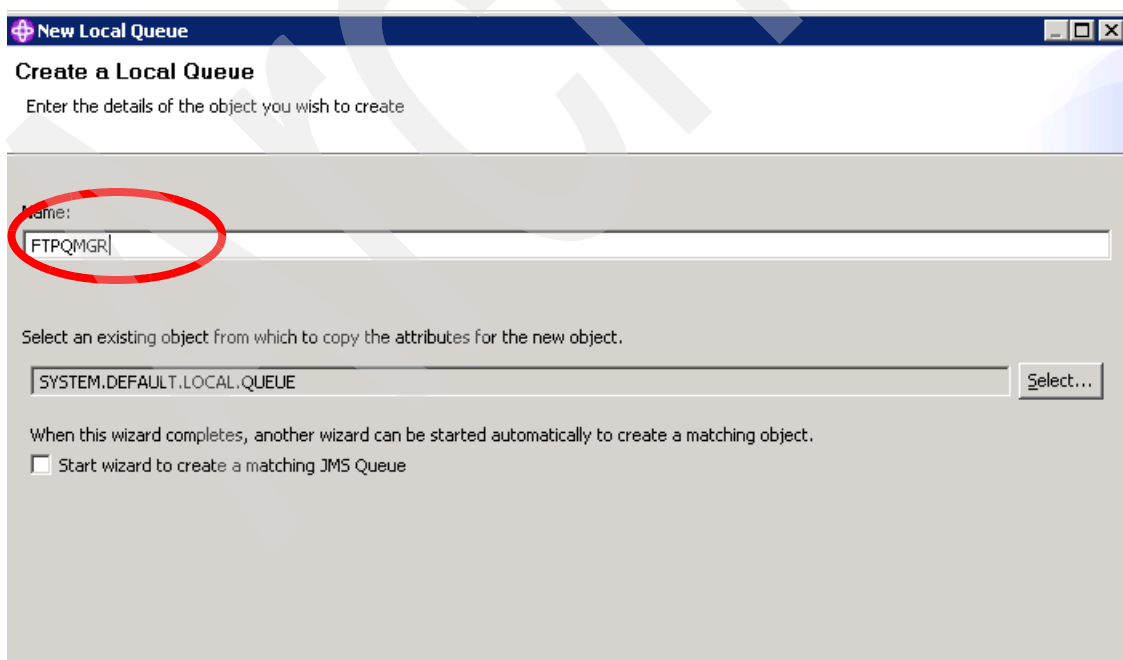


Figure A-7 Create a local queue

- On the Change properties panel, select **Transmission** from the Usage drop-down menu and then click **Finish** (Figure A-8). The local queue is created.

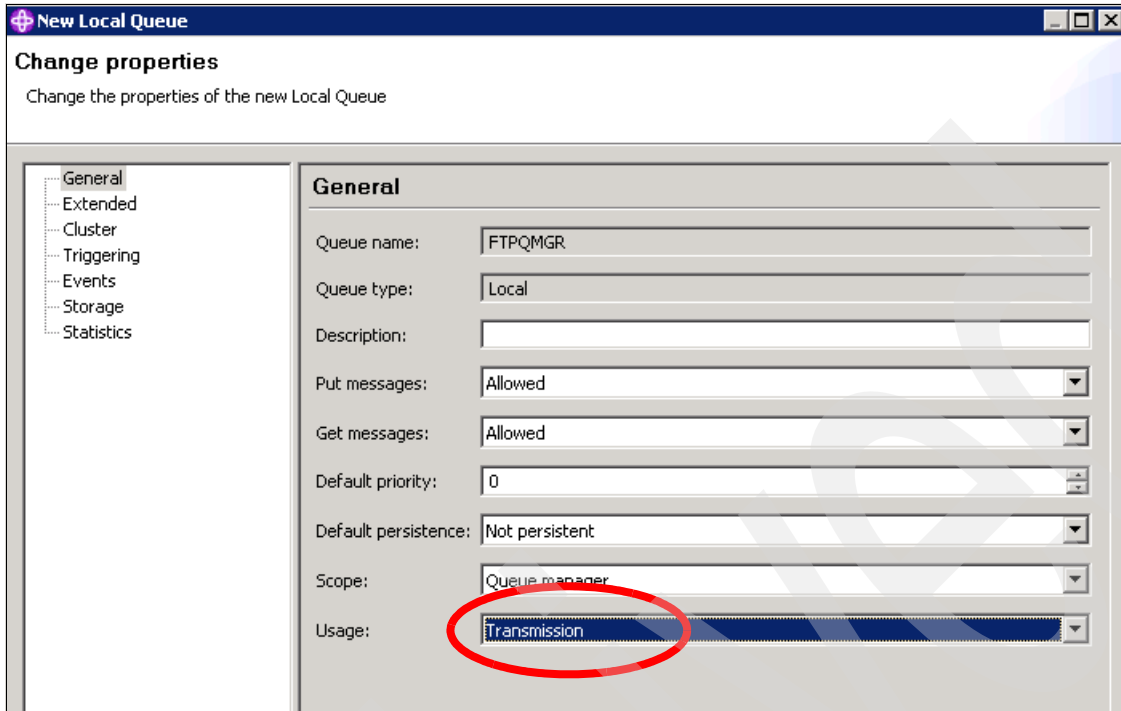


Figure A-8 Change local queue

To start both of the channels:

- In WebSphere MQ Explorer, in the left pane, left click **Channels**, then in the right pane, right-click the sender channel, and then click **Start** (Figure A-9).

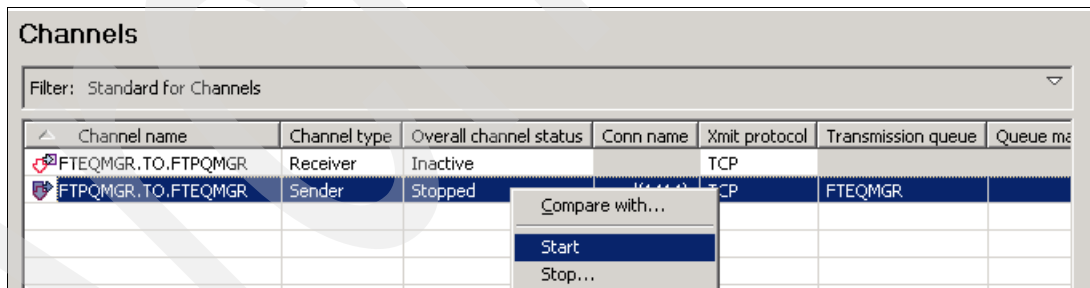


Figure A-9 Start channel

- You can ensure that the channel is working correctly by checking WebSphere MQ Explorer. In the left pane, click **Channels**. The channel status is shown in the right pane (Figure A-10).

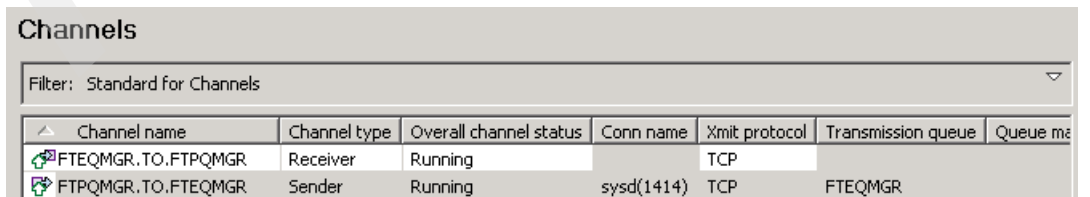


Figure A-10 Channel status

Creating the WebSphere MQ objects using MQSC commands

We can also create the WebSphere MQ objects using MQSC commands. Example A-2 is a sample mqsc script that defines the sender channel, receiver channel, and local queue on FTPQMGR.

Example A-2 MQSC script to create WebSphere MQ objects on FTPQMGR

```
*-----
* Sender Channel
*-----
define channel(FTPQMGR.TO.FTEQMGR) +
chltype(sender) +
conname('sysd(1414)') +
xmitq(FTEQMGR) +
replace

*-----
* Sender Channel
*-----
define channel(FTEQMGR.TO.FTPQMGR) +
chltype(receiver) +
replace

*-----
* Local Queue
*-----
define qlocal(FTEQMGR) +
usage(XMITQ) +
replace
```

Example A-3 is a sample mqsc script that defines the sender channel, receiver channel, and local queue on FTEQMGR.

Example A-3 MQSC script to create WebSphere MQ objects on FTEQMGR

```
*-----
* Sender Channel
*-----
define channel(FTEQMGR.TO.FTPQMGR) +
chltype(sender) +
conname('sysc(1414)') +
xmitq(FTPQMGR) +
replace

*-----
* Sender Channel
*-----
define channel(FTPQMGR.TO.FTEQMGR) +
chltype(receiver) +
replace

*-----
* Local Queue
*-----
define qlocal(FTPQMGR) +
```

usage(XMITQ) +
replace

Configuring WebSphere MQ File Transfer Edition

In this section, we show how to configure the WebSphere MQ File Transfer Edition components required by our scenarios.

Defining the coordination, command, and agent queue manager and WebSphere MQ File Transfer agents

We defined the coordination, command, and agent queue manager and the WebSphere MQ File Transfer Edition agents through the installation procedures of the WebSphere MQ File Transfer Edition Server package.

Table A-6 summarizes the WebSphere MQ File Transfer Edition Topology.

Table A-6 WebSphere MQ File Transfer Edition Topology

Agent name	System	Coordination queue manager to connect to (transport mode)	Command queue manager to connect to (transport mode)	Agent queue manager to connect to (transport mode)
SYSCAGT	SysC	FTEQMGR(client)	FTPQMGR(bindings)	FTPQMGR(bindings)
SYSDAGT	SysD	FTEQMGR(binding)	FTEQMGR(bindings)	FTEQMGR(bindings)

We show only the configuration steps without showing the detailed steps in the installation procedure.

Additional information: For more information about the installation of the WebSphere MQ File Transfer Edition Server, see the WebSphere MQ File Transfer Edition Information Center at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.doc/install_win_unix.htm

Configuring the queue managers during installation

If you have not installed WebSphere MQ File Transfer Edition, you can configure the queue managers during the installation. If you have already installed the product, use the commands in “Configuring the queue managers using commands” on page 361 instead.

To configure the coordination, command, and agent queue manager on both systems, follow these steps:

1. Start the installer, then select the language. Click **Select**.
2. Click **Next** on the Introduction panel.
3. Accept the license agreement and click **Next**.
4. Enter the installation directory, and then click **Next**.
5. Enter the configuration folder, and then click **Next**.

6. On the coordination queue manager configuration panel, enter the coordination queue manager name, then follow these steps:
 - a. For SysC, select **Client** transport mode, then click **Next** (Figure A-11). On the next pane, enter the hostname, port number, and channel name of FTEQMGR (Figure A-12). Click **Next**.

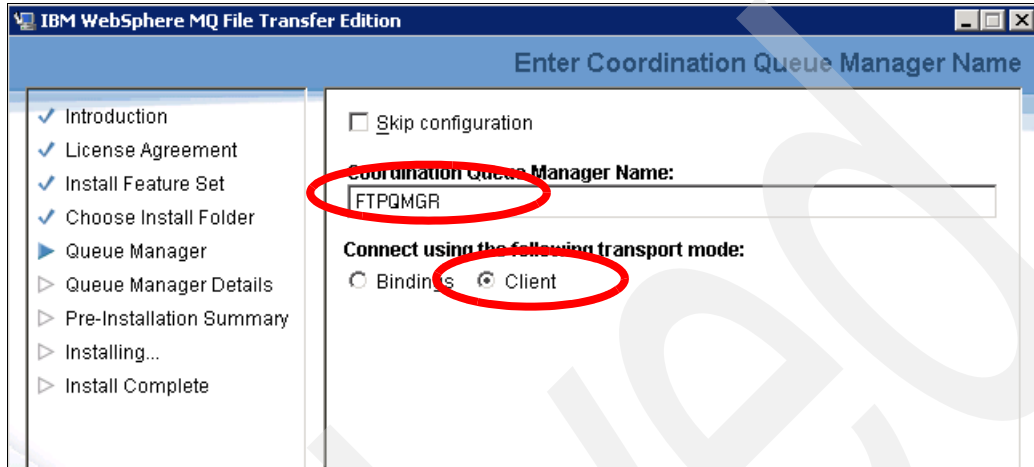


Figure A-11 Enter the coordination queue manager name

- b. On the next pane, enter the hostname, port number, and channel name of FTEQMGR (Figure A-12). Click **Next**.

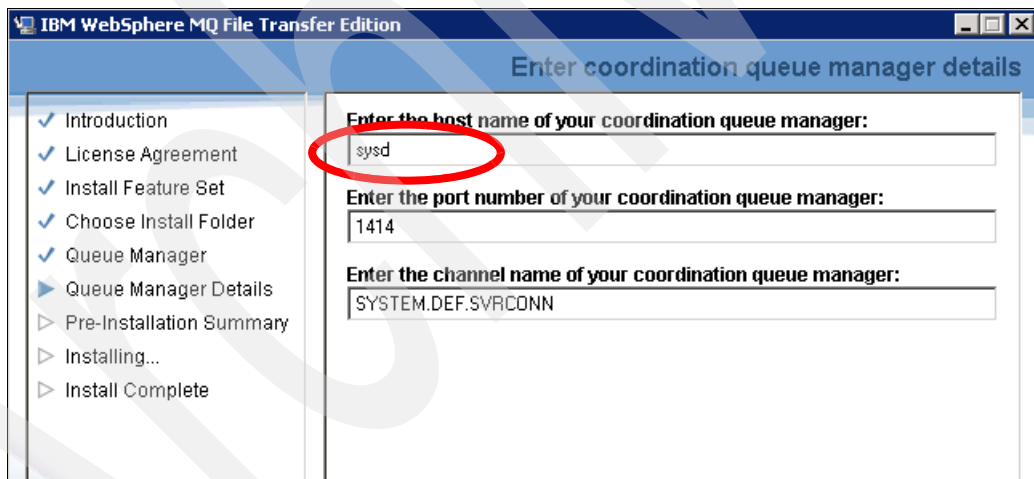


Figure A-12 Enter the coordination queue manager details

- c. For SySD, select **Bindings** for the transport mode (Figure A-13).

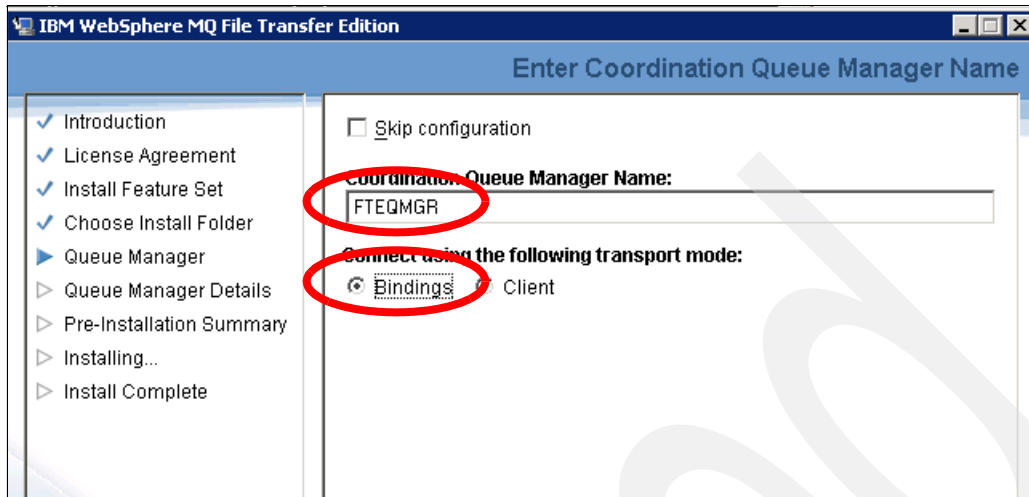


Figure A-13 Enter the coordination queue manager name

Click **Next**.

7. On the command queue manager configuration panel, enter the command queue manager name (Figure A-14).

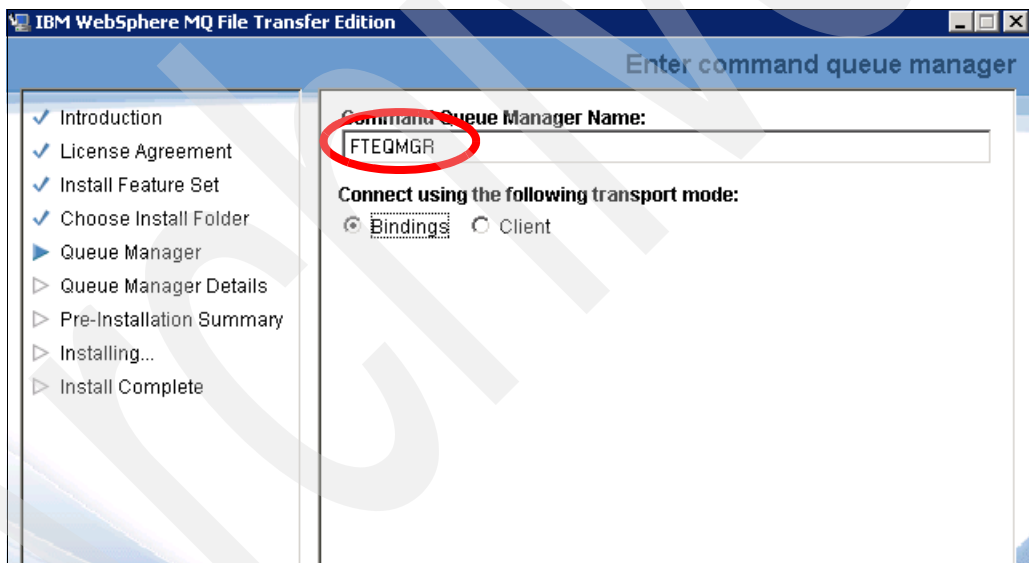


Figure A-14 Enter command queue manger name

Click **Next**.

- On the agent queue manager configuration panel, enter the agent name and agent queue manager name (Figure A-15).

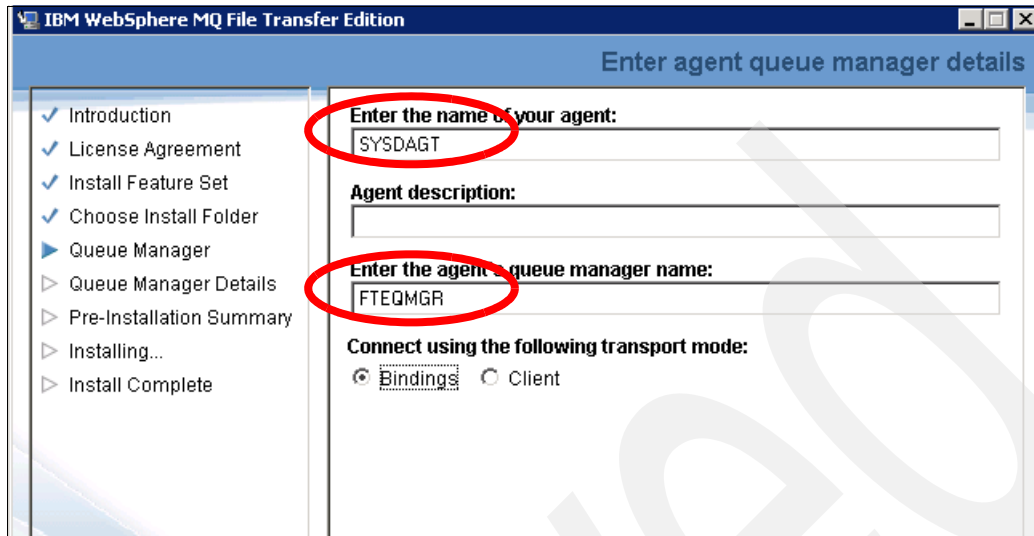


Figure A-15 Enter agent queue manager details

- Click **Next** on the next Pre-Installation Summary panel. Click **Install**.
- During the installation, the path for a file containing MQSC commands will be displayed. You must run these commands against the coordination queue manager (Figure A-16). Make a note of the location of the file and run these commands.



