

SAP NetWeaver Process Integration 7.1



SAP NetWeaver Process Integration - Simple Use Cases

Document Version 1.0 - December 14, 2007



SAP AG

Dietmar-Hopp-Allee 16
69190 Walldorf
Germany
T +49/18 05/34 34 24
F +49/18 05/34 34 20
www.sap.com

© Copyright 2007 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, Informix, i5/OS, POWER, POWER5, OpenPower and PowerPC are trademarks or registered trademarks of IBM Corporation.

Adobe, the Adobe logo, Acrobat, PostScript, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

Disclaimer

Some components of this product are based on Java™. Any code change in these components may cause unpredictable and severe malfunctions and is therefore expressly prohibited, as is any decompilation of these components.

Any Java™ Source Code delivered with this product is only to be used by SAP's Support Services and may not be modified or altered in any way.

Documentation in SAP Developer Network (SDN)

You can find this documentation in SAP Developer Network (SDN) at

<https://www.sdn.sap.com/irj/sdn/nw-soa> → **Service Bus**. On the *Service Bus-Based Integration* page, select *Getting Started*.

1	Preface	4
2	Introduction and Prerequisites	4
2.1	Introduction.....	4
2.2	Prerequisites	6
2.2.1	General Prerequisites.....	6
2.2.2	Providing the necessary User and Authorizations.....	6
2.2.3	Special Prerequisites for Configuring the Variants 1 up to 4 and 6.....	7
2.2.4	Special Prerequisites for Configuring Variant 5.....	10
3	Variant 1: One Sender and One Receiver	14
3.1	Design Objects Used.....	14
3.2	Configuring the Process Integration Scenario	16
3.3	Execution.....	21
4	Variant 2: One Sender and Two Receivers	21
4.1	Design Objects Used.....	21
4.2	Configuring the Process Integration Scenario	22
4.3	Execution.....	23
5	Variant 3: One Sender and Two Receivers – with Routing Condition	23
5.1	Design Objects Used.....	23
5.2	Configuring the Process Integration Scenario	24
5.3	Execution.....	24
6	Variant 4: One Sender and Two Receivers – with Mapping ... 25	
6.1	Design Objects Used.....	25
6.2	Configuring the Process Integration Scenario	26
6.3	Execution.....	27
7	Variant 5: One Sender and One Receiver - Proxy-to-Proxy Communication	28
7.1	Design Objects Used.....	28
7.2	Configuring the Process Integration Scenario	29
7.3	Execution.....	30
8	Variant 6: One Sender and One Receiver using Advanced Adapter Engine	30
8.1	Design Objects Used.....	30
8.2	Configuring the Process Integration Scenario	31

1 Preface

Constraints

The texts, references, and graphics contained in this manual have been compiled with utmost care; nevertheless, it is impossible to guarantee that they are fully without error. SAP cannot assume any responsibility for the correctness or completeness of the following documentation; the user alone is responsible for verifying the information contained therein.

SAP will only assume liability for damage arising from the use of this documentation – irrespective of the pertinent legal basis – in the case of intentional or active negligence, under no other circumstances will a warranty be made.

Definition

This manual describes simple application cases for process integration and all the configuration steps that are necessary to execute the application cases on the basis of SAP NetWeaver Process Integration 7.1 (SAP NetWeaver PI 7.1).

This document is updated on a regular basis in SAP Developer Network (SDN) at <https://www.sdn.sap.com/irj/sdn/nw-soa> → *Service Bus-Based Integration* → *Getting Started*.

Intended Audience

This manual is intended to be used by both technology and application consultants.

Structure

The structure of this document follows the sequence of steps required to configure and run the use cases.

Additional Documentation

Additional Documentation

[Demo Examples](#) (contains introduction to demo examples and description of the process integration scenarios)

[Configuration of SAP NetWeaver Process Integration](#) (describes configuration steps required to set up SAP NetWeaver PI)

2 Introduction and Prerequisites

2.1 Introduction

Use cases are based on simple examples that enable you to learn the main features of SAP NetWeaver Process Integration. These instructions will enable you to understand, configure, and execute simple scenarios (that are based on fundamental communication patterns), as well as check that the scenarios are executed correctly. The prerequisite for this document is that you have a general understanding of SAP NetWeaver Process Integration.

The following use case variants are available at present:

Variant	Description
Variants 1, 5, and 6: One sender and one receiver (see 3, see 7, see 8)	Represents message exchange between a sender and a receiver Variant 6 used the Advanced Adapter Engine capabilities.
Variant 2: One sender and two	Represents message exchange between one sender and two receivers

receivers (see 4)	
Variant 3: One sender and two receivers – with routing condition (see 5)	Represents message exchange between one sender and two receivers in which the way the message is forwarded to the receivers is determined by routing conditions
Variant 4: One sender and two receivers – with mapping (see 6)	Represents message exchange between one sender and two receivers. A mapping is executed when the message is forwarded to one of the two receivers.

You can aim to learn about the following with the use cases:

- Working with the Enterprise Services Builder – Enterprise Services Repository: Understanding existing design objects
- Working with the Integration Builder – Integration Directory: Configuring simple scenarios and understanding created configuration objects
- Working with the runtime: Executing the use cases and checking the results by using monitoring

SAP provides the required design objects and files for these learning targets. This document contains a complete set of instructions that enable you to configure and execute the use cases without any additional special prerequisites.