Figure A-16 A MQSC file path for Coordination queue manager

The file provides the mqsc scripts that create the queues, topics, and namelists for the coordination queue manager. You must run these scripts against the coordination queue manager from the command line using the command shown in Example A-4.

Example A-4 Run the MQSC script for the coordination queue manager

```
runmqsc FTEQMGR < C:\Documents and Settings\All Users\Application
Data\IBM\WMQFTE\config\FTEQMGR\FTEQMGR.mqsc
```

- After running the command, click **Next**.

12. On the next panel is the path for the file containing MQSC commands, which you must run against the agent queue manager, which will be displayed as shown in Figure A-17. Make a note of the location of the file and run these commands.

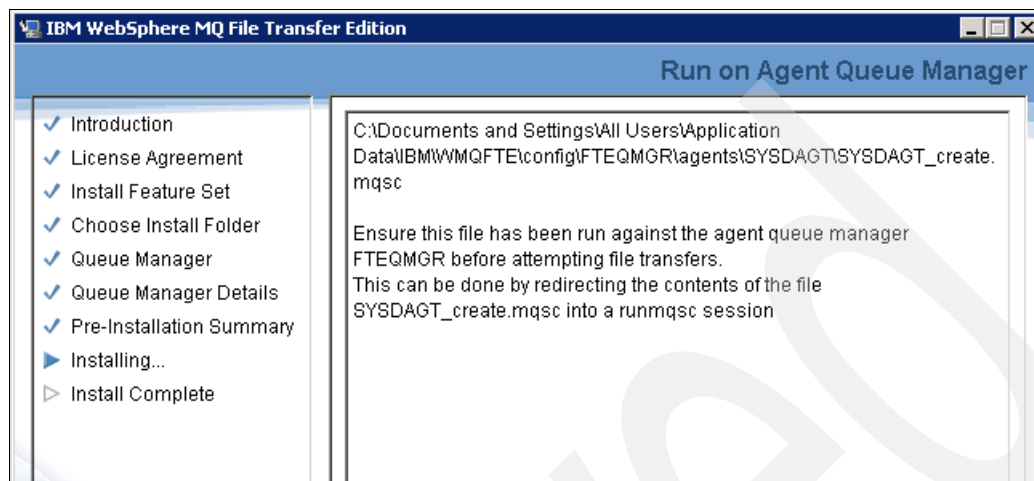


Figure A-17 A MQSC file path for Agent queue manager

The file provides the mqsc scripts that create the queues for the agent queue manager. You must run these scripts against the agent queue manager from the command line using the command shown in Example A-5.

Example A-5 Run a MQSC script for the agent queue manager

```
runmqsc FTEQMGR < C:\Documents and Settings\All Users\Application  
Data\IBM\WMQFTE\config\FTEQMGR\agents\SYSCAGT\SYSCAGT_create.mqsc
```

13. Click **Next** to continue. On the Install complete panel, click **Done**.

The coordination, command, and agent queue manager and WebSphere MQ File Transfer agents are defined.

To make sure that the coordination queue manager is successfully defined, check that the following files are created.

- The config directory (C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config\FTEQMGR)
- The wmqfte.properties file
- The coordination.properties file
- The FTEQMGR.mqsc file

14. To make sure that the command queue manager is successfully defined, check that the command.properties file is created in the coordination queue manager config directory.

To make sure that the agent queue manager is successfully defined, check the following:

- The following files are created in the coordination queue manager config directory:
 - The agent.properties file
 - The SYSCAGT_create.mqsc file
 - The SYSCAGT_delete.mqsc file
 - The UserSandboxes.xml file
 - The exits directory
 - The logs directory

- The agent is successfully registered.

Check the **fteListAgents** command (Example A-6). If the agent is successfully registered, the agent name is output.

Example A-6 Run fteListAgent command

```
>fteListAgents
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED
Agent Name:                               Queue Manager Name:   Status:
SYSCAGT                                   FTPQMGR               STOPPED
SYSDAGT                                   FTEQMGR               STOPPED
```

- We can also check the agent status through WebSphere MQ Explorer. In the left pane, left-click **Agents**. If the agent is successfully registered, the agent name is shown in the Agent status view (Figure A-18).

Agents				
Name	Description	Status ▲	Source transfer	Destination transfer
SYSDAGT		Stopped	0	0
SYSCAGT		Stopped	0	0

Figure A-18 Agents status view of WebSphere MQ File Explorer

If you see a message that the agent was configured but could not be registered, this means that the coordination queue manager could not be contacted because it is not available or your configuration parameters are not correct.

The effect is that the agent can be started and transfer files, but it is not listed by the **fteListAgents** command or in the WebSphere MQ File Transfer Edition Explorer. This means that you cannot define a transfer request in the WebSphere MQ File Transfer Edition Explorer using this agent, and furthermore, status messages of this agent are not shown in the WebSphere MQ File Transfer Edition Explorer Transfer Log view.

Configuring the queue managers using commands

We can also define FTEQMGR and FTPQMGR as the coordination queue manager and command queue manager using WebSphere MQ File Transfer Edition command.

To create the coordination, command, and agent queue manager on the both system, follow these steps:

1. Open a command window and run the commands shown in Example A-7 and Example A-8 from the command line.

Example A-7 shows the command to set up the coordination queue manager on SysC.

Example A-7 Run fteSetupCoordination command on SysC

```
ffteSetupCoordination -coordinationQMgr FTPQMGR-coordinationQMgrHost sysd
-coordinationQMgrPort 1414
```

Example A-8 shows the command to set up the coordination queue manager on SysD.

Example A-8 Run fteSetupCoordination command on SysD

```
fteSetupCoordination -coordinationQMgr FTEQMGR
```

2. Run FTEQMGR.mqsc scripts against the queue manager FTEQMGR from the command line using the command shown in Example A-9.

Example A-9 Run a MQSC script for the coordination queue manager

```
runmqsc FTEQMGR < C:\Documents and Settings\All Users\Application  
Data\IBM\WMQFTE\config\FTEQMGR\FTEQMGR.mqsc
```

3. Run the command shown in Example A-10.

Example A-10 Run fteSetupCommands command

```
fteSetupCommands -connectionQMGr FTEQMGR
```

This command creates the command.properties file in the coordination queue manager config directory (for example, C:\Documents and Settings\All Users\Application Data\IBM\WMQFTE\config\FTEQMGR).

4. Run the command shown in Example A-11

Example A-11 Run fteCreateAgent command to create FTEQMGR agent

```
fteCreateAgent -agentname SYSDAGT -agentQMGr FTEQMGR
```

5. At a command prompt, enter the command shown in Example A-12.

Example A-12 Run a MQSC script for the agent queue manager

```
>runmqsc FTEQMGR <  
C:\IBM\WMQFTE\config\FTEQMGR\agents\SYSDAGT\SYSDAGT_create.mqsc
```

6. Now we start the SYSDAGT agent using the **fteStartAgent** command (Example A-13).

Example A-13 fteStartAgent command

```
ftestartagent SYSDAGT
```

Starting the agent

Now start the SYSCAGT agent on SysC and the SYSDAGT agent on SysD using the **fteStartAgent** command (Example A-14).

Example A-14 Starting the agent

```
ftestartagent SYSCAGT
```

You can check the agent's log file to see whether the agent started successfully.

- ▶ Check the **fteListAgents** command (Example A-6 on page 361). If the agent is successfully started, the status is READY.

Example A-15 Run fteListAgent command

```
>fteListAgents  
5655-U80, 5724-R10 Copyright IBM Corp. 2008, 2010. ALL RIGHTS RESERVED  
Agent Name: Queue Manager Name: Status:  
SYSCAGT FTPQMGR READY  
SYSDAGT FTEQMGR READY
```

- ▶ We can also check the agent status through WebSphere MQ Explorer. On the left pane, click **Agents**. If the agent is successfully started, the status is *Ready* (Figure A-19).

Agents				
Name	Description	Status ▲	Source transfer	Destination transfer
SYSCAGT		Ready	0	0
SYSDAGT		Ready	0	0

Figure A-19 Agents status view of WebSphere MQ File Explorer

Archived

Building the WebSphere Message Broker flow

In Chapter 7, “External transfers using IBM WebSphere Message Broker and IBM Sterling File Gateway” on page 245, we use a WebSphere Message Broker message flow to mediate an input file and produce an output file. The WebSphere Message Broker flow that we use in this book is based on the Simplified Database Routing sample found in the WebSphere Message Broker Information Center. The sample is a message flow application that is based on the scenario of an employee management processing system. It demonstrates how you can use a range of simplified message flows that require no programming, how to access databases using JDBC, and how to use values in an acquired result set from a database query to either route messages dynamically or to update their content.

The Simplified Database Routing sample is available in the WebSphere Message Broker Information Center:

<http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.samples.simplifieddbrouting.doc/doc/overview.htm>

We modified the sample flow to use WebSphere MQ File Transfer Edition by using the FTEInput and FTEOutput nodes in WebSphere Message Broker. This appendix provides information about how to modify the sample to work with our file transfer scenario.

The Simplified Database Routing sample uses an MQInput node to trigger the message flow. In our modified flow, we use an FTEInput node to trigger the flow. Additionally, the Simplified Database Routing sample uses MQOutput nodes to put messages on a queue based on the processing. We also use MQOutput and FTEOutput nodes based on the mediation results in our flow based upon the sample.

You can find instructions for building the sample at:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.samples.simplifieddbrouting.doc/doc/create_flow_simplifiedDBRouting.htm

Overview of the flow modifications

Figure B-1 shows the flow for the Simplified Database Routing sample.

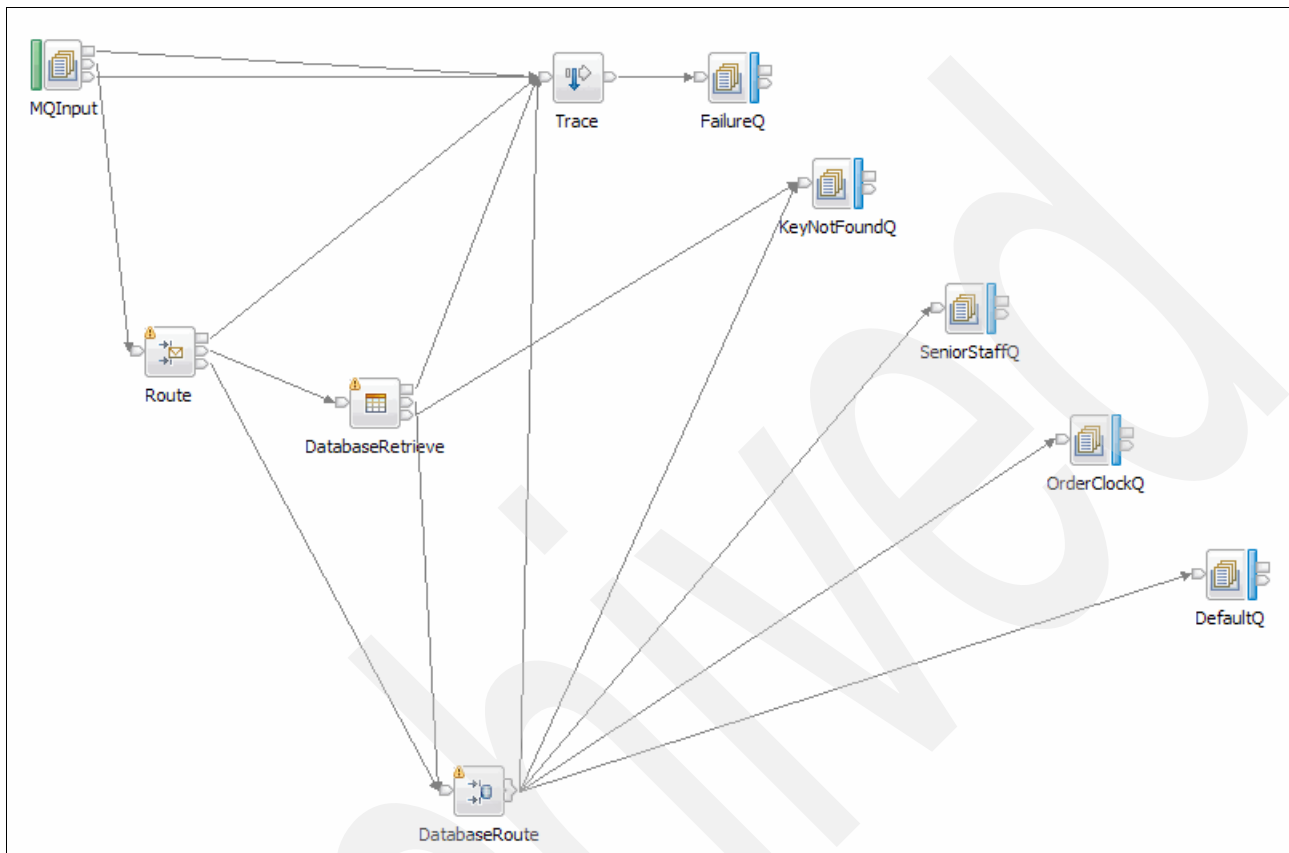


Figure B-1 Message flow for the Simplified Database Routing sample

Figure B-2 shows the flow after you make modifications to use managed file transfer.

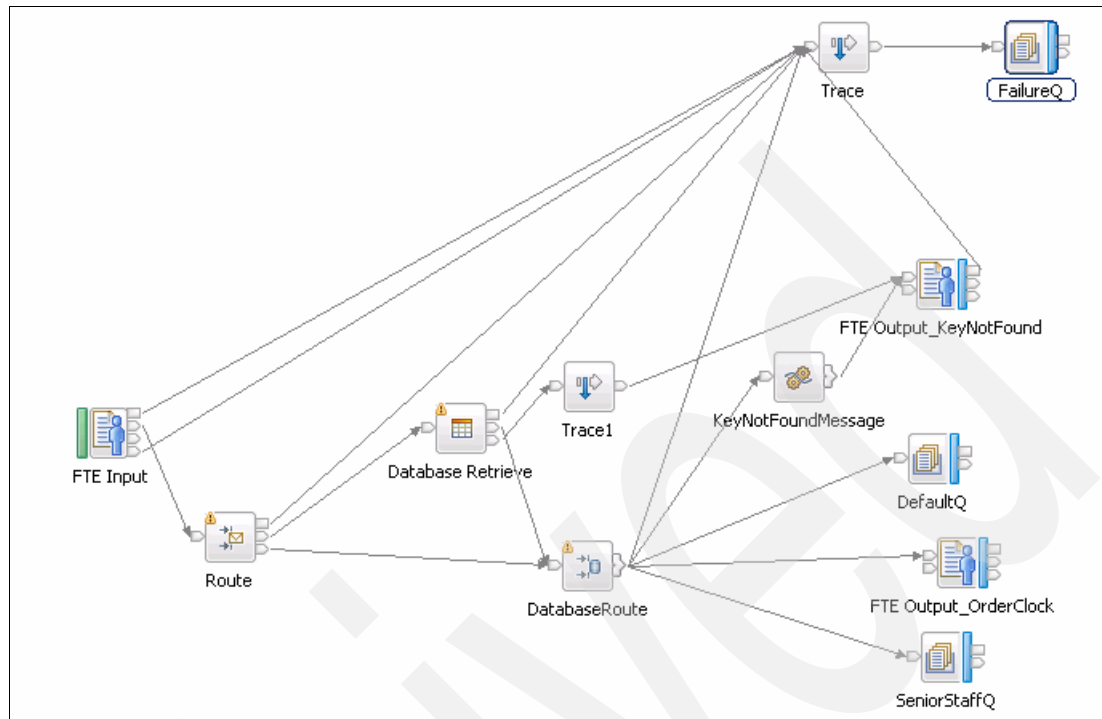


Figure B-2 Message flow that we use in this book

Modifying the sample

This section provides details about the modifications that you need to make to the Simplified Database Routing sample so that it works with the file transfer scenario in this book.

Nodes

Use the nodes as listed in Table B-1 when you build the sample.

Table B-1 WebSphere Message Broker nodes

Palette drawers	Node type	Node name
File	FTEInput	FTE Input
File	FTEOutput	FTE Output_KeyNotFound
File	FTEOutput	FTE Output_OrderClock
WebSphereMQ	MQOutput	FailureQ (MQ Queue)
WebSphereMQ	MQOutput	SeniorStaffQ (MQ Queue)
WebSphereMQ	MQOutput	DefaultQ (MQ Queue)
Routing	Route	Route
Database	DatabaseRetrieve	DatabseRetrieve

Palette drawers	Node type	Node name
Database	DatabaseRoute	DatabaseRoute
Construction	Trace	Trace
Transformation	Compute	KeyNotFoundMessage

Connect the nodes as shown in Table B-2 when building the sample.

Table B-2 Node connection information

Node name	Terminal	Connect to this node
FTEInput	Out	Route
	Failure	Trace
	Catch	Trace
Route	Default	DatabaseRetrieve
	Failure	Trace
	Match	DatabaseRoute
DatabaseRetrieve	Out	DatabaseRoute
	Failure	Trace
	KeyNotFound	KeyNotFoundMessage
DatabaseRoute	KeyNotFound	KeyNotFoundMessage
	Failure	Trace
	TenYearsService	FTE Output_OrderClock
	OlderThanMe	SeniorStaffQ
	Default	DefaultQ
Trace	Out	FailureQ
KeyNotFoundMessage	Out	FTE Output_KeyNotFound

Properties

Use the information in Table B-3 to set the properties on only the nodes that are used in the scenario in this book.

Table B-3 Node properties

Node name	Page	Property	Value
FTEInput	Basic	Directory filter	C:\FileTransfersInbound
	Basic	File name filter	*.xml
	Basic	Action on successful processing	Delete
	Input Message Processing	Message domain	XMLNSC For XML messages (namespace aware, validation, low memory use)

Node name	Page	Property	Value
FTE Output_KeyNotFound	Basic	Job name	InvalidWorkDepartment
	Basic	Agent	WMBBRIDGEAGT
	Basic	Queue manager	FTPQMGR
	Basic	File directory	/SysD_WMB_Partner/To_MyFG_Partner
	Basic	File name	KeyNotFound.txt
	Basic	Mode	Text transfer (ASCII/EBCDIC and CD/LF automated)
	Basic	Overwrite files on destination system	check
FTE Output_OrderClock	Basic	Job name	OrderClock
	Basic	Agent	WMBBRIDGEAGT
	Basic	Queue manager	FTPQMGR
	Basic	File directory	/SysD_WMB_Partner/To_SFTP_Partner
	Basic	File name	ClockOrder.xml
	Basic	Mode	Text transfer (ASCII/EBCDIC and CR/LF automated)
	Basic	Overwrite files on destination system	check
Compute	Basic	ESQL module	InboundFileTransferFlow_ErrorMessage
Route	Basic	Filter pattern	\$Body/EmpRecord/WorkDept This property is an XPath 1.0 Expression, specifying in this case a path location to an expected element within the input message to this node. The node attempts to find a work department filed and if not present resolves to false.
	Basic	Routing output terminal	Match This property is the name of a dynamic output terminal to which to propagate the input message if the filter expression resolves to true. The expression resolves to true if the expected element is located within the input message. Note: This terminal must first be created by right-clicking the node and clicking Add Output Terminal .

Node name	Page	Property	Value
DatabaseRetrieve	Basic	Data source name	SIMPLEROUTEDB The alias used to locate the JDBC Provider details that are stored in the broker registry. The alias is used to locate and build the JDBC connection URL that is used to connect to a database.
	Basic	Copy message	Yes This property indicates that a copy of the original incoming message is required because the message tree is updated.
DatabaseRetrieve	Basic	Query elements	See Table B-4 on page 372 for the information to configure the query elements. When all of the query elements are entered, the DatabaseRetrieve node SQL statement window has a SQL statement (Example B-1 on page 372).
	Data Element Table	Data elements	Use the information in Table B-5 on page 373 to configure the data elements. Each row in Table B-5 on page 373 specifies a location in the output message where a retrieved column value is inserted before propagation of the message from this node. If the location does not exist in the output message, it is created. The default value for basic property, Multiple rows, is set to no, causing only the values in the first row of a result set to process by the node. The result set is obtained by running the query specified in the SQL statement window and configured by using the query elements table content.

Node name	Page	Property	Value
DatabaseRoute	Basic	Datasource	SIMPLEROUTEDB The alias that is used to locate JDBC provider details that are stored in the broker registry. The alias is used to locate and build the JDBC connection URL that is used to connect to a database.
	Basic	Query elements	Use the information provided in Table B-6 on page 373 to configure the query elements for the DatabaseRoute node. After the elements are selected, a SQL statement is generated in the SQL statement window of the DatabaseRoute node (Example B-2 on page 373).
	Basic	Distribution mode	All This property determines the routing behavior of this node when an inbound message matches multiple expressions. When the Distribution mode is set to All, the message is propagated to all matching output terminals. If there is no matching output terminal, the message is sent to the default terminal.
DatabaseRoute	Filter Expression Table	Filter table	Each row in Table B-7 on page 373 specifies an XPath 1.0 expression where retrieved column values are represented in the form of variable references. Each expression is cast as a boolean. If the expression resolves to be true, then the node propagates the input message to the dynamic output terminal that is specified for the routing output terminal value of the row. This terminal must first be created by right-clicking the node and selecting Add Output Terminal .
Trace	Basic	Destination	Local Error Log This property instructs the node to write trace information to the local error log. On Windows, this is the Event Viewer. On Linux, this is the syslog.
	Basic	Pattern	The trace pattern used to extract the entire message tree is: Root > \${Root} ExceptionList > \${ExceptionList}
	Basic	Message number	3051

Node name	Page	Property	Value
FailureQ	Basic	Queue name	SIMPLERROUTEDB_FAILURE This property is the queue that the message flow puts the message into if the processing fails.
SeniorStaffQ	Basic	Queue name	SIMPLERROUTEDB_SNRSTAFF This property is the queue that the message flow puts the message into.
DefaultQ	Basic	Queue name	SIMPLERROUTEDB_DEFAULT This property is the queue that the message flow puts the message into.

Query elements

Use the information in Table B-4 to configure the DatabaseRetrieve node query elements.

Table B-4 DatabaseRetrieve node query elements

Table name	Column name	Operator	Value type	Value
EMPLOYEE E	LASTNM	ASC	None	None
EMPLOYEE E	FIRSTNM	ASC	None	None
EMPLOYEE E	YEARSSERVICE	ASC	None	None
EMPLOYEE E	AGEINYRS	ASC	None	None
EMPLOYEE E	WORKDEPT	ASC	None	None
EMPLOYEE E	EMPNUM	=	Element	\$InputBody/EmpRecord/ EmpNumber

The query elements that you select in the DatabaseRetrieve node generate the SQL shown in Example B-1. This information displays in the SQL statement window of the DatabaseRetrieve node.

Example B-1 SQL statement generated by selection of query elements

```
SELECT E.LASTNM, E.FIRSTNM, E.YEARSSERVICE, E.AGEINYRS, E.WORKDEPT
FROM EMPLOYEE E
WHERE E.EMPNUM = ?
ORDER BY E.LASTNM ASC, E.FIRSTNM ASC, E.YEARSSERVICE ASC, E.AGEINYRS ASC,
E.WORKDEPT ASC
```

The data elements for the DatabaseRetrieve node specify the location in the output message where a retrieved column from Table B-5 is inserted before propagation of the message from this node.

Table B-5 Data elements for DatabaseRetrieve node

Column name	Message element
E.FIRSTNM	\$OutputRoot/XMLNSC/EmpRecord/FirstName
E.LASTNM	\$OutputRoot/XMLNSC/EmpRecord/LastName
E.YEARSSERVICE	\$OutputRoot/XMLNSC/EmpRecord/YrsInService
E.AGEINYRS	\$OutputRoot/XMLNSC/EmpRecord/AgeInYrs
E.WORKDEPT	\$OutputRoot/XMLNSC/EmpRecord/WorkDept

Table B-6 lists the query elements properties for the DatabaseRoute node.

Table B-6 Query elements for the DatabaseRoute node

Table name	Column name	Operator	Value type	Value
EMPLOYEE M	EMPNUM	ASC	None	None
EMPLOYEE M	LASTNM	ASC	None	None
EMPLOYEE M	AGEINYRS	ASC	None	None
DEPARTMENT D	DEPTNUM	=	Element	\$Body/EmpRecord/WorkDept
DEPARTMENT D	MGRNUM	=	Column	(EMPLOYEE)M.EMPNUM

After you configure the query element properties for the DatabaseRoute node, an SQL statement similar to the one shown Example B-2 displays in the SQL statement window on the Basic tab of the DatabaseRoute node.

Example B-2 DatabaseRoute node SQL statement generated from selection of query elements

```
SELECT M.EMPNUM, M.LASTNM, M.AGEINYRS
FROM EMPLOYEE M, DEPARTMENT D
WHERE D.DEPTNUM = ?
AND D.MGRNUM = M.EMPNUM
ORDER BY M.EMPNUM ASC, M.LASTNM ASC, M.AGEINYRS ASC
```

Filter expressions

Use the information in Table B-7 to configure the Filter Expressions Filter table for the DatabaseRoute node.

Table B-7 Filter expressions for the DatabaseRoute node

Filter pattern	Routing output terminal
(\$Body/EmpRecord/AgeInYears > \$M_AGEINYRS) and (\$M_LASTNM = 'KWAN')	OlderThanMe
\$Body/EmpRecord/YrsInService >= 10	TenYearsService

Creating the ESQL module

To create the ESQL module:

1. Select **File** → **New** → **Message Flow ESQL File** (Figure B-3).

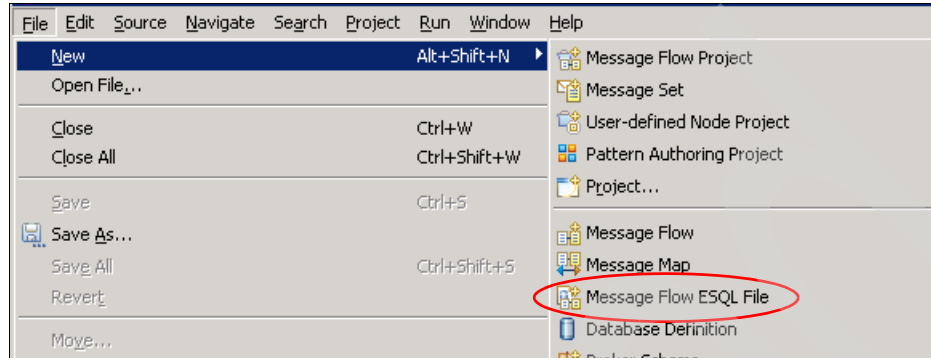


Figure B-3 Create a new ESQL file

2. Select your current working project, **InboundFileTransfer**, and enter an ESQL file name, **InboundFileTransferFlow** (Figure B-4).

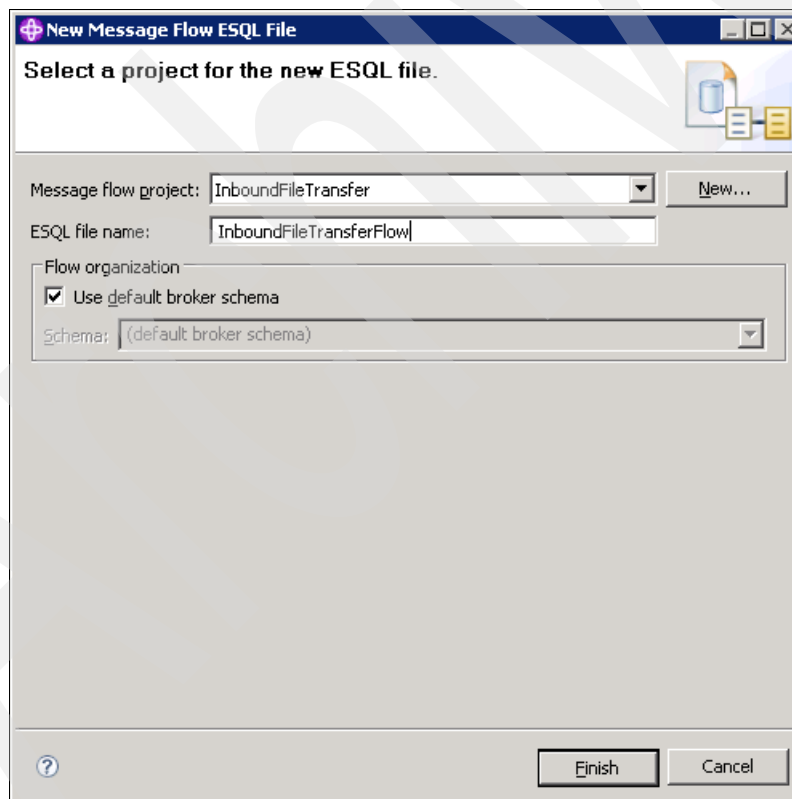


Figure B-4 Create an ESQL file name

3. Double-click the **InboundFileTransferFlow.esql** file in the projects view of the WebSphere Message Broker Toolkit to open the ESQL editor (Figure B-5). Enter the text from Example B-3, which we modified and created for this flow.

Example B-3 ESQL code

```

CREATE COMPUTE MODULE InboundFileTransferFlow_ErrorMessage
  CREATE FUNCTION Main() RETURNS BOOLEAN
  BEGIN
    -- CALL CopyMessageHeaders();
    CALL CopyEntireMessage();
    RETURN TRUE;
  END;

  CREATE PROCEDURE CopyMessageHeaders() BEGIN
    DECLARE I INTEGER 1;
    DECLARE J INTEGER;
    SET J = CARDINALITY(InputRoot.*[]);
    WHILE I < J DO
      SET OutputRoot.*[I] = InputRoot.*[I];
      SET I = I + 1;
    END WHILE;
  END;

  CREATE PROCEDURE CopyEntireMessage() BEGIN
    SET OutputRoot = InputRoot;
    SET OutputRoot.XMLNSC.EmpRecord.ErrorMessage = 'The Employees Work
Department was not found.';
  END;
END MODULE;

```

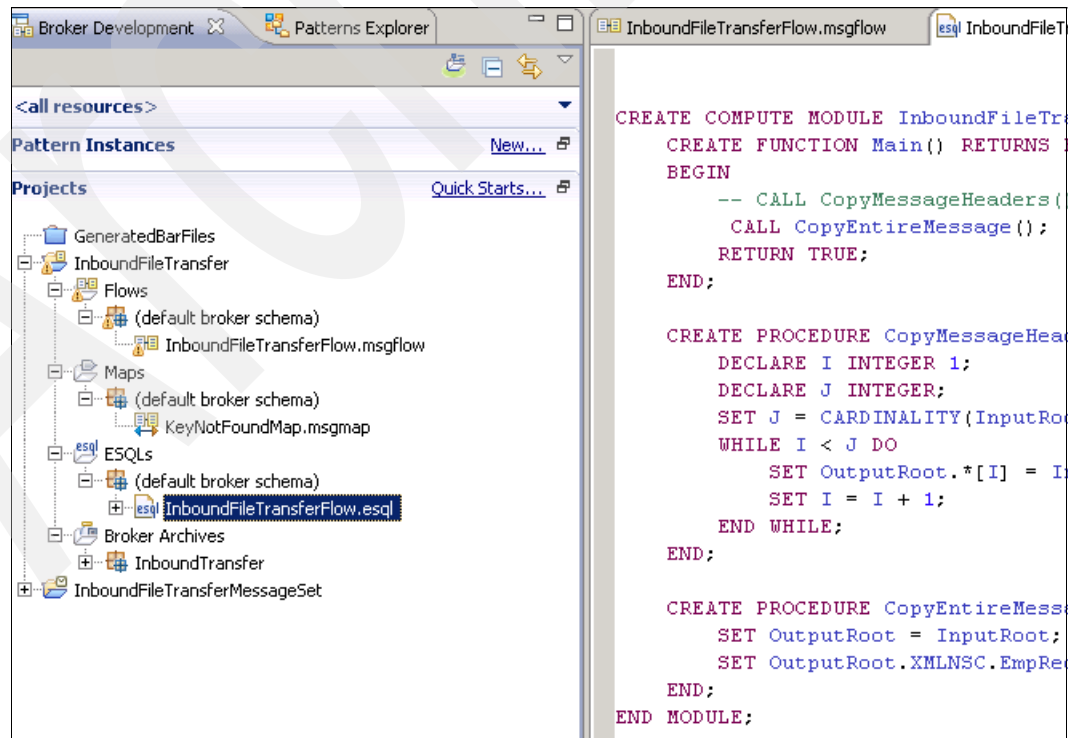


Figure B-5 Edit the ESQL in WebSphere Message Broker Toolkit

4. Save the ESQL file, which creates the module that is imported into the compute node.

Other configuration tasks

To complete the configuration, make the following additional changes:

- ▶ Populate the database and create a JDBC provider entry, as detailed in the “Setting up the Database” page of the Simplified Database Routing sample.
- ▶ Create the MQ queues that correspond to the MQOutput nodes. Although these queues are not used in the file transfer scenario, it can be beneficial to have these queues as you test the message flow.

Example files to test the flow

Test the flow using the following example files. These files are taken from the sample, put into a text document, and saved as file type .xml.

The RetrieveData.xml file (Example B-4) contains an employee number. When the information that is associated with that employee number is retrieved and sent to the DatabaseRoute node, the conditions that are set up in the node determine that the employee has 10 years of service. An order for an anniversary clock is sent to Sterling File Gateway using the FTEOutput_OrderClock node.

Example B-4 RetrieveData.xml contents

```
<EmpRecord><EmpNumber>000010</EmpNumber></EmpRecord>
```

The UseRouteNode-KeyNotFound.xml file (Example B-5) contains information about an employee with an invalid work department. Because the work department information is included in the input file, the contents are sent automatically to the DatabaseRoute node. The DatabaseRoute node can compare and find that the work department sent is not found in the listing of work departments in the database. The contents of the input file are then sent to the compute node to add an error message to it. The contents are sent to myFileGateway using the FTE Output_KeyNotFound node for the initial sender to review and correct.

Example B-5 UseRouteNode-KeyNotFound.xml contents

```
<EmpRecord>
  <EmpNumber>000010</EmpNumber>
  <FirstName>DAVID</FirstName>
  <LastName>BROWN</LastName>
  <YrsInService>10</YrsInService>
  <AgeInYears>54</AgeInYears>
  <WorkDept>E00</WorkDept>
</EmpRecord>
```

There are two other sample files that can be sent through the flow. These samples result in messages being placed on a WebSphere MQ queue. They do not fully use file transfer, and we do not highlight them in this scenario. However, these files can be useful in testing the broker flow.

The RetrieveData-NoMatch.xml file (Example B-6 on page 377) contains information about a valid employee, but that data does not match the condition that the employee has 10 years of

service. The employee is also the manager of the department and determines senior staff members. Because the employee data is compared to itself, it cannot be older or younger. Because none of the filtering expressions are satisfied, a message is placed in the default queue.

Example B-6 Retrieve Data-NoMatch.xml contents

```
<EmpRecord>
  <EmpNumber>00020</EmpNumber>
  <FirstName>SALLY</FirstName>
  <LastName>KWAN</LastName>
  <YrsInService>9</YrsInService>
  <AgeInYears>27</AgeInYears>
  <WorkDept>D00</WorkDept>
</EmpRecord>
```

The UseRouteNode.xml file (Example B-7) contains information about an employee who is older than his manager. The contents of this file are never sent to the DatabaseRetrieve node. After the DatabaseRoute node filtering expressions evaluates the input message, a message is placed in the SeniorStaff queue. Additionally, this employee also meets the criteria for a anniversary clock. This information results in a second output being generated to send the employee information to Sterling File Gateway using WebSphere MQ File Transfer Edition through the FTE Output_OrderClock node.

Example B-7 UseRouteNode.xml contents

```
<EmpRecord>
  <EmpNumber>000010</EmpNumber>
  <FirstName>DAVID</FirstName>
  <LastName>BROWN</LastName>
  <YrsInService>10</YrsInService>
  <AgeInYears>54</AgeInYears>
  <WorkDept>D01</WorkDept>
</EmpRecord>
```

Archived



Troubleshooting

This appendix provides basic troubleshooting advice for when a file transfer is not completing successfully in one of the scenarios that we describe in this book. This appendix is arranged by product. In the event of a file transfer failure, review the sections in this appendix, which might help diagnose the cause of the failure.

Guidance is provided on the following topics:

- ▶ How to make sure that the products are running as expected
- ▶ Where to find product log files
- ▶ The host names, URLs, and port numbers that are used for each protocol into Sterling File Gateway, both directly and using a secure proxy server

Sterling File Gateway and Sterling B2B Integrator

This section describes basic troubleshooting for Sterling File Gateway and Sterling B2B Integrator.

Software running

To make sure that Sterling B2B Integrator and Sterling File Gateway are running, log on to the Sterling B2B Integrator system (SysC) and view the Services console in Microsoft Windows (**Start** → **Run** → **services.msc** → **OK**). Verify that all of the Sterling Commerce services are started (Figure C-1).

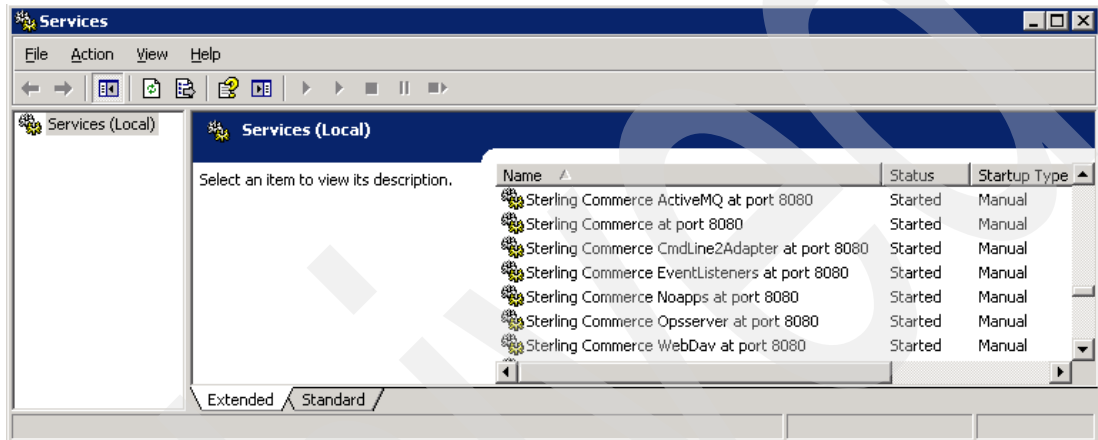


Figure C-1 Status of Sterling products

If in any doubt, restart Sterling B2B Integrator (and Sterling File Gateway):

1. Stop Sterling B2B Integrator. Run:
`<SI_install_dir>/bin/stopWindowsService.cmd`
2. Restart Sterling B2B Integrator by double-clicking the **Sterling_Integrator_at_8080** desktop icon.

Sterling File Gateway transfer status

If a file is not routed as expected, view the file transfer status in Sterling File Gateway. If the product that is involved in sending the file into Sterling File Gateway (either WebSphere MQ File Transfer Edition or Sterling Connect:Direct, depending on whether it is the inbound or outbound scenario) reports a success, then Sterling File Gateway is the logical place to check.

Follow these steps:

1. Track the file transfer in Sterling File Gateway on SysC. Start Internet Explorer and go to:
`http://<servername>:<port>/filegateway/`
2. Log in using your administrator user ID and password (Figure C-2). The default user ID for Sterling File Gateway is `fg_sysadmin`.

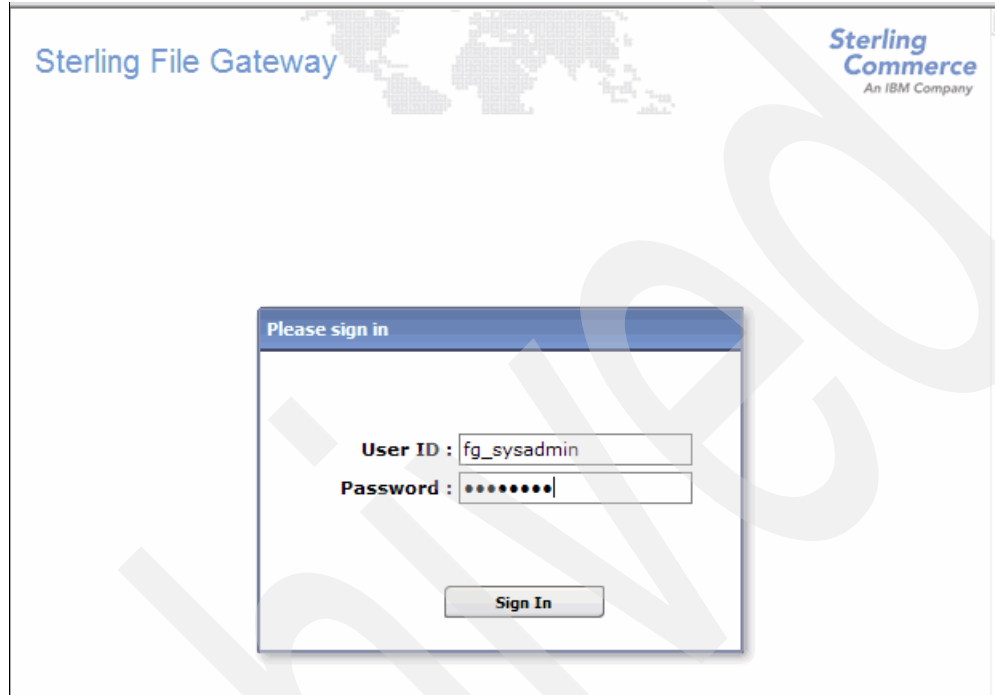


Figure C-2 Log in page for Sterling File Gateway

3. On the main page for Sterling File Gateway (Figure C-3), click **Find** to view all transfers through Sterling File Gateway.

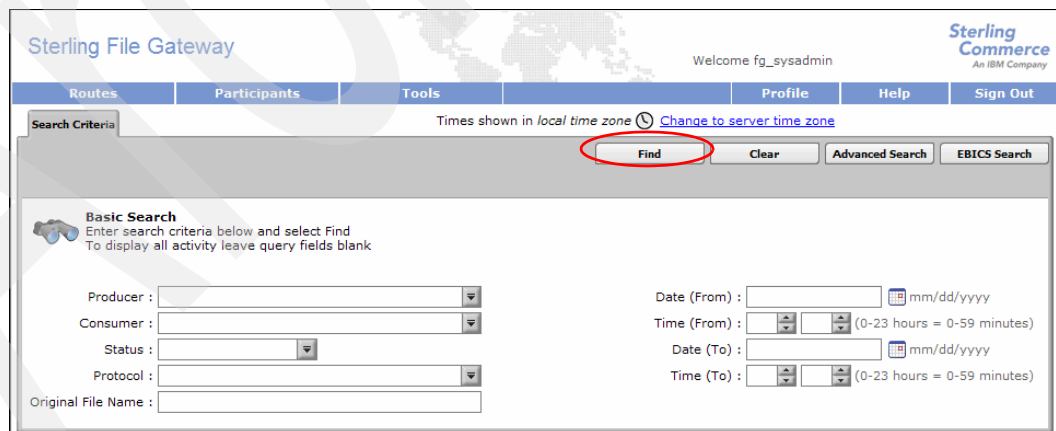


Figure C-3 First page when logged in to Sterling File Gateway

4. The Arrived File tab shows all Sterling File Gateway activity. Find the transfer that has failed and click it to view the detailed status of the transfer. Browse between the Arrived File, Route, and Delivery tabs to find the cause of the failure. Click any links to workflow processes that were invoked because they often provide details about failures during the processing or routing of the file.

Incorrect mailbox path

If the file transfer does not appear in Sterling File Gateway, but the application (WebSphere MQ File Transfer Edition or Sterling Connect:Direct) used to send the file into Sterling File Gateway reports a successful transfer, it suggests that the mailbox path was specified incorrectly in the transfer. The file has probably been transferred successfully to Sterling B2B Integrator, but did not get placed in the correct mailbox path for the correct user. The routing channel was not activated to route the file to the destination, and no status displays on the Sterling File Gateway console. If this is the case, the file will still be visible in the Sterling B2B Integrator console.

Follow these steps:

1. Start Internet Explorer and go to:
`http://<servername>:<port>/filegateway/`
2. Log in using the Sterling File Gateway administrator user ID and password (Figure C-4). The default administrator user ID is `fg_sysadmin`.

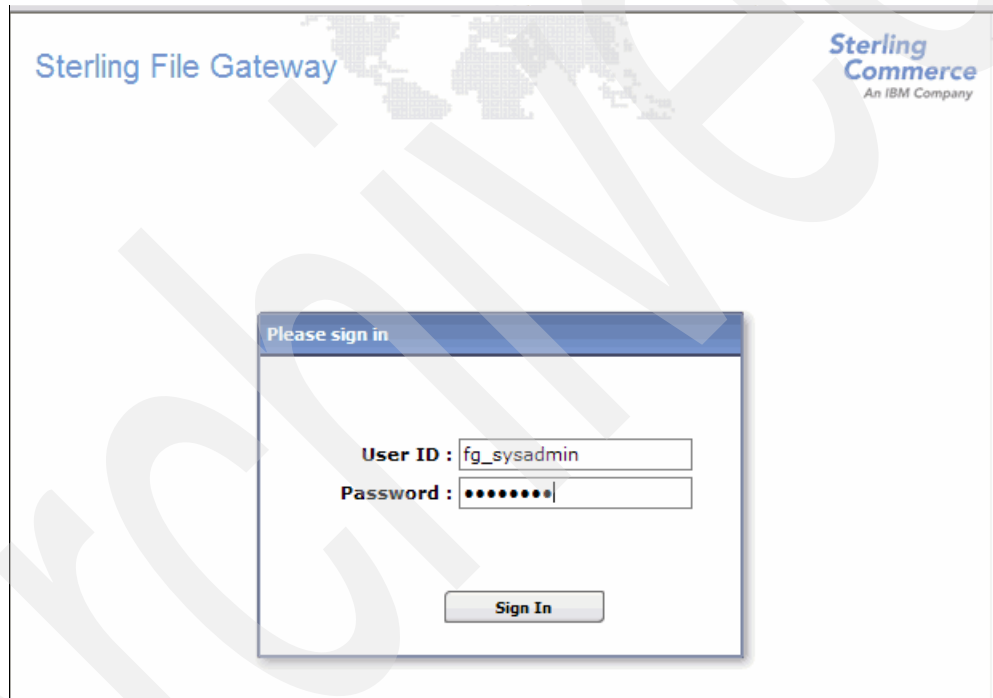


Figure C-4 Sterling File Gateway login page

3. In Sterling File Gateway, go to **Tools** → **B2B Console** (Figure C-5) to open the Sterling B2B Integrator console in a new browser window.



Figure C-5 Open the Sterling B2B Integrator console

4. Go to **Deployment** → **Mailboxes** → **Messages** and click **Go!** to view the messages in all mailboxes (Figure 7-114).

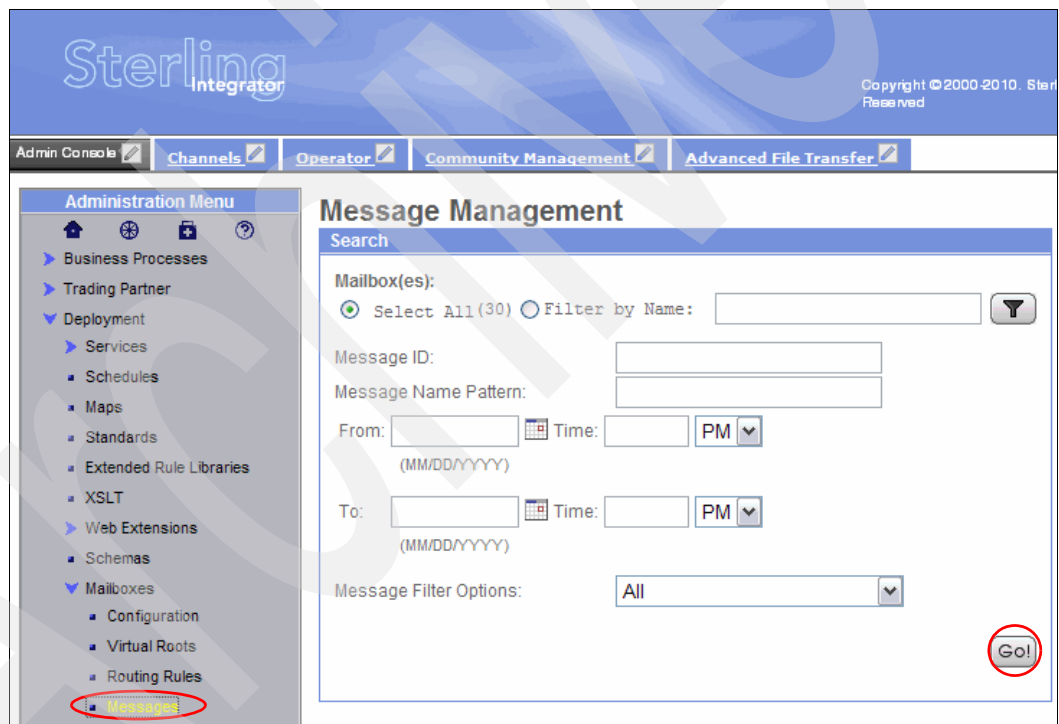


Figure 7-114 View all mailboxes

5. All the messages in all mailboxes should display with the location of the transferred file (Figure C-6) so that you can validate whether it was sent to the correct mailbox.



Message Management						
Messages 31-45 of 54			Page: < 1 2 3 4 >			
Select	Name ▲▼	Id ▲▼	Created ▲▼	Size ▲▼	Mailbox ▲▼	Extract Policy
	SysA_to_SysD.txt	269	12/02/2010 4:16:14 PM EST	106	/SysD_Partner/Inbox	Extractable Count
	SysD_to_SysA.txt	280	12/03/2010 9:45:36 AM EST	124	/SysA_CD_Partner/Inbox	Extractable Count

Figure C-6 Example of messages shown in Sterling B2B Integrator

Sterling File Gateway and Sterling B2B Integrator log files

If your transferred file is not visible in Sterling File Gateway or Sterling B2B Integrator, it might be necessary to view the product log files. The Sterling B2B Integrator logs are on SysC in the `<SI_install_root>/logs` directory. Sort the files by date, and view the most recent files to determine whether a problem occurred in Sterling B2B Integrator.

WebSphere MQ File Transfer Edition

This section provides information about troubleshooting WebSphere MQ and WebSphere MQ File Transfer Edition and how to diagnose errors that might occur in the scenarios. You can find detailed information about WebSphere MQ File Transfer Edition diagnostic messages in the information center at:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/index.jsp?topic=/com.ibm.wmqfte.messages.doc/messages_main.htm

You can find detailed information about WebSphere MQ error codes at:

<http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>

Checking the system requirements

It is important for the installation, configuration, and operation of all the WebSphere MQ and WebSphere MQ File Transfer Edition objects that your systems are compliant with the system requirements for the software. See the following website for a list of the system requirements:

<http://www-01.ibm.com/software/integration/wmq/filetransfer/requirements>

If you are in doubt, contact IBM and ask for help.

Software running

To make sure that WebSphere MQ is running, log in to the WebSphere MQ systems and view the Services console in Microsoft Windows (**Start** → **Run** → **services.msc** → **OK**). Verify that IBM MQSeries® is started (Figure C-7).



Figure C-7 Services console showing that IBM MQSeries is running

There should also be a WebSphere MQ icon in the Microsoft Windows system tray with a green icon, which indicates that WebSphere MQ is running (Figure C-8).

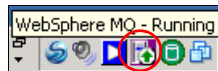


Figure C-8 WebSphere MQ icon in the system tray

Log files

When errors occur or file transfers do not start, examine the following log files:

- ▶ Agent log file (output0.l log)

The agent log file is created in the agent's log directory:

```
<configuration_directory>\<coordination_qmgr_name>\agents\<agent_name>\logs
```

For example, for the SYSDAGT agent the log directory is
C:\IBM\WMQFTE\config\QMSAFE\agents\SYSDAGT\logs.

The log contains records of the agent's events.

- ▶ The agent.lck file

This file is created in the agent's configuration directory:

```
<configuration_directory>\<coordination_qmgr_name>\agents\<agent_name>\logs
```

It contains the agent's process ID (PID), if it is running.

- ▶ FFDC/ABEND files

These files are created in the agent's log directory at:

```
<configuration_directory>\<coordination_qmgr_name>\agents\<agent_name>\logs
```

They are often created as a result of an unexpected error. Analyzing the information in these files requires assistance from IBM support or IBM service.

- ▶ Queue manager log files

Log files are created for each queue manager in the WebSphere MQ

```
<installation_directory>\Qmgrs\<queue_manager_name>\errors directory.
```

Gathering diagnostics by enabling trace files

Tracing can be enabled for each agent by executing the following commands:

- ▶ **fteSetAgentTraceLevel -traceLevel all**

This sets the trace level for an already running agent.

- ▶ **fteStartAgent -trace =all**

This starts the agent with trace enabled.

The trace output from both commands goes into the agent's log directory:

<configuration_directory>\<coordination_qmgr_name>\agents\<agent_name>\logs

You can set the following trace levels:

- ▶ off (default)
- ▶ flow
- ▶ moderate
- ▶ all

Be careful when enabling tracing for agents, because that can have a significant impact on performance and produce a huge amount of data.

No agent listed in MQ Explorer or by the fteListAgents command

If you have created an agent using the `fteCreateAgent` or `fteCreateBridgeAgent` command but your agent is not listed in WebSphere MQ Explorer or is not shown when you execute the `fteListAgents` command, the problem might be the result of any one of a number of causes. The flow chart in Figure C-9 can help you identify the cause.

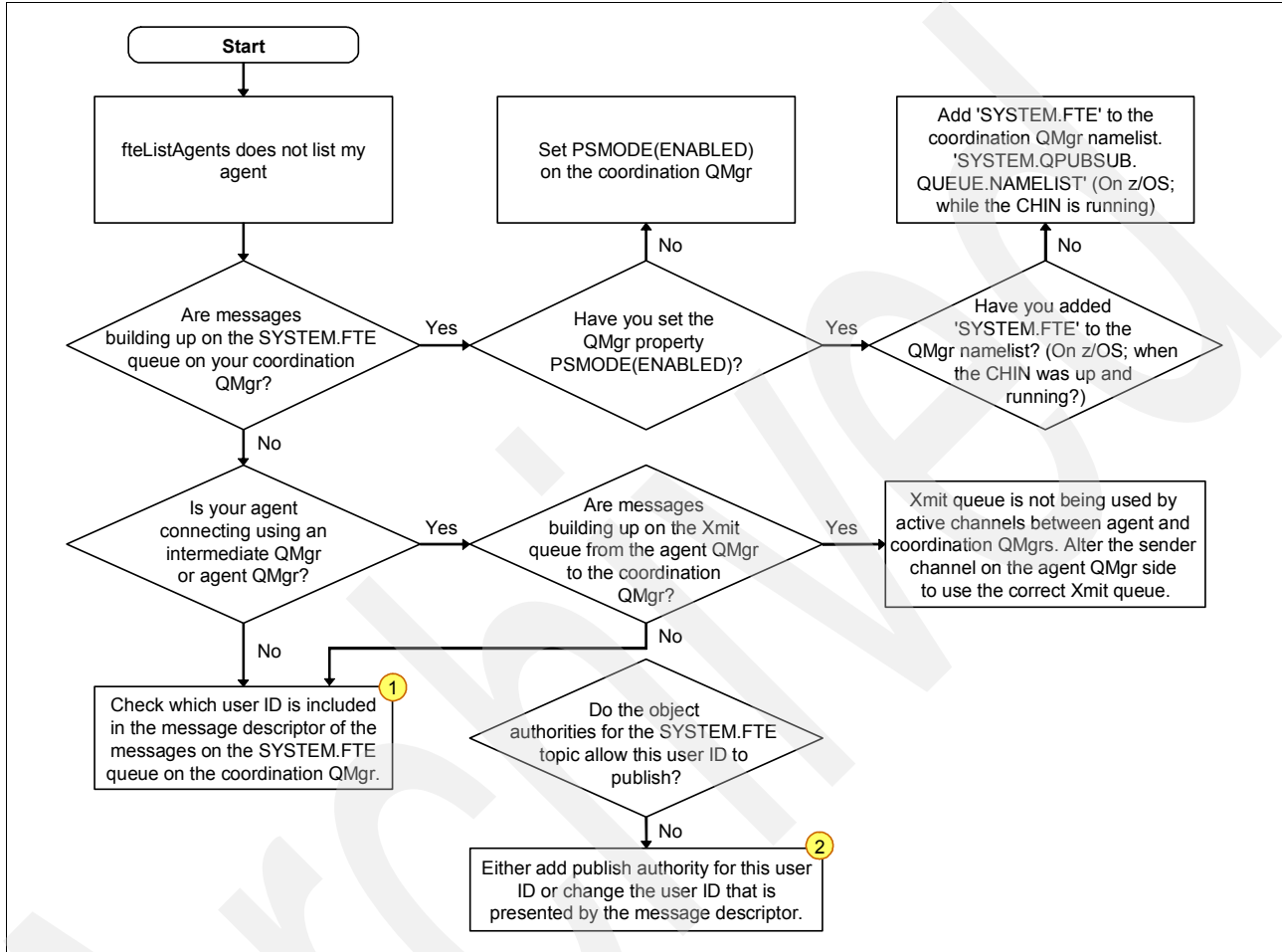


Figure C-9 Flowchart for debugging an agent problem

Visit the following website for further information about how to solve the problem:

http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/index.jsp?topic=/com.ibm.wmqfte.admin.doc/list_agents_pd.htm

File transfer does not start

There can be various reasons for a file transfer request to fail to start. The following steps can help you find the problem:

1. Determine whether the transfer requests are getting to the agent.
 - a. Stop the agent.
 - b. Send a new transfer request.
 - c. Check to see whether the request arrives on the agent's command queue (the `SYSTEM.FTE.COMMAND.<agent_name>` at the agent queue manager).
2. Make sure that all the relevant channels between the queue managers (cluster-sender, cluster-receiver) are running.

3. Check whether the source agent is running and has started the transfer:
 - a. Ping the agent using the **ftePingAgent** command.
 - b. Check the agent's command queue. It should be empty.
 - c. Use the **fteListAgents** and **fteShowAgentDetails** commands to show information about current transfers (use the -v option).
4. Determine whether the transfer took place but is not shown in the WebSphere MQ File Transfer Edition Explorer Log view. Make sure that the user in the MQMD user field is authorized to send status messages to the coordination queue manager.

Sterling Connect:Direct

In Sterling Connect:Direct, all activity and statistics are logged so that there are verifiable audit trails of any actions. Each step of a Connect:Direct process from process submission to process completion are logged in the Connect:Direct statistics logs. The following procedures show how to access and view the Connect:Direct statistics logs.

The statistics logs can be accessed through a command-line interface (CLI) or the desktop client, called Connect:Direct requester. We begin by showing how to access the statistics logs using the Connect:Direct requester. Next, we show how to access the statistics logs using the CLI.

Connect:Direct requester

Accessing the static logs using Connect:Direct requester:

1. Select **Start** → **All Programs** → **Sterling Commerce Sterling Connect:Direct vX.X** → **CD Requester**.
2. Double-click **Select Statistics** (Figure C-10).

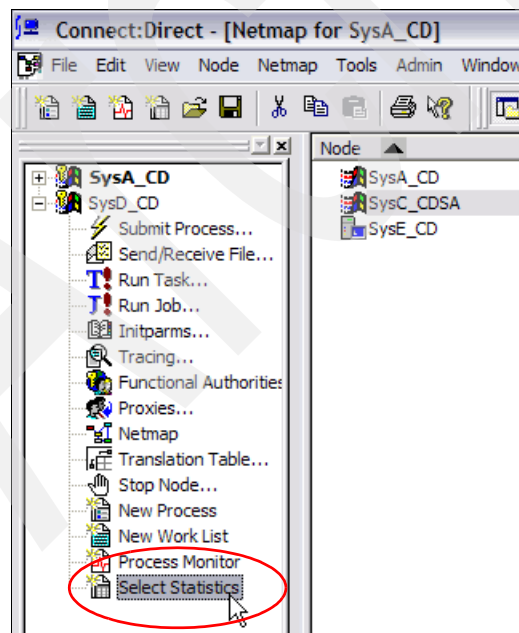


Figure C-10 Launching the Select Statistics dialog within Connect:Direct requester

- In the Select Statistics dialog box, enter the amount of time for which Sterling Connect:Direct should retrieve statistics logs. The time is in hours and minutes. If Connect:Direct requester has access to multiple Connect:Direct nodes, optionally select the node from which to retrieve statistics.

In our test, we retrieve the last 10 minutes of statistics logs that are on SysD_CD (Figure C-11). After making your selections, click **OK**.

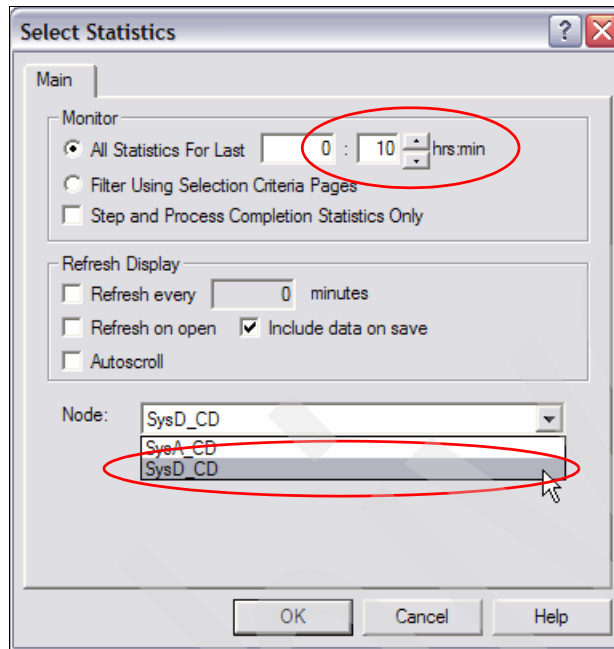


Figure C-11 Filtering the Select Statistics query within Connect:Direct requester

- In Figure C-12, the statistics logs are shown as records. This view enables a user to quickly scan over the records. The CC column provides the success or failure value of an individual step within a Connect:Direct process. A value of zero indicates success. Any other value indicates a failure of that step. Figure C-12 shows the records for a complete process starting from record 2 through record 10. Double-click any record to obtain more details about that record.

	Log Date/Time	Type	RecID	CC	FDBK	MSGID	PName	PNum	Step Name
1	12/3/2010 4:14:34 PM	CAEV	SPCA	4	0	CSPA314W		0	
2	12/3/2010 4:14:34 PM	CAEV	SMIN	0	0	LSMI001I		0	
3	12/3/2010 4:14:34 PM	CAEV	PMRC	0	0			0	
4	12/3/2010 4:14:34 PM	CAEV	SSTR	0	0	LSMI003I		0	
5	12/3/2010 4:14:34 PM	CAPR	PSTR	0	0	LSMG200I	ToSysD	27	
6	12/3/2010 4:14:35 PM	CAPR	RSST	0	0		ToSysD	27	step01
7	12/3/2010 4:14:35 PM	CAPR	CTRC	0	0	SCPA000I	ToSysD	27	step01
8	12/3/2010 4:14:35 PM	CAPR	PRED	0	0	XSMG252I	ToSysD	27	
9	12/3/2010 4:14:35 PM	CAEV	SEND	0	0	LSMI008I		0	
10	12/3/2010 4:14:35 PM	CAEV	SMED	0	0	LSMI002I		0	

Figure C-12 Connect:Direct requester Statistics records view

- Figure C-13 shows the detailed view for a statistics record. This example shows the detail for record number 7. Double-clicking the record launches the detailed view of the record. In this output, a file was copied successfully. If there is any failure in the Connect:Direct transfer, the detailed explanation within the Statistics Detail dialog box shows that information.

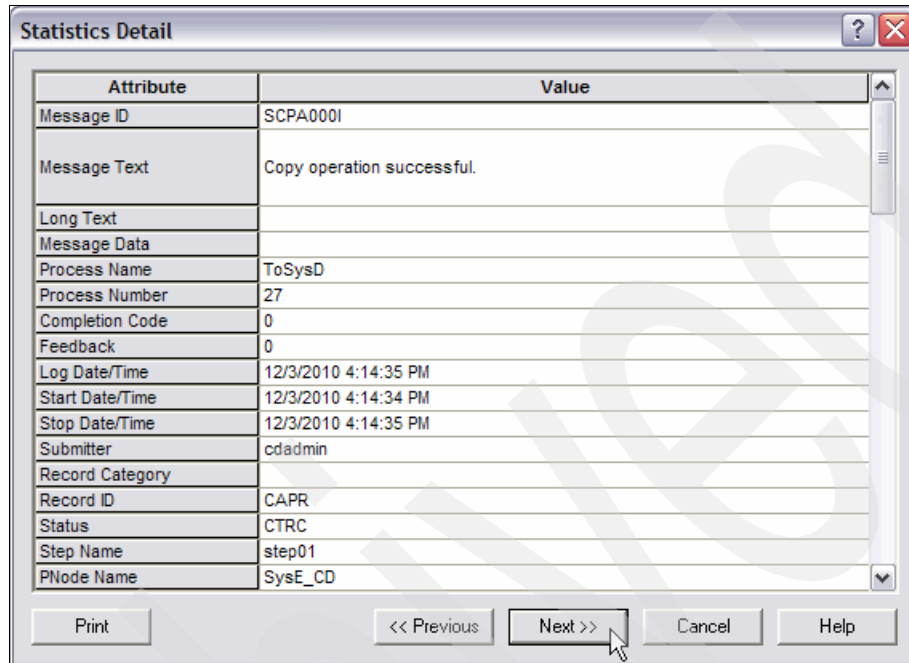


Figure C-13 Connect:Direct requester Statistics Detail dialog

Sterling Connect:Direct CLI

You can start the Sterling Connect:Direct CLI by running the **direct** command:

- On UNIX and Linux, the **direct** command is located in the `<installation path>/ndm/bin` directory.
- On Windows, the **direct** command is located in the `C:\Program Files\Sterling Commerce\Connect Direct v4.5.01\Common Utilities\direct.exe` directory.

This example shows the usage of the **direct** command on Linux. To view the statistics logs on Sterling Connect:Direct Linux, execute the **direct** command as shown in Example C-1.

In this scenario, a Connect:Direct process was previously submitted at 11:45. The **select statistics** command prints all statistics logs by default. This example limits the output by passing in arguments to the **select statistics** command. The argument **startt=(today, 11:40:00)** instructs the **direct** program to show only statistics that have been generated today since 11:40.

Example C-1 The select statistics command using time-based filtering

```

$ ./direct
Direct> select statistics startt=(today, 11:40:00);
=====
                        SELECT  STATISTICS
=====
P  RECID LOG TIME          PNAME          PNUMBER  STEPNAME  CCOD  FDBK  MSGID
E  RECID LOG TIME          MESSAGE TEXT
=====

```

```

E CXIT 12/04/2010 11:45:18 CMGR exited. Pid=9687. Exitcode=1.
E SGON 12/04/2010 11:45:21 User sign on completed.
E QCEX 12/04/2010 11:46:45 TCQ queue change from WAIT to EXEC, status PE.
E SSTR 12/04/2010 11:46:45 Session started, SNODE:SYSD_CD, Protocol:tcp
                                LCLP      9.42.170.168, PORT=33312
                                RMTTP     9.42.170.129, PORT=1364
P PSTR 12/04/2010 11:46:45 ToSysD      28          0          XSMG200I
P LSST 12/04/2010 11:46:45 ToSysD      28 step01    0          XSMG201I
P CTXC 12/04/2010 11:46:46 ToSysD      28 step01    0          SCPA000I
P PRED 12/04/2010 11:46:46 ToSysD      28          0          XSMG252I
E SEND 12/04/2010 11:46:46 Session ended, Session Manager shutting down SNODE:
                                SYSD_CD
E CXIT 12/04/2010 11:46:46 Pnode SMGR exited. Pid=9730. Exitcode=1.
=====
Select Statistics Completed Successfully.
Direct>

```

The statistics logs are shown as records. This view enables you to scan quickly over the records. The CC column provides the success or failure value of an individual step within a Connect:Direct process. A value of zero indicates success. Any other value indicates a failure of that step. Example C-1 on page 390 shows the records for a complete process.

To obtain more detailed output using select statistics from the CLI, use the additional arguments **detail=yes** (Example C-2).

Example C-2 The detailed statistics command output using the detail argument

```

Direct> select statistics startt=(today, 11:40:00) detail=yes;
=====
                        SELECT  STATISTICS
=====
EVENT RECORD      Record Id => CXIT
Process Name      =>          Stat Log Date   => 12/04/2010
Process Number    => 0          Stat Log Time   => 11:45:18
Submitter Class   =>
Submitter Id      =>

Step Start Date   =>          Step Start Time =>
Step Stop Date    =>          Step Stop Time  =>
Step Elapsed Time=>

From node         => S
Rstr              =>
SNODE             =>
Completion Code   => 0
Message Text      => CMGR exited. Pid=9687. Exitcode=1.
-----
EVENT RECORD      Record Id => SGON
Process Name      =>          Stat Log Date   => 12/04/2010
Process Number    => 0          Stat Log Time   => 11:45:21
Submitter Class   =>
Submitter Id      => cdadmin

Step Start Date   => 12/04/2010 Step Start Time => 11:45:21
Step Stop Date    =>          Step Stop Time  =>
Step Elapsed Time=>

```

From node => S
Rstr =>
SNODE =>
Completion Code => 0
Message Text => User sign on completed.

EVENT RECORD Record Id => QCEX
Process Name => ToSysD Stat Log Date => 12/04/2010
Process Number => 28 Stat Log Time => 11:46:45
Submitter Class =>
Submitter Id => cdadmin@SysE_CD

Step Start Date => Step Start Time =>
Step Stop Date => Step Stop Time =>
Step Elapsed Time=>

From node => S
Rstr =>
SNODE => SysD_CD
Completion Code => 0
Message Text => TCQ queue change from WAIT to EXEC, status PE.

EVENT RECORD Record Id => SSTR
Process Name => Stat Log Date => 12/04/2010
Process Number => 28 Stat Log Time => 11:46:45
Submitter Class =>
Submitter Id =>

Step Start Date => 12/04/2010 Step Start Time => 11:46:45
Step Stop Date => 12/04/2010 Step Stop Time => 11:46:45
Step Elapsed Time=> 00:00:00

From node => S
Rstr =>
SNODE => SYSD_CD
Completion Code => 0
Message Text => Session started, SNODE:SYSD_CD, Protocol:tcp
LCLP 9.42.170.168, PORT=33312 RMTP 9.42.170.129, PORT=1364

PROCESS RECORD Record Id => PSTR
Process Name => ToSysD Stat Log Date => 12/04/2010
Process Number => 28 Stat Log Time => 11:46:45
Submitter Class => 1
Submitter Id => cdadmin@SysE_CD

Step Start Date => 12/04/2010 Step Start Time => 11:46:45
Step Stop Date => 12/04/2010 Step Stop Time => 11:46:45
Step Elapsed Time=> 00:00:00

From node => S
Rstr =>
SNODE => SYSD_CD
Completion Code => 0
Message Id => XSMG200I

Short Text => Process started, process:28 name:ToSysD SNODE:SYSD_CD

PROCESS RECORD Record Id => LSST
Process Name => ToSysD Stat Log Date => 12/04/2010
Process Number => 28 Stat Log Time => 11:46:45
Submitter Class =>
Submitter Id => cdadmin@SysE_CD

Step Start Date => 12/04/2010 Step Start Time => 11:46:45
Src File => msgfile.cfg
Dest File => C:\CDWindows_files\download\cddelete.me

Step Name => step01
From node => P
Rstr => N
SNODE => SYSD_CD
Completion Code => 0
Message Id => XSMG201I
Short Text => Local Step started.

PROCESS RECORD Record Id => CTRC
Process Name => ToSysD Stat Log Date => 12/04/2010
Process Number => 28 Stat Log Time => 11:46:46
Submitter Class =>
Submitter Id => cdadmin@SysE_CD

Step Start Date => 12/04/2010 Step Start Time => 11:46:45
Step Stop Date => 12/04/2010 Step Stop Time => 11:46:46
Step Elapsed Time=> 00:00:01

Step Name => step01
From node => P
Rstr => N
SNODE => SYSD_CD
Completion Code => 0
Message Id => SCPA000I
Short Text => Copy step successful.

Ckpt=>Y Lkfl=>N Rstr=>N Xlat=>N Scmp=>N Ecmp=>Y Ecpr=>77.13 CRC=>N
Zlvl=>1 Zwin=>13 Zmem=>4

Local node => P
From node => P
Src File => msgfile.cfg
Dest File => C:\CDWindows_files\download\cddelete.me

Source		Destination	
Ccode	=>0	Ccode	=>0
Msgid	=>SCPA000I	Msgid	=>SCPA000I
Bytes Read	=>4564255	Bytes Written	=>4564255
Recs Read	=>210443	Recs Written	=>210443
Bytes Sent	=>1043738	Bytes Recvd	=>1043738
Rus Sent	=>19	Rus Recvd	=>19
Ru Size	=>65535		

PROCESS RECORD Record Id => PRED
Process Name => ToSysD Stat Log Date => 12/04/2010

Process Number => 28 Stat Log Time => 11:46:46
Submitter Class =>
Submitter Id => cdadmin@SysE_CD

Step Start Date => 12/04/2010 Step Start Time => 11:46:45
Step Stop Date => 12/04/2010 Step Stop Time => 11:46:46
Step Elapsed Time=> 00:00:01

From node => S
Rstr =>
SNODE => SYSD_CD
Completion Code => 0
Message Id => XSMG252I
Short Text => A C->D process has completed successfully.

EVENT RECORD Record Id => SEND
Process Name => Stat Log Date => 12/04/2010
Process Number => 28 Stat Log Time => 11:46:46
Submitter Class =>
Submitter Id =>

Step Start Date => 12/04/2010 Step Start Time => 11:46:45
Step Stop Date => 12/04/2010 Step Stop Time => 11:46:46
Step Elapsed Time=> 00:00:01

From node => S
Rstr =>
SNODE => SYSD_CD
Completion Code => 0
Message Text => Session ended, Session Manager shutting down SNODE:SYS
 D_CD

EVENT RECORD Record Id => CXIT
Process Name => Stat Log Date => 12/04/2010
Process Number => 0 Stat Log Time => 11:46:46
Submitter Class =>
Submitter Id =>

Step Start Date => Step Start Time =>
Step Stop Date => Step Stop Time =>
Step Elapsed Time=>

From node => S
Rstr =>
SNODE =>
Completion Code => 0
Message Text => Pnode SMGR exited. Pid=9730. Exitcode=1.

=====
Select Statistics Completed Successfully.
Direct>

Check ports

If you are having connection issues, make sure that you are using the correct ports, as listed in Table 7-11. Table 7-11 describes the ports as they were set up in this book. Your ports might differ, depending on your installation.

Table 7-11 Ports used in our environment

Product	Protocol	URL	Direct	Using a proxy server
Sterling File Gateway administration console	HTTP	http://<hostname>:<port>/filegateway	SysC, port 8080	Not available through proxy server
Sterling File Gateway administration console	HTTPS	http://<hostname>:<port>/filegateway	SysC, port 8081	Not available through proxy server
myFileGateway	HTTP	http://<hostname>:<port>/myfilegateway	SysC, port 8080	Not available through proxy server
myFileGateway	HTTPS	http://<hostname>:<port>/myfilegateway	SysC, port 10000	SysB, port 10050
Sterling B2B Integrator administration console	HTTP	http://<hostname>:<port>/dashboard	SysC, port 8080	Not available through proxy server
Sterling B2B Integrator administration console	HTTPS	http://<hostname>:<port>/dashboard	SysC, port 8081	Not available through proxy server
Sterling Connect:Direct from SysA_CD to SysC_CDSA	Connect:Direct	N/A	SysC, port 1364	SysB, port 1364
Sterling Connect:Direct from SysC_CDSA to SysA_CD	Connect:Direct	N/A	SysA, port 1364	SysB, port 1364
Sterling Connect:Direct from SysD_CD to SysE_CD	Connect:Direct	N/A	SysE, port 1364	Not available through proxy server (internal transfer)
Sterling Connect:Direct from SysE_CD to SysD_CD	Connect:Direct	N/A	SysD, port 1364	Not available through proxy server (internal transfer)
SFTP client from SysA to SysC	SFTP	N/A	SysC, port 8119	SysB, port 10052

WebSphere Message Broker tips

The WebSphere Message Broker flow is built such that when an error occurs, the failure terminal from a node is routed to a trace node. You can configure the trace node to write records to a user trace file, another file, or the local error log. Many of the errors that occur when trying to test and run a new message flow can be revealed through the use of a trace node.

You can find more information about the trace node in the WebSphere Message Broker Information Center:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.doc/ac04840_.htm

Transfers do not appear in WebSphere MQ File Transfer Edition Explorer

After your flow is working and you can write files to the local file system, check the WebSphere MQ File Transfer Edition Explorer Transfer Log to see whether file transfers that are sent from BRKR.AGT to another agent are appearing. If they are not, use the following steps to diagnose the issue.

For use with other troubleshooting: You can also use the following steps to help diagnose other issues that are associated with WebSphere Message Broker that keep the flow from working as designed but that are not creating any errors captured from the failure terminal of a node or with the trace node.

1. From a command prompt, enter `mqsiprofile` (Example C-3). The `mqsiprofile` command completes the environment initialization on systems such as Windows, where the components run as services. The services do not inherit the environment that is set for the command prompt. However, the services do run the `mqsiprofile` command when they start to pick up any environmental changes.

Example C-3 WebSphere Message Broker mqsiprofile command

```
>mqsiprofile
```

```
MQSI 7.0.0.1
```

2. To set the tracing characteristics for a broker to aid in diagnosing the issue, enter the following command:

```
mqsichangetrace <MB7BROKER> -u -e <default> -r -l debug
```

This command uses the following parameters and variables:

<MB7BROKER>	Replace this with the name of your broker.
-u	Specifies that the collection of user trace is being modified.
-e	Gives the execution group that the trace is collected on.
<default>	Replace this with the name of your execution group where the flow is deployed.
-r	Requests that the trace log is reset to discard all current records to ensure that all records in the log are unique to the new trace.
-l	Sets the trace level. The debug option provides the most comprehensive trace.

For more information about the `mqsichangetrace` command, refer to the `mqsichangetrace` topic in the WebSphere Message Broker Information Center:

http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.doc/an28100_.htm

Example C-4 shows the command that we used in our environment.

Example C-4 Capture trace at a debug level

```
>mqsichangetrace BRKR -u -e AGT -r -l debug  
BIP8071I: Successful command completion.
```

3. Run a test on your flow by sending a file to BRKR.AGT to be processed in WebSphere Message Broker.

4. To read the trace records and write the records to a file, enter the following command:

```
mqsireadlog <MB7BROKER> -u -e <default> -o <C:\temp\trace.xml>
```

The records are written to the log in an XML format. This command uses the following parameters and variables:

<MB7BROKER>	The name of your broker.
-u	Specifies to read the log contents from the user trace log.
-e	Specifies the execution group for which the log information is read.
<default>	The name of your execution group running the flow.
-o	Specifies the name of the file to which to write the log data.
<C:\temp\trace.xml>	The location and file name where you want the log written.

Example C-5 shows the command for our scenario.

Example C-5 Read the user trace to a log file

```
>mqsireadlog BRKR -u -e AGT -o C:\temp\trace.xml  
BIP8071I: Successful command completion.
```

5. To format the log created in the previous step from XML to a form suitable for your locale, enter the following command into a command prompt:

```
mqsiformatlog -i <C:\temp\trace.xml> -o <C:\temp\trace.txt>
```

This command uses the following parameters and variables:

-i	Specifies the input file to be formatted.
<C:\temp\trace.xml>	The XML file created in the prior step where the <code>mqsireadlog</code> command is issued.
-o	Specifies the location where the new output file is written.
<C:\temp\trace.txt>	The output file that is created when the <code>mqsiformatlog</code> command completes.

Example C-6 shows the command for our scenario.

Example C-6 Parse the XML log

```
>mqsiformatlog -i C:\temp\trace.xml -o C:\temp\trace.txt  
BIP8071I: Successful command completion.
```

6. To turn off the trace, enter the following command into a command prompt:

```
mqsichangetrace <MB7BROKER> -u -e <default> -r -l none
```

This command uses the following parameters and variables:

<MB7BROKER>	Replace with the name of your broker.
-u	Specifies that the collection of user trace is being modified.
-e	Gives the execution group that the trace is collected on.
<default>	Replace with the name of your execution group where the flow is deployed.
-r	Requests that the trace log is reset to discard all current records to ensure that all records in the log are unique to the new trace.
-l	Sets the trace level. None turns the trace off.

Example C-7 shows the command that we used in our scenario.

Example C-7 Turn off trace

```
>mqsichangetrace BRKR -u -e AGT -r -l none  
BIP8071I: Successful command completion.
```

7. Review the log file that you created to check for errors encountered during the broker flow that are not errors from the broker nodes. The log contains an entry similar to Example C-8.

Example C-8 Log entry

Timestamps are formatted in local time, 300 minutes before GMT.

Trace written by version ; formatter version 7001 (build S700-FP01)

```
2010-11-24 11:36:18.856901      2564  UserTrace  BIP4040I: The Execution Group  
'AGT' has processed a configuration message successfully.  
A configuration message has been processed successfully. Any configuration  
changes have been made and stored persistently.  
No user action required.  
2010-11-24 11:36:18.857168      2564  UserTrace  BIP2638I: The MQ output node  
'outputNode' attempted to write a message to queue  
'SYSTEM.BROKER.EXECUTIONGROUP.REPLY' connected to queue manager 'BRKRQMGR'.  
The MQCC was '0' and the MQRC was '0'.  
2010-11-24 11:36:18.857177      2564  UserTrace  BIP2622I: Message  
successfully output by output node 'outputNode' to queue  
'SYSTEM.BROKER.EXECUTIONGROUP.REPLY' on queue manager 'BRKRQMGR'.  
2010-11-24 11:36:18.857307      2564  Information BIP2154I: Execution group  
finished with Configuration message.  
A command response will be sent to the broker.  
No user action required.  
2010-11-24 11:36:36.436061      5168  UserTrace  BIP3368I: FTEInput node 'FTE  
Input' in message flow 'InboundFileTransferFlow' has received a transfers  
from agent 'WMBBRIDGEAGT'. Details: Job name=''; Transfer  
ID='323031303131323431313132333633343835383200000000'; File  
path='C:\FileTransfersInbound\UseRouteNode-KeyNotFound.xml'; File  
name='UseRouteNode-KeyNotFound.xml'.  
The FTEInput node is about to process the transfer.  
No action is required.
```

2010-11-24 11:36:36.436260 5168 UserTrace BIP4144I: Entered function 'cniCreateElementAsLastChild'(1c54f150, ee9ba0, 'N/A', 'N/A', 'N/A', 'N/A', 'N/A', 'N/A', 'N/A', 'N/A', 'N/A').
Entered the specified function with the specified parameters.
No user action required.

8. An error message regarding your broker agent's ability to publish status messages to the WebSphere MQ File Transfer Edition coordination queue manager might appear (similar to Example C-9). The agent's ability to publish status messages also allows the transfer to be seen in WebSphere MQ File Transfer Edition Explorer under the transfer log and provides information to the database logger through the coordination queue manager.

Example C-9 WebSphere MQ File Transfer Edition failed publication error message

2010-11-24 11:36:37.773422 6576 UserTrace BIP3484E: 'BFGCH0069W: A transfer log message has failed publication. Details of the failure are as follows : Agent Name : 'BRKR.AGT' Transfer Id : '414d512042524b52514d475220202015d8e24c20e6d103' Message Type : 'Log' Topic Name : 'SYSTEM.FTE' MQ Reason Code : '3081' MQ Error Code : '2035''
An embedded component has written the diagnostic message included here.
Refer to the appropriate message in the embedded component's documentation.

Archived

Sample files

This appendix provides samples of the Ant scripts that were used in 4.3.5, “Configuring WebSphere MQ File Transfer Edition” on page 75, and the customization files used in 6.3.10, “Creating custom WebSphere MQ File Transfer Edition protocol” on page 199.

The customization files referred to in “Sample files used in WebSphere MQ File Transfer Edition within Sterling File Gateway” on page 406 are available for download. See Appendix E, “Additional material” on page 425, for detailed download instructions.

Sample Ant scripts

Three sample Ant scripts were created in Chapter 4, “Managed file transfer within an enterprise” on page 51. The sample code for these scripts follow.

Push_to_CD.xml script

The sample code shown in Example D-1 is used in the scenario described in 4.2.5, “WebSphere MQ File Transfer Edition pushing to Sterling Connect:Direct” on page 60.

Example D-1 Push_to_CD.xml

```
<?xml version="1.0" ?>
<project xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs" name="Push_to_CD"
default="job" basedir=". ">

<!-- Set global properties for this FTE JOB -->
  <target name="init" description="Set Global variables">
    <!-- In this part we set the global properties which are used in the following
procedure using mainly property task.-->
      <tstamp>
        <format property="timestamp" pattern="ddMMMyy_HHmss" />
      </tstamp>
      <property name="#T" value="{timestamp}" />
      <!-- File name -->
      <property name="srcfile" value="C:\sysctmp\TestFile_from_SysC.txt" />
      <property name="dstfile" value="TestFile_from_SysC.txt" />
      <!-- Temp directory which C:D uses -->
      <property name="uploaddir" value="C:\CDWindows_files\upload" />
      <!-- C:D Process File template -->
      <property name="cdTemplate"
value="C:\CDWindows_files\processes\Template_process_for_push.cdp" />
      <!-- MQFTE Configuration -->
      <property name="SYSCAGT" value="SYSCAGT@FTPQMGR" />
      <property name="SYSDAGT" value="SYSDAGT@FTEQMGR" />
      <property name="cmdqm" value="FTPQMGR" />
      <property name="jobName" value="Push to CD" />
      <property name="deptId" value="FTEAdmin" />
      <property name="priority" value="0" />
      <property name="type" value="text" />
      <property name="overwrite" value="true" />
      <!-- Configuration for InvokeCD.xml -->
      <property name="post_cmd" value="InvokeCD.xml" />
      <property name="post_retry_count" value="0" />
      <property name="post_retry_waitTime" value="0" />
      <property name="post_type" value="antscript" />
      <!-- Generating a unique id for the file transfer job using uuid task-->
      <fte:uuid length="8" property="jobNumber" />
      <condition property="call_post">
        <length string="{post_cmd}" when="gt" length="0" />
      </condition>
    </target>
```

```

<!-- Transfer files -->
<target name="copy" depends="init" description="Transfer files">
  <!-- Requesting to transfer the file using filecopy task -->
  <fte:filecopy cmdqm="{cmdqm}" src="{SYSCAGT}" dst="{SYSDAGT}"
priority="{priority}" idproperty="copy.uuid" outcome="await" jobname="{jobName}"
rcproperty="copy.result">
  <!-- Setting the metadatas related to the file transfer using metadata
task -->
  <fte:metadata>
    <fte:entry name="deptId" value="{deptId}" />
    <fte:entry name="jobNumber" value="{jobNumber}" />
  </fte:metadata>
  <!-- Setting the transferring file -->
  <fte:filespec srcfilespec="{srcfile}"
dstfile="{uploaddir}\{jobNumber}" conversion="{type}" overwrite="true" />
  </fte:filecopy>
  <echo message="[Info] uuid for copy={copy.uuid}" />
</target>

<!-- Call post program -->
<target name="post" depends="copy" description="Call post program"
if="call_post">
  <!-- Requesting to SYSDAGT to run Connect:Direct process which is wrapped by
the InvokeCD.xml script -->
  <fte:call cmdqm="{cmdqm}" agent="{SYSDAGT}" rcproperty="post.result"
jobname="{jobName}" idproperty="post.uuid">
    <fte:command command="{post_cmd}" type="{post_type}" successsrc="0"
retrycount="{post_retry_count}" retrywait="{post_retry_waitTime}">
      <!-- Setting the configurations which are used by InvokeCD.xml -->
      <fte:property name="srcfile" value="{jobNumber}" />
      <fte:property name="dstfile" value="{dstfile}" />
      <fte:property name="UUID" value="{jobNumber}" />
      <fte:property name="jobNumber" value="{jobNumber}" />
      <fte:property name="template" value="{cdTemplate}" />
    </fte:command>
    <!-- Setting the metadata related to the calling task using metadata task
-->
    <fte:metadata>
      <fte:entry name="deptId" value="{deptId}" />
      <fte:entry name="jobNumber" value="{jobNumber}" />
    </fte:metadata>
  </fte:call>
</target>

<!-- call the targets in the order corresponding to the init, copy and post -->
<target name="job" depends="init,copy,post" />

</project>

```

Pull_from_CD.xml script

The sample code shown in Example D-2 is used in the scenario described in 4.2.6, “WebSphere MQ File Transfer Edition pulling from Sterling Connect:Direct” on page 61.

Example D-2 Pull_from_CD.xml

```
<?xml version="1.0" ?>
<project xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs" name="Pull_from_CD"
default="job" basedir=". ">

<!-- Set global properties for this FTE JOB -->
  <target name="init" description="Set Global variables">
    <!-- In this part we set the global properties which are used in the following
procedure using mainly property task.-->
      <tstamp>
        <format property="timestamp" pattern="ddMMMyy_HHmss" />
      </tstamp>
      <property name="#T" value="{timestamp}" />
      <!-- File name -->
      <property name="srcfile" value="TestFile.txt" />
      <property name="dstfile" value="TestFile.txt" />
      <!-- C:D Process File template -->
      <property name="cdTemplate"
value="C:\CDWindows_files\processes\Template_process_for_pull.cdp" />
      <!-- Temp directory which C:D uses -->
      <property name="uploaddir" value="C:\CDWindows_files\download" />
      <!-- MQFTE Configuration -->
      <property name="SYSCAGT" value="SYSCAGT@FTPQMGR" />
      <property name="SYSDAGT" value="SYSDAGT@FTEQMGR" />
      <property name="cmdqm" value="FTPQMGR" />
      <property name="jobName" value="Pull from CD" />
      <property name="deptId" value="FTEAdmin" />
      <property name="priority" value="0" />
      <property name="type" value="text" />
      <property name="overwrite" value="true" />
      <!-- Job Configuration -->
      <property name="pre_cmd" value="InvokeCD.xml" />
      <property name="pre_retry_count" value="0" />
      <property name="pre_retry_waitTime" value="0" />
      <property name="pre_type" value="antscript" />
      <!-- Generating a unique id for the file transfer job using uuid task-->
      <fte:uuid length="8" property="jobNumber" />
      <condition property="call_pre">
        <length string="{pre_cmd}" when="gt" length="0" />
      </condition>
    </target>

    <!-- Call pre program -->
    <target name="pre" depends="init" description="Call pre program" if="call_pre">
      <!-- Requesting to SYSDAGT to run Connect:Direct process which is wrapped by
the InvokeCD.xml script -->
```

```

    <fte:call cmdqm="{cmdqm}" agent="{SYSDAGT}" rcproperty="post.result"
jobname="{jobName}" idproperty="post.uuid">
    <fte:command command="{pre_cmd}" type="{pre_type}" successsrc="0"
retrycount="{pre_retry_count}" retrywait="{pre_retry_waitTime}">
    <!-- Setting the configuration which are used by InvokeCD.xml -->
    <fte:property name="srcfile" value="{srcfile}" />
    <fte:property name="dstfile" value="{jobNumber}" />
    <fte:property name="UUID" value="{jobNumber}" />
    <fte:property name="jobNumber" value="{jobNumber}" />
    <fte:property name="template" value="{cdTemplate}" />
    </fte:command>
    <!-- Setting the metadata related to the calling task using metadata task
-->
    <fte:metadata>
    <fte:entry name="deptId" value="{deptId}" />
    <fte:entry name="jobNumber" value="{jobNumber}" />
    </fte:metadata>
    </fte:call>
</target>

<!-- Transfer files -->
<target name="copy" depends="pre" description="Transfer files">
    <!-- Requesting to transfer the file using filecopy task -->
    <fte:filecopy cmdqm="{cmdqm}" src="{SYSDAGT}" dst="{SYSCAGT}"
priority="{priority}" idproperty="copy.uuid" outcome="await" jobname="{jobName}"
rcproperty="copy.result">
    <!-- Setting the metadatas related to the file transfer using metadata
task -->
    <fte:metadata>
    <fte:entry name="deptId" value="{deptId}" />
    <fte:entry name="jobNumber" value="{jobNumber}" />
    </fte:metadata>
    <!-- Setting the transferring file -->
    <fte:filespec srcfilespec="{uploaddir}\{jobNumber}"
dstfile="{dstfile}" conversion="{type}" overwrite="true" />
    </fte:filecopy>
    <echo message="[Info] uuid for copy={copy.uuid}" />
</target>

<!-- call the targets in the order corresponding to the init, pre and post -->
<target name="job" depends="init,pre,copy" />
</project>

```

InvokeCD.xml script

InvokeCD.xml is called by both the Push_to_CD.xml and the Pull_from_CD.xml scripts. It builds a Connect:Direct process using a template and then invokes the process. Example D-3 shows the sample code used.

Example D-3 InvokeCD.xml

```
<?xml version="1.0" ?>
<project xmlns:fte="antlib:com.ibm.wmqfte.ant.taskdefs" name="InvokeCD"
default="execCD" basedir="C:\Documents and Settings\fteadmin">
  <target name="execCD" description="Invoke Connect:Direct">
    <copy file="${template}" tofile="${UUID}.cdp" />
    <replace file="${UUID}.cdp" token="#{srcfile}" value="${srcfile}" />
    <replace file="${UUID}.cdp" token="#{dstfile}" value="${dstfile}" />
    <replace file="${UUID}.cdp" token="#{jobNumber}" value="${jobNumber}" />
    <!-- Now invoke the "direct" command passing in the unique template and
capturing the output -->
    <exec dir="C:\Documents and Settings\fteadmin" executable="C:\Program
Files\Sterling Commerce\Connect Direct v4.5.01\Common Utilities\Direct.exe"
input="${UUID}.cdp" output="output.log" error="error.log"
outputproperty="cdOutput" logerror="true" errorproperty="cdEOutput" />

    <delete file="${UUID}.cdp" />
    <echo message="${cdOutput}" />
  </target>
</project>
```

Sample files used in WebSphere MQ File Transfer Edition within Sterling File Gateway

The Sterling B2B Integrator customization files were written to demonstrate interoperability between Sterling File Gateway and WebSphere MQ File Transfer Edition. They have been tested in the basic scenarios used, but the code is given as is. We suggest thoroughly analyzing the solution to ensure that it meets the requirements of your setup, and ensuring the solution is fully tested before it is used in any production environment.

SFGFTECreateTransfer.xslt file

The sample code in Example D-4 is the XSD Schema file that defines the structure of the XML file transfer message used in 6.3.10, "Creating custom WebSphere MQ File Transfer Edition protocol" on page 199.

Example D-4 XSD schema file to define the structure of the XML file transfer request message

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:output method="xml" version="1.0" encoding="UTF-8" indent="yes"/>
  <xsl:template match="/">
    <request xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="4.00"
xsi:noNamespaceSchemaLocation="FileTransfer.xsd">
```

```

<managedTransfer>
  <originator>
    <hostName>
      <xsl:value-of select="/ProcessData/hostName/text()"/>
    </hostName>
    <userID>
      <xsl:value-of select="/ProcessData/ConsumerName/text()"/>
    </userID>
  </originator>
  <sourceAgent>
    <xsl:attribute name="QMgr"><xsl:value-of
select="/ProcessData/SOURCEQM/text()"/></xsl:attribute>
    <xsl:attribute name="agent"><xsl:value-of
select="/ProcessData/SOURCEAGENT/text()"/></xsl:attribute>
  </sourceAgent>
  <destinationAgent>
    <xsl:attribute name="QMgr"><xsl:value-of
select="/ProcessData/DESTQM/text()"/></xsl:attribute>
    <xsl:attribute name="agent"><xsl:value-of
select="/ProcessData/DESTAGENT/text()"/></xsl:attribute>
  </destinationAgent>
  <reply>
    <xsl:attribute name="QMGR"><xsl:value-of
select="/ProcessData/REPLYQM/text()"/></xsl:attribute>
    <xsl:value-of select="/ProcessData/REPLYQUEUE/text()"/>
  </reply>
  <transferSet>
    <xsl:attribute name="priority"><xsl:value-of
select="/ProcessData/PRIORITY/text()"/></xsl:attribute>
    <metaDataSet>
      <metaData>
        <xsl:attribute name="key">RouteDataFlowId</xsl:attribute>
        <xsl:value-of select="/ProcessData/RouteDataflowId/text()"/>
      </metaData>
      <metaData>
        <xsl:attribute name="key">RouteId</xsl:attribute>
        <xsl:value-of select="/ProcessData/RouteID/text()"/>
      </metaData>
      <metaData>
        <xsl:attribute name="key">DeliveryBPID</xsl:attribute>
        <xsl:value-of select="/ProcessData/FG/WorkflowId/text()"/>
      </metaData>
      <metaData>
        <xsl:attribute name="key">DeliveryKey</xsl:attribute>
        <xsl:value-of select="/ProcessData/FG/DeliveryKey/text()"/>
      </metaData>
    </metaDataSet>
    <item>
      <xsl:attribute name="checksumMethod"><xsl:value-of
select="/ProcessData/CHECKSUMMETHOD/text()"/></xsl:attribute>
      <xsl:attribute name="mode"><xsl:value-of
select="/ProcessData/CONVERSION/text()"/></xsl:attribute>
      <source>
        <xsl:attribute name="disposition">delete</xsl:attribute>
        <xsl:attribute name="recursive">>false</xsl:attribute>
      </source>
    </item>
  </transferSet>

```

```

        <file><xsl:value-of
select="/ProcessData/FTE/MessageNameAndPath/text()"/></file>
    </source>
    <destination>
        <xsl:attribute name="exist"><xsl:value-of
select="/ProcessData/DESTFILEEXISTS/text()"/></xsl:attribute>
        <xsl:attribute name="type">file</xsl:attribute>
        <file><xsl:value-of
select="/ProcessData/DESTDIR/text()"/></xsl:value-of
select="/ProcessData/DestinationMessageName/text()"/></file>
    </destination>
</item>
</transferSet>
</managedTransfer>
</request>
</xsl:template>
</xsl:stylesheet>

```

CustomFileGatewayDeliveryFTE.bmpl file

The sample code shown in Example D-5 is a custom business process, which is invoked to initiate the transfer to route a file from Sterling File Gateway using WebSphere MQ File Transfer Edition, as defined in 6.3.10, “Creating custom WebSphere MQ File Transfer Edition protocol” on page 199.

This business process demonstrates techniques for generating transfer requests. However, it is not intended for use in a production/high-concurrency environment. When a reply queue is specified, waiting for transfer status reply messages is synchronous and concurrency will be limited to the number of threads configured for the business process queue (see Sterling Integrator documentation). A production/high-concurrency version of the business process would avoid extended blocking on the reply queue and would allow business process threads to service many requests.

Example D-5 CustomFileGatewayDeliveryFTE.bmpl

```

<process name="CustomFileGatewayDeliveryFTE">
  <rule name="isReplyQueueSpecified?">
    <condition>boolean(REPLYQUEUE)</condition>
  </rule>
  <rule name="isReplySourceAcknowledgement?">
    <condition>MQ/RetrievedReply/reply/status/@resultCode = "-2"</condition>
  </rule>
  <rule name="isReplySuccess?">
    <condition>MQ/RetrievedReply/reply/status/@resultCode = "0"</condition>
  </rule>
  <rule name="isReplyFailure?">
    <condition>MQ/RetrievedReply/reply/status/@resultCode != "-2"</condition>
    <condition>MQ/RetrievedReply/reply/status/@resultCode != "0"</condition>
  </rule>
  <rule name="wasTimeout?">
    <condition>not(boolean(MQ/RetrievedReply/reply/status))</condition>
  </rule>

```



```

<rule name="isPasswordSet?">
  <condition>boolean(SOURCEQMPASSWORD)</condition>
</rule>
<rule name="mqSessionExists?">
  <condition>boolean(wsmq_sessionid)</condition>
</rule>
<sequence name="fteCreateTransfer">
  <sequence name="CreateMessageForFTEAgent">
    <assign to="FTE/MessageSuffix" from="concat('_FTE_', FG/DeliveryKey)"/>
    <assign to="FTE/MessageName" from="concat(DestinationMessageName,
FTE/MessageSuffix)"/>
    <assign to="FTE/MessageNameAndPath" from="concat(FG/MailboxPath, '/',
FTE/MessageName)"/>
    <operation name="Mailbox Add Service">
      <participant name="MailboxAdd"/>
      <output message="MailboxAddServiceTypeInputMessage">
        <assign to="DocumentId" from="PrimaryDocumentId/text()"/>
        <assign to="MessageName" from="FTE/MessageName/text()"/>
        <assign to="MailboxPath" from="FG/MailboxPath/text()"/>
        <assign to="Extractable">YES</assign>
        <assign to="." from="*" />
      </output>
      <input message="inmsg">
        <assign to="FG/FTEAddResults" from="*" append="false"/>
      </input>
    </operation>
    <onFault>
      <sequence>
        <operation name="FileGatewayRouteEventService">
          <participant name="FileGatewayRouteEventService"/>
          <output message="Xout">
            <assign to="." from="RouteEntityKey"/>
            <assign to="." from="RouteEntityType"/>
            <assign to="." from="RouteDataflowId"/>
            <assign to="." from="RouteMetadata"/>
            <assign to="EventCode">FTE_0794</assign>
            <assign to="ExceptionLevel">Normal</assign>
            <!-- Set attributes for custom events for FTE transfers -->
            <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
            <assign to="EventAttributes/DestinationAgent"
from="DESTAGENT/text()"/>
            <assign to="EventAttributes/Failure" from="Unable to add
message to consumer mailbox"/>
            <assign to="." from="*" />
          </output>
          <input message="Input">
            <assign to="." from="*" />
          </input>
        </operation>
        <operation name="generateException">
          <participant name="BPEExceptionService"/>
          <output message="Xout">
            <assign to="exceptionCode">FTEHandledError</assign>
            <assign to="." from="*" />
          </output>
        </operation>
      </sequence>
    </onFault>
  </sequence>

```

```

        </output>
        <input message="Xin">
            <assign to="." from="*" />
        </input>
    </operation>
</sequence>
</onFault>
</sequence>
<sequence name="buildTransferRequestDocument">
    <assign to="hostName" from="sci-get-property('noapp','admin_host.1')"/>
    <!-- Needed in XSLT to set originator values -->
    <operation name="XSLTService">
        <participant name="XSLTService"/>
        <output message="XSLTServiceInputMessage">
            <assign to="xml_input_from">ProcData</assign>
            <assign to="input_pd_xpath">/ProcessData</assign>
            <assign to="xml_input_validation">NO</assign>
            <assign to="xslt_name">SFGFTECreateTransfer</assign>
            <assign to="." from="*" />
        </output>
        <input message="inmsg">
            <assign to="FTEManagedTransferDoc" from="PrimaryDocument"/>
        </input>
    </operation>
    <onFault>
        <sequence>
            <operation name="FileGatewayRouteEventService">
                <participant name="FileGatewayRouteEventService"/>
                <output message="Xout">
                    <assign to="." from="RouteEntityKey"/>
                    <assign to="." from="RouteEntityType"/>
                    <assign to="." from="RouteDataflowId"/>
                    <assign to="." from="RouteMetadata"/>
                    <assign to="EventCode">FTE_0794</assign>
                    <assign to="ExceptionLevel">Abnormal</assign>
                    <!-- Set attributes for custom events for FTE transfers -->
                    <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
                    <assign to="EventAttributes/DestinationAgent"
from="DESTAGENT/text()" />
                    <assign to="EventAttributes/Failure" from="Failed to create
managedTransfer XML document"/>
                    <assign to="." from="*" />
                </output>
                <input message="Input">
                    <assign to="." from="*" />
                </input>
            </operation>
            <operation name="generateException">
                <participant name="BPEExceptionService"/>
                <output message="Xout">
                    <assign to="exceptionCode">FTEHandledError</assign>
                    <assign to="." from="*" />
                </output>
                <input message="Xin">

```

```

        <assign to="." from="*" />
    </input>
</operation>
</sequence>
</onFault>
</sequence>
<sequence name="MQ Operations">
    <!-- Build variables needed: queue name, msg id, etc -->
    <sequence name="Prior To Transfer Request Submission">
        <assign to="MQ/SrcAgentCommandQueue"
from="concat('SYSTEM.FTE.COMMAND.', SOURCEAGENT)"/>
        <assign to="MQ/MsgId" from="FG/DeliveryKey/text()"/>
        <choice>
            <select>
                <case ref="isPasswordSet?" activity="OpenSessionWithPassword"/>
                <case ref="isPasswordSet?" negative="true"
activity="OpenSessionWithoutPassword"/>
            </select>
            <!-- open mq session, open command queue and reply queue -->
            <operation name="OpenSessionWithoutPassword">
                <participant name="WSMQ_OpenSession"/>
                <output message="WSMQOpenSessionInputMessage">
                    <assign to="wsmq_hostname" from="SOURCEQMHOST/text()"/>
                    <assign to="wsmq_port" from="SOURCEQMPORT/text()"/>
                    <assign to="wsmq_channel">SYSTEM.AUTO.SVRCONN</assign>
                    <assign to="wsmq_userid" from="SOURCEQMUSERID/text()"/>
                    <assign to="wsmq_qmanager" from="SOURCEQM/text()"/>
                    <assign to="." from="*" />
                </output>
                <input message="inmsg">
                    <assign to="." from="*" />
                </input>
            </operation>
            <operation name="OpenSessionWithPassword">
                <participant name="WSMQ_OpenSession"/>
                <output message="WSMQOpenSessionInputMessage">
                    <assign to="wsmq_hostname" from="SOURCEQMHOST/text()"/>
                    <assign to="wsmq_port" from="SOURCEQMPORT/text()"/>
                    <assign to="wsmq_channel">SYSTEM.AUTO.SVRCONN</assign>
                    <assign to="wsmq_userid" from="SOURCEQMUSERID/text()"/>
                    <assign to="wsmq_password"
from="revealObscured(SOURCEQMPASSWORD)"/>
                    <assign to="wsmq_qmanager" from="SOURCEQM/text()"/>
                    <assign to="." from="*" />
                </output>
                <input message="inmsg">
                    <assign to="." from="*" />
                </input>
            </operation>
        </choice>
        <operation name="WebSphereMQ Suite Open Agent Command Queue">
            <participant name="WSMQ_OpenQueue"/>
            <output message="WSMQOpenQueueInputMessage">
                <assign to="wsmq_qname" from="MQ/SrcAgentCommandQueue/text()"/>
                <assign to="wsmq_MQOO_failifquiescing">Yes</assign>
            </output>
        </operation>
    </sequence>
</sequence>

```

```

        <assign to="wsmq_MQ00_type">PUT</assign>
        <assign to="." from="*" />
    </output>
</operation>
<input message="inmsg">
    <assign to="." from="*" />
</input>
</operation>
<choice>
    <select>
        <case ref="isReplyQueueSpecified?" activity="WebSphereMQ Suite
Open Transfer Reply Queue" />
    </select>
    <operation name="WebSphereMQ Suite Open Transfer Reply Queue">
        <participant name="WSMQ_OpenQueue" />
        <output message="WSMQOpenQueueInputMessage">
            <assign to="wsmq_qname" from="REPLYQUEUE/text()" />
            <assign to="wsmq_MQ00_failifquiescing">Yes</assign>
            <assign to="wsmq_MQ00_type">GET</assign>
            <assign to="." from="*" />
        </output>
        <input message="inmsg">
            <assign to="." from="*" />
        </input>
    </operation>
</choice>
<operation name="WebSphereMQ Suite Put Message">
    <participant name="WSMQ_PutMessage" />
    <output message="WSMQPutMessageInputMessage">
        <assign to="." from="*" />
        <assign to="wsmq_document"
from="FTEManagedTransferDoc/PrimaryDocument/@*" />
        <assign to="wsmq_MQMD_msgId" from="MQ/MsgId/text()" />
        <assign to="wsmq_MQMD_priority" from="PRIORITY/text()" />
        <!-- necessary or will agent handle this?-->
        <assign to="wsmq_qname" from="MQ/SrcAgentCommandQueue/text()" />
    </output>
    <input message="inmsg">
        <assign to="." from="*" />
    </input>
</operation>
<operation name="WebSphereMQ Suite Commit">
    <participant name="WSMQ_Commit" />
    <output message="WSMQCommitInputMessage">
        <assign to="." from="*" />
    </output>
    <input message="inmsg">
        <assign to="." from="*" />
    </input>
</operation>
<operation name="FileGatewayRouteEventService">
    <participant name="FileGatewayRouteEventService" />
    <output message="Xout">
        <assign to="." from="RouteEntityKey" />
        <assign to="." from="RouteEntityType" />
        <assign to="." from="RouteDataflowId" />
    </output>

```

```

        <assign to="." from="RouteMetadata"/>
        <assign to="EventCode">FTE_0745</assign>
        <assign to="ExceptionLevel">Abnormal</assign>
        <!-- Set attributes for custom events for FTE transfers -->
        <assign to="EventAttributes/TransferId" from="MQ/MsgId/text()"
append="true"/>
        <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
        <assign to="." from="*" />
    </output>
    <input message="Input">
        <assign to="." from="*" />
    </input>
</operation>
<onFault>
    <sequence>
        <operation name="FileGatewayRouteEventService">
            <participant name="FileGatewayRouteEventService" />
            <output message="Xout">
                <assign to="." from="RouteEntityKey" />
                <assign to="." from="RouteEntityType" />
                <assign to="." from="RouteDataflowId" />
                <assign to="." from="RouteMetadata" />
                <assign to="EventCode">FTE_0794</assign>
                <assign to="ExceptionLevel">Abnormal</assign>
                <!-- Set attributes for custom events for FTE transfers -->
                <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
                <assign to="EventAttributes/DestinationAgent"
from="DESTAGENT/text()" />
                <assign to="EventAttributes/Failure" from="concat('Failure
in communication with Source Agent Queue Manager: ', ERROR_SERVICE/ADV_STATUS)"/>
                <assign to="." from="*" />
            </output>
            <input message="Input">
                <assign to="." from="*" />
            </input>
        </operation>
        <operation name="generateException">
            <participant name="BPExceptionService" />
            <output message="Xout">
                <assign to="exceptionCode">FTEHandledError</assign>
                <assign to="." from="*" />
            </output>
            <input message="Xin">
                <assign to="." from="*" />
            </input>
        </operation>
    </sequence>
</onFault>
</sequence>
<choice>
    <select>
        <case ref="isReplyQueueSpecified?" activity="retrieveReply"/>

```

```

        <case ref="isReplyQueueSpecified?" activity="assumeSuccess"
negative="true"/>
    </select>
    <sequence name="retrieveReply">
        <operation name="GET message(s)">
            <participant name="WSMQ_GetMessage"/>
            <output message="toService">
                <assign to="wsmq_sessionid" from="string(wsmq_sessionid)"/>
                <assign to="wsmq_qname" from="REPLYQUEUE/text()"/>
                <assign to="wsmq_MQMO_corId" from="MQ/MsgId/text()"/>
                <assign to="wsmq_metadata1">4095</assign>
                <assign to="wsmq_metadata2">4095</assign>
                <assign to="wsmq_type">GETONE</assign>
                <assign to="wsmq_MQGMO_wait">Yes</assign>
                <assign to="wsmq_MQGMO_waitInterval"
from="number(TransferTimeOut) * 1000"/>
            </output>
            <input message="fromService">
                <assign to="MQ/RetrievedReply" from="DocToDOM(WSMQ/Document1,
'false', 'false')"/>
            </input>
        </operation>
        <!-- retrieveReply -->
        <choice name="ActOnReply">
            <select>
                <case ref="isReplySourceAcknowledgement?"
activity="RepeatRetrieveReply"/>
                <case ref="isReplySuccess?" activity="SuccessfulReply"/>
                <case ref="isReplyFailure?" activity="ErrorReply"/>
                <case ref="wasTimeout?" activity="Timeout"/>
            </select>
            <sequence name="RepeatRetrieveReply">
                <operation name="FileGatewayRouteEventService">
                    <participant name="FileGatewayRouteEventService"/>
                    <output message="Xout">
                        <assign to="." from="RouteEntityKey"/>
                        <assign to="." from="RouteEntityType"/>
                        <assign to="." from="RouteDataflowId"/>
                        <assign to="." from="RouteMetadata"/>
                        <assign to="EventCode">FTE_0746</assign>
                        <assign to="ExceptionLevel">Normal</assign>
                        <!-- Set attributes for custom events for FTE transfers
-->
                        <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
                        <assign to="EventAttributes/ResultCode"
from="string(MQ/RetrievedReply/reply/status/@resultCode)"/>
                        <assign to="." from="*" />
                    </output>
                    <input message="Input">
                        <assign to="." from="*" />
                    </input>
                </operation>
                <operation name="Remove Reply">
                    <participant name="ReleaseService"/>

```

```

        <output message="releaseMessage">
            <assign
to="TARGET">/ProcessData/MQ/RetrievedReply</assign>
            </output>
            <input message="inmsg"/>
        </operation>
        <repeat ref="retrieveReply"/>
    </sequence>
    <sequence name="SuccessfulReply">
        <assign to="SUCCESSREPLY">true</assign>
    </sequence>
    <sequence name="ErrorReply">
        <operation name="FileGatewayRouteEventService">
            <participant name="FileGatewayRouteEventService"/>
            <output message="Xout">
                <assign to="." from="RouteEntityKey"/>
                <assign to="." from="RouteEntityType"/>
                <assign to="." from="RouteDataflowId"/>
                <assign to="." from="RouteMetadata"/>
                <assign to="EventCode">FTE_0795</assign>
                <assign to="ExceptionLevel">Abnormal</assign>
                <!-- Set attributes for custom events for FTE transfers
-->
                <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
                <assign to="EventAttributes/DestinationAgent"
from="DESTAGENT/text()"/>
                <assign to="EventAttributes/TransferId"
from="MQ/MsgId/text()"/>
                <assign to="EventAttributes/ResultCode"
from="string(MQ/RetrievedReply/reply/status/@resultCode)"/>
                <assign to="." from="*"/>
            </output>
            <input message="Input">
                <assign to="." from="*"/>
            </input>
        </operation>
        <operation name="generateException">
            <participant name="BPExceptionService"/>
            <output message="Xout">
                <assign to="exceptionCode">FTEHandledError</assign>
                <assign to="." from="*"/>
            </output>
            <input message="Xin">
                <assign to="." from="*"/>
            </input>
        </operation>
    </sequence>
    <sequence name="Timeout">
        <operation name="FileGatewayRouteEventService">
            <participant name="FileGatewayRouteEventService"/>
            <output message="Xout">
                <assign to="." from="RouteEntityKey"/>
                <assign to="." from="RouteEntityType"/>
                <assign to="." from="RouteDataflowId"/>

```

```

        <assign to="." from="RouteMetadata"/>
        <assign to="EventCode">FTE_0796</assign>
        <assign to="ExceptionLevel">Abnormal</assign>
        <!-- Set attributes for custom events for FTE transfers
-->
        <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
        <assign to="EventAttributes/DestinationAgent"
from="DESTAGENT/text()"/>
        <assign to="EventAttributes/TransferId"
from="MQ/MsgId/text()"/>
        <assign to="." from="*" />
    </output>
    <input message="Input">
        <assign to="." from="*" />
    </input>
</operation>
<operation name="generateException">
    <participant name="BPExceptionService"/>
    <output message="Xout">
        <assign to="exceptionCode">FTEHandledError</assign>
        <assign to="." from="*" />
    </output>
    <input message="Xin">
        <assign to="." from="*" />
    </input>
</operation>
</sequence>
</choice>
<operation name="WebSphereMQ Suite Close Reply Queue">
    <participant name="WSMQ_CloseQueue"/>
    <output message="WSMQCloseQueueInputMessage">
        <assign to="." from="*" />
        <assign to="wsmq_qname" from="REPLYQUEUE/text()"/>
    </output>
    <input message="inmsg">
        <assign to="." from="*" />
    </input>
</operation>
</sequence>
<sequence name="assumeSuccess">
    <operation name="FileGatewayRouteEventService">
        <participant name="FileGatewayRouteEventService"/>
        <output message="Xout">
            <assign to="." from="RouteEntityKey"/>
            <assign to="." from="RouteEntityType"/>
            <assign to="." from="RouteDataflowId"/>
            <assign to="." from="RouteMetadata"/>
            <assign to="EventCode">FTE_0747</assign>
            <assign to="ExceptionLevel">Normal</assign>
            <!-- Set attributes for custom events for FTE transfers -->
            <assign to="EventAttributes/TransferId"
from="MQ/MsgId/text()" append="true"/>
            <assign to="." from="*" />
        </output>

```



```

        <input message="Input">
            <assign to="." from="*" />
        </input>
    </operation>
</sequence>
</choice>
<operation name="WebSphereMQ Suite Close Queue">
    <participant name="WSMQ_CloseQueue" />
    <output message="WSMQCloseQueueInputMessage">
        <assign to="." from="*" />
        <assign to="wsmq_qname" from="MQ/SrcAgentCommandQueue/text()" />
    </output>
    <input message="inmsg">
        <assign to="." from="*" />
    </input>
</operation>
<operation name="WebSphereMQ Suite Close Session">
    <participant name="WSMQ_CloseSession" />
    <output message="WSMQCloseSessionInputMessage">
        <assign to="." from="*" />
    </output>
    <input message="inmsg">
        <assign to="." from="*" />
    </input>
</operation>
</sequence>
<!-- put manageTransferMessage -->
<!-- get reply message(s) and parse results -->
<!-- close queues and session -->
<operation name="FileGatewayRouteEventService">
    <participant name="FileGatewayRouteEventService" />
    <output message="Xout">
        <assign to="." from="RouteEntityKey" />
        <assign to="." from="RouteEntityType" />
        <assign to="." from="RouteDataflowId" />
        <assign to="." from="RouteMetadata" />
        <assign to="EventCode">FTE_0744</assign>
        <assign to="ExceptionLevel">Normal</assign>
        <!-- Set attributes for custom events for FTE transfers -->
        <assign to="EventAttributes/SourceAgent" from="SOURCEAGENT/text()"
append="true" />
        <assign to="EventAttributes/DestinationAgent"
from="DESTAGENT/text()" />
        <assign to="EventAttributes/TransferId" from="MQ/MsgId/text()" />
        <assign to="." from="*" />
    </output>
    <input message="Input">
        <assign to="." from="*" />
    </input>
</operation>
<onFault code="FTEHandledError">
    <!-- this is for errors raised within this BP for which an SFG event has
already been raised -->
    <sequence>
        <choice>

```

```

        <select>
            <case ref="mqSessionExists?" activity="closeSession"/>
        </select>
        <operation name="closeSession">
            <participant name="WSMQ_CloseSession"/>
            <output message="WSMQCloseSessionInputMessage">
                <assign to="." from="*" />
            </output>
            <input message="inmsg">
                <assign to="." from="*" />
            </input>
        </operation>
    </choice>
    <operation name="generateException">
        <participant name="BPExceptionService"/>
        <output message="Xout">
            <assign to="exceptionCode">ERROR, WMQFTE TRANSFER ATTEMPT FAILED
</assign>
            <assign to="." from="*" />
        </output>
        <input message="Xin">
            <assign to="." from="*" />
        </input>
    </operation>
</sequence>
</onFault>
<onFault>
    <!-- General error handler, need to raise SFG failure event -->
    <sequence>
        <operation name="FileGatewayRouteEventService">
            <participant name="FileGatewayRouteEventService"/>
            <output message="Xout">
                <assign to="." from="RouteEntityKey"/>
                <assign to="." from="RouteEntityType"/>
                <assign to="." from="RouteDataflowId"/>
                <assign to="." from="RouteMetadata"/>
                <assign to="EventCode">FTE_0799</assign>
                <assign to="ExceptionLevel">Abnormal</assign>
                <!-- Set attributes for custom events for FTE transfers -->
                <assign to="EventAttributes/SourceAgent"
from="SOURCEAGENT/text()" append="true"/>
                <assign to="EventAttributes/DestinationAgent"
from="DESTAGENT/text()" />
                <assign to="EventAttributes/TransferId" from="MQ/MsgId/text()" />
                <assign to="." from="*" />
            </output>
            <input message="Input">
                <assign to="." from="*" />
            </input>
        </operation>
    </choice>
    <select>
        <case ref="mqSessionExists?" activity="closeSession"/>
    </select>
    <operation name="closeSession">

```

```

        <participant name="WSMQ_CloseSession"/>
        <output message="WSMQCloseSessionInputMessage">
            <assign to="." from="*" />
        </output>
        <input message="inmsg">
            <assign to="." from="*" />
        </input>
        </operation>
    </choice>
    <operation name="generateException">
        <participant name="BPEXceptionService"/>
        <output message="Xout">
            <assign to="exceptionCode">ERROR, WMQFTE TRANSFER ATTEMPT FAILED
</assign>
            <assign to="." from="*" />
        </output>
        <input message="Xin">
            <assign to="." from="*" />
        </input>
        </operation>
    </sequence>
</onFault>
</sequence>
</process>

```

Customer_overrides.properties file

The sample code in Example D-6 defines custom Sterling File Gateway events that are used in the CustomFileGatewayDeliveryFTE business process described in 6.3.10, “Creating custom WebSphere MQ File Transfer Edition protocol” on page 199.

Example D-6 Customer_overrides.properties file

```

filegateway_eventcodes.FTE_0744.name=WMQFTEFileTransferComplete
filegateway_eventcodes.FTE_0744.attributes=SourceAgent, DestinationAgent, TransferId
filegateway_eventcodes.FTE_0744.text=WMQFTE Transfer from agent {0} to agent {1}
with Transfer Id {2} completed successfully
filegateway_eventcodes.FTE_0744.description=Event from Custom FTE Listening
Protocol when transfer completed
filegateway_eventcodes.FTE_0744.permissions=producer, consumer, subscription

filegateway_eventcodes.FTE_0745.name=WMQFTEFileTransferRequestSubmitted
filegateway_eventcodes.FTE_0745.attributes=TransferId, SourceAgent
filegateway_eventcodes.FTE_0745.text=Transfer request with Id {0} submitted to
agent {1}
filegateway_eventcodes.FTE_0745.description=Event from Custom FTE Listening
Protocol indicating request successfully put to agent queue
filegateway_eventcodes.FTE_0745.permissions=producer, consumer, subscription

filegateway_eventcodes.FTE_0746.name=WMQFTETransferRequestAcknowledged
filegateway_eventcodes.FTE_0746.attributes=SourceAgent, ResultCode
filegateway_eventcodes.FTE_0746.text=Source Agent {0} Acknowledged Transfer
Request with ResultCode {1}

```

filegateway_eventcodes.FTE_0746.description=Indicates the source agent sent a reply acknowledging receipt of transfer request
filegateway_eventcodes.FTE_0746.permissions=producer,consumer,subscription

filegateway_eventcodes.FTE_0747.name=WMQFTERepliesDisabled
filegateway_eventcodes.FTE_0747.attributes=TransferId
filegateway_eventcodes.FTE_0747.text=No reply queue specified, assuming successful delivery of {0} without confirmation
filegateway_eventcodes.FTE_0747.description=SFG will not request a reply to WMQFTE transfer request and will assume delivery was successful without confirmation
filegateway_eventcodes.FTE_0747.permissions=producer,consumer,subscription

filegateway_eventcodes.FTE_0794.name=WMQFTEFileTransferFailedRequestNotSubmitted
filegateway_eventcodes.FTE_0794.attributes=SourceAgent,DestinationAgent,Failure
filegateway_eventcodes.FTE_0794.text=WMQFTE Transfer from agent {0} to agent {1} was not submitted: {2}
filegateway_eventcodes.FTE_0794.description=Event from Custom FTE Listening Protocol when transfer has failed and no request was submitted to WMQFTE
filegateway_eventcodes.FTE_0794.permissions=producer,consumer,subscription

filegateway_eventcodes.FTE_0795.name=WMQFTEFileTransferFailedWMQFTEIndicatedFailure
filegateway_eventcodes.FTE_0795.attributes=SourceAgent,DestinationAgent,TransferId,ResultCode
filegateway_eventcodes.FTE_0795.text=WMQFTE Transfer from agent {0} to agent {1} failed with ResultCode {3} in reply from WMQFTE
filegateway_eventcodes.FTE_0795.description=Event from Custom FTE Listening Protocol when transfer has failed as indicated by reply from WMQFTE
filegateway_eventcodes.FTE_0795.permissions=producer,consumer,subscription

filegateway_eventcodes.FTE_0796.name=WMQFTETimeoutWaitingOnReplyDeliveryStatusUnknown
filegateway_eventcodes.FTE_0796.attributes=SourceAgent,DestinationAgent,TransferId
filegateway_eventcodes.FTE_0796.text=WMQFTE Transfer from agent {0} to agent {1} submitted, but delivery status reply not received prior to timeout. Pessimistically assuming failure, check WMQFTE for ultimate delivery status, Transfer Id {2}
filegateway_eventcodes.FTE_0796.description=Event from Custom FTE Listening Protocol when transfer did not result in reply from WMQFTE prior to timeout.
filegateway_eventcodes.FTE_0796.permissions=producer,consumer,subscription

filegateway_eventcodes.FTE_0799.name=WMQFTEGeneralError
filegateway_eventcodes.FTE_0799.attributes=SourceAgent,DestinationAgent,TransferId
filegateway_eventcodes.FTE_0799.text=WMQFTE Transfer from agent {0} to agent {1} with Transfer Id {2} encountered an error
filegateway_eventcodes.FTE_0799.description=Event from Custom FTE Listening Protocol indicating a general error not handled elsewhere with a more specific event
filegateway_eventcodes.FTE_0799.permissions=producer,consumer,subscription

AFTEExtensionsCustomer.properties source file

The sample code in Example D-7 defines input parameters required for the new protocol. See 6.3.10, “Creating custom WebSphere MQ File Transfer Edition protocol” on page 199, for details.

Example D-7 AFTEExtensionsCustomer.properties file

```
#####
# CUSTOM Websphere MQ FTE parameters
#####
fte.protocol.label.fteprotocol = WebSphere MQ FTE
fte.instance.group1.title = WebSphere MQ FTE Parameters
fte.label.fteprotocol.SOURCEAGENT = Source Agent Name (-sa)
fte.label.fteprotocol.SOURCEQM = Source Agent Queue Manager (-sm)
fte.label.fteprotocol.SOURCEQMHOST=Source Agent Queue Manager Host Name
fte.label.fteprotocol.SOURCEQMPORT=Source Agent Queue Manager Port
fte.label.fteprotocol.SOURCEQMUSERID=Source Agent Queue Manager User Id
fte.label.fteprotocol.SOURCEQMPASSWORD=Source Agent Queue Manager Password
fte.label.fteprotocol.DESTAGENT = Destination Agent Name (-da)
fte.label.fteprotocol.DESTQM = Destination Agent Queue Manager (-dm)
fte.label.fteprotocol.DESTDIR = Destination Agent's Directory (-dd)
fte.label.fteprotocol.DESTFILEEXISTS = Destination File Already Exists (-de)
fte.label.fteprotocol.REPLYQUEUE = Queue For Transfer Status Reply Messages
fte.label.fteprotocol.PRIORITY = Priority (-pr)
fte.label.fteprotocol.CONVERSION = Conversion (-t)
fte.label.fteprotocol.CHECKSUMMETHOD = Checksum Method (-cs)
fte.label.fteprotocol.TransferTimeout = Transfer Timeout (seconds)

#####
# CUSTOM Custom labels for OPTION Elements
#####
custom.error=error
custom.override=overwrite
custom.zero=0
custom.one=1
custom.two=2
custom.three=3
custom.four=4
custom.five=5
custom.six=6
custom.seven=7
custom.eight=8
custom.nine=9
custom.binary=binary
custom.text=text
custom.md5=MD5
custom.none=none
```

AFTExtensionsCustomer.xml source file

The sample code in Example D-8 is defined and used in 6.3.10, “Creating custom WebSphere MQ File Transfer Edition protocol” on page 199.

Example D-8 AFTExtensionsCustomer.xml

```
<!--
    Taking care to backup any files that may get displaced, save this
    file into the following two directories:

<install-dir>/container/Applications/aft/WEB-INF/classes/resources/xml
<install-dir>/container/Applications/myaft/WEB-INF/classes/resources/xml
-->

<AFTExtensions>
### WEBSHERE FTE LISTENING PROTOCOL ###
  <AFTExtension name="fte-protocol" type="consumer-delivery-protocol"
label="fte.protocol.label.fteprotocol" bp="CustomFileGatewayDeliveryFTE">
  <GROUP title="fte.instance.group1.title">
<VARDEF varname="SOURCEAGENT" type="String" htmlType="text"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.SOURCEAGENT" required="yes" />
<VARDEF varname="SOURCEQM" type="String" htmlType="text" validator="ALPHANUMERIC"
size="30" maxSize="250" label="fte.label.fteprotocol.SOURCEQM" required="yes" />
<VARDEF varname="SOURCEQMHOST" type="String" htmlType="text"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.SOURCEQMHOST" required="no" />
<VARDEF varname="SOURCEQMPORT" type="String" htmlType="text"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.SOURCEQMPORT" required="no" />
<VARDEF varname="SOURCEQMUSERID" type="String" htmlType="text"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.SOURCEQMUSERID" required="no" />
<VARDEF varname="SOURCEQMPASSWORD" type="String" htmlType="password"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.SOURCEQMPASSWORD" required="no" />
<VARDEF varname="DESTAGENT" type="String" htmlType="text" validator="ALPHANUMERIC"
size="30" maxSize="250" label="fte.label.fteprotocol.DESTAGENT" required="yes" />
<VARDEF varname="DESTQM" type="String" htmlType="text" validator="ALPHANUMERIC"
size="30" maxSize="250" label="fte.label.fteprotocol.DESTQM" required="yes" />
<VARDEF varname="DESTDIR" type="String" htmlType="text" validator="ALPHANUMERIC"
size="30" maxSize="250" label="fte.label.fteprotocol.DESTDIR" required="yes" />
<VARDEF varname="DESTFILEEXISTS" type="String" htmlType="select"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.DESTFILEEXISTS" options="FTEDESTFILEEXISTS"
required="yes" defaultVal="error"/>
<VARDEF varname="REPLYQUEUE" type="String" htmlType="text"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.REPLYQUEUE" required="no" />
<VARDEF varname="PRIORITY" type="String" htmlType="select"
validator="ALPHANUMERIC" size="30" maxSize="250"
label="fte.label.fteprotocol.PRIORITY" options="FTEPRIORITY" required="yes"
defaultVal="0"/>

```

```

<VARDEF varname="CONVERSION" type="String" htmlType="select"
validator="ALPHANUMERIC" size="30" maxsize="250"
label="fte.label.fteprotocol.CONVERSION" options="FTECONVERSION" required="yes"
defaultVal="binary"/>
<VARDEF varname="CHECKSUMMETHOD" type="String" htmlType="select"
validator="ALPHANUMERIC" size="30" maxsize="250"
label="fte.label.fteprotocol.CHECKSUMMETHOD" options="FTECHECKSUMMETOHD"
required="yes" defaultVal="MD5"/>
<VARDEF varname="TransferTimeOut" type="String" htmlType="text"
validator="ALPHANUMERIC" size="30" maxsize="250"
label="fte.label.fteprotocol.TransferTimeOut" required="no" />

    </GROUP>
</AFTEExtension>

<OPTION name="FTEDESTFILEEXISTS">
    <ELE value="error" displayname="custom.error"/>
    <ELE value="overwrite" displayname="custom.overwrite"/>
</OPTION>
<OPTION name="FTEPRIORITY">
    <ELE value="0" displayname="custom.zero"/>
    <ELE value="1" displayname="custom.one"/>
    <ELE value="2" displayname="custom.two"/>
    <ELE value="3" displayname="custom.three"/>
    <ELE value="4" displayname="custom.four"/>
    <ELE value="5" displayname="custom.five"/>
    <ELE value="6" displayname="custom.six"/>
    <ELE value="7" displayname="custom.seven"/>
    <ELE value="8" displayname="custom.eight"/>
    <ELE value="9" displayname="custom.nine"/>
</OPTION>
<OPTION name="FTECONVERSION">
    <ELE value="binary" displayname="custom.binary"/>
    <ELE value="text" displayname="custom.text"/>
</OPTION>
<OPTION name="FTECHECKSUMMETOHD">
    <ELE value="MD5" displayname="custom.md5"/>
    <ELE value="none" displayname="custom.none"/>
</OPTION>
</AFTEExtensions>

```

Archived

Additional material

This book refers to additional material that can be downloaded from the internet as described in this appendix.

Locating the web material

The web material associated with this book is available in softcopy on the internet from the IBM Redbooks web server. Point your web browser to:

<ftp://www.redbooks.ibm.com/redbooks/SG24-7927-00>

Alternatively, you can go to the IBM Redbooks website at:

ibm.com/redbooks

Select **Additional materials** and open the directory that corresponds with the IBM Redbooks form number, SG247927.

Using the web material

The additional material that accompanies this book includes the following files:

- ▶ `AFTExtensionsCustomer.xml`

This file contains one or more `AFTExtension` elements, each defining a custom protocol. For further details see 6.3.3, “Sterling B2B Integrator and Sterling File Gateway customization” on page 182.

- ▶ `AFTExtensionsCustomer.properties`

This file contains the text strings that will be displayed in the Sterling File Gateway user interface when configuring a partner using a custom protocol. For further details see 6.3.3, “Sterling B2B Integrator and Sterling File Gateway customization” on page 182.

- ▶ CustomFileGatewayDeliveryFTE.bmp1

This file contains the implementation of the business process that creates a transfer request on the WebSphere MQ File Transfer Edition. For further details see 6.3.3, “Sterling B2B Integrator and Sterling File Gateway customization” on page 182.

- ▶ SFGFTECreateTransfer.xslt

This XSLT stylesheet is used by the CustomFileGatewayDeliveryFTE business process to build the XML document requesting an transfer, incorporating parameters specific to the current file and partner. For further details see 6.3.3, “Sterling B2B Integrator and Sterling File Gateway customization” on page 182.

- ▶ Customer_overrides.properties

This file defines custom Sterling File Gateway events that are used in the CustomFileGatewayDeliveryFTE business process.

You can use these files to build the sample scenarios that were used in Chapter 6, “External Transfers with Protocol Switching between IBM Sterling Connect:Direct and WebSphere MQ File Transfer Edition via Sterling File Gateway” on page 167.

Downloading and extracting the web material

Create a subdirectory (folder) on your workstation, and extract the contents of the SG247927.zip file into this folder.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks

The following IBM Redbooks publications provide additional information about the topic in this document. Note that some publications referenced in this list might be available in softcopy only.

- ▶ *Multi-Enterprise File Transfer with WebSphere Connectivity*, SG24-7886
- ▶ *Getting Started with WebSphere MQ File Transfer Edition V7*, SG24-7760

You can search for, view, or download Redbooks, Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy Redbooks publications, at this website:

ibm.com/redbooks

Online resources

These websites are also relevant as further information sources:

- ▶ Information about WebSphere Business Adapters
<http://www-01.ibm.com/software/integration/wbiadapters/>
- ▶ Customizing WebSphere MQ File Transfer Edition with user exit routines
http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.doc/user_exits.htm
- ▶ Using groups to manage authorities for resources specific to WebSphere File Transfer Edition
http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/group_resource_access.htm
- ▶ Information about WebSphere MQ File Transfer Edition and sandboxing
<http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/sandboxes.htm>
- ▶ WebSphere MQ File Transfer Edition commandPath property
http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/command_path.htm
- ▶ WebSphere MQ File Transfer Edition Information Center
<http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.doc/fteant.htm>
- ▶ Using groups to manage authorities for resources specific to WebSphere File Transfer Edition
http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/topic/com.ibm.wmqfte.admin.doc/group_resource_access.htm

- ▶ WebSphere MQ File Transfer Edition 7.0.3 Information Center
http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/index.jsp?topic=/com.ibm.wmqfte.home.doc/help_home_wmqfte.htmfirst
- ▶ WebSphere MQ Information Center as part of Tutorial 1: Sending a message to a local queue
http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/topic/com.ibm.mq.explorer.tutorials.doc/bi00257_.htm?resultof=%22%63%72%65%61%74%65%22%20%22%63%72%65%61%74%22%20%22%6c%6f%63%61%6c%22%20%22%71%75%65%75%65%22%20
- ▶ WebSphere MQ V7 Information Center
<http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>
- ▶ WebSphere Message Broker Information Center
<http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.samples.simplifieddbrouting.doc/doc/overview.htm>
- ▶ Instructions for building the sample
http://publib.boulder.ibm.com/infocenter/wmbhelp/v7r0m0/topic/com.ibm.etools.mft.samples.simplifieddbrouting.doc/doc/create_flow_simplifiedDBRouting.htm
- ▶ Detailed information about WebSphere MQ File Transfer Edition diagnostic messages
http://publib.boulder.ibm.com/infocenter/wmqfte/v7r0/index.jsp?topic=/com.ibm.wmqfte.messages.doc/messages_main.htm
- ▶ Detailed information about WebSphere MQ error codes
<http://publib.boulder.ibm.com/infocenter/wmqv7/v7r0/index.jsp>
- ▶ WebSphere MQ and WebSphere MQ File Transfer Edition system requirements
<http://www-01.ibm.com/software/integration/wmq/filetransfer/requirements>

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services



IBM Sterling Managed File Transfer Integration with WebSphere Connectivity for a Multi-Enterprise Solution

Archived



IBM Sterling Managed File Transfer Integration with WebSphere Connectivity for a Multi-Enterprise Solution



Using Sterling File Gateway and Sterling B2B Integrator for robust managed file transfer

This IBM Redbooks publication describes how IBM has enhanced its managed file transfer portfolio consisting of MQ File Transfer Edition with the Sterling Business Integration Suite. The Sterling Business Integration Suite consists of Sterling File Gateway and Sterling Connect:Direct. Sterling Commerce, an IBM company, transforms and optimizes your business collaboration network by improving business agility, efficiency, and performance.

Extending file transfer capabilities with WebSphere Message Broker

These managed file transfer components from Sterling Commerce, an IBM company, partnered with MQ File Transfer Edition deliver proven value by protecting privacy and integrity of data in transit with governance, eliminate operations cell center traffic regarding file transfer exceptions, show a faster time to revenue, and bring a six-sigma level performance to key business processes. The integration and combination of these products allows for organizations to switch between protocols internally, allowing for diversity across business needs while still positioning the organization to easily move files outside their secured intra-enterprise network through an edge server to the external trading partner regardless of what protocol the external trading partner is using.

Integrating Sterling Connect:Direct with WebSphere MQ File Transfer Edition

This book is intended for organizations that find themselves wanting to trade data in a secure, reliable, and auditable way across both intra-enterprise and multi-enterprise protocols.

**INTERNATIONAL
TECHNICAL
SUPPORT
ORGANIZATION**

**BUILDING TECHNICAL
INFORMATION BASED ON
PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:
ibm.com/redbooks**

SG24-7927-00

ISBN 0738435368