SAP NetWeaver Process Integration must be installed and configured, and be error free. In this respect, note the information under 2.2.1.

In variants 1 ... 4, file directories on the host of the SAP system on which the Integration Server is installed serve as the senders and receivers of messages. These use cases rely on file directories (and therefore the file/FTP adapter) to keep the technical prerequisites as simple as possible.

In variant 5, an SAP system serves as sender and receiver of messages. This use case relies on the ABAP proxy runtime.

In variant 6, file directories on the host of the SAP system on which the Integration Server is installed serve as the senders and receivers of messages. These use cases rely on file directories (and therefore the File/FTP adapter) to keep the technical prerequisites as simple as possible. The messages are sent directly from the sender adapter to the receiver adapter without using the ABAP proxy runtime to show the possibility of using the Advanced Adapter Engine.

This document assumes that you will configure the variants **in succession**, in the order specified. While it is technically possible to configure the variants in a different order, doing so would mean that the order of the step-by-step instructions in this document would no longer be correct.



If you need to configure and execute the use cases as quickly as possible, skip the chapters concerning the design objects used (3.1, 4.1, 5.1 and 6.1) and start instead with configuration. You can then return to the skipped chapters at a later stage for a better understanding of the various concepts.

2.2 Prerequisites

2.2.1 General Prerequisites

To configure and execute the use cases, SAP NetWeaver with usage type Process Integration must be correctly installed and configured. The following table lists the prerequisites and the relevant guides:

List of required steps

Step	Documentation
1. You have installed SAP NetWeaver Process Integration 7.1.	<i>Master Guide – SAP NetWeaver</i> <i>Installation Guide – SAP Web AS ABAP and Java</i>
2. You have imported the content for the Enterprise Services Repository corresponding to the latest support package/patch.	Importing ESR Content  The demo examples are located in the Enterprise Services Repository in the software component SAP BASIS, software component version SAP BASIS 7.10, in the namespaces http://sap.com/xi/XI/Demo/Agency and http://sap.com/xi/XI/Demo/Airline .
3. You have configured SAP NetWeaver Process Integration 7.1.	Configuration of SAP NetWeaver Process Integration
4. You have configured the System Landscape Directory (SLD).	Configuring, Working with and Administering System Landscape Directory



Note that you set up the use cases in your development or test system, but **not** in your productive system.

2.2.2 Providing the necessary User and Authorizations

To log on to the SAP system to configure the simple use cases, you have to create a user *XIDEMO* with the following roles in the integration server client 001:

- SAP_XI_ADMINISTRATOR
- SAP_XI_CONFIGURATOR
- SAP_XI_MONITOR
- SAP_XI_DEMOAPP

2.2.3 Special Prerequisites for Configuring the Variants 1 up to 4 and 6

The variants 1 ... 4 and 6 are based on a very simple system landscape that comprises various different file systems (on the host of the SAP system on which the Integration Server is installed). When the use cases are executed, files are moved between file directories. Therefore, to be able configure the use cases, the file directories must have been created and described in the System Landscape Directory. Only then can you access the directories during configuration in the Integration Directory. To set up and describe the system landscape you must carry out the following steps.

Setting Up the File Directories

Perform the steps below to create the required directories on the host of the SAP system on which the Integration Server is installed.

1. Using Windows Explorer, open the file directory of the SAP system on which the Integration Server is installed.
2. Create a directory in which you can save files temporarily and create a sub directory in it. Give the sub directory an appropriate name.



A work directory already exists on many computers that can be used for this purpose and under which you can create the required directories.

3. Create three further sub directories in the first sub directory. Suggested names: `XiPattern1`, `XiPattern2`, `XiPattern3`, `XiPattern4`, `XiPattern5`.



You must have authorization to access this directory to be able to test the process integration scenario. Bear this in mind when you define the directory, and, if applicable, contact your system administrator.

4. Copy the following two files to directory `XiPattern1` and `XiPattern4`.
 - `ZXiPatternSenderFile1.xml`
 - `ZXiPatternSenderFile2.xml`



These files are provided by SAP in a ZIP file. The ZIP file is located in SAP Developer Network (SDN) at <https://www.sdn.sap.com/irj/sdn/nw-soa> → **Service Bus**. On the page *Service Bus-Based Integration* select *Getting Started*. On the page *Getting Started With Service Bus-Based Integration* select the hyperlink *SAP NetWeaver Process Integration – Test Files for Simple Use Cases*.

Creating the Technical and Business Systems in the System Landscape Directory

Perform the following steps to create the required technical and business systems in the System Landscape Directory (SLD):

1. To call the System Landscape Directory, choose *Process Integration* → *Start Integration Builder* in the user menu.
2. Start the SLD. To do so, on the Process Integration start page, choose *System Landscape Directory*.
3. Log on using the XIDEMO user you created in 2.2.2.

4. On the initial screen of the SLD, choose *Technical Systems*.
5. Create a new technical system. To do so, choose *New Technical System*.
6. Select the type *Standalone* by choosing the appropriate radio button.
7. Choose *Next*.
8. On the *Technical System Wizard - System Details* screen, enter the following information about the technical system:
 - *Technical System Identification*: Enter a name according to the following naming convention: <SID>_FileSystem. In this case, <SID> is the system ID of the SAP system (on which the Integration Server is installed).
 - *System Host Name*: Enter the host name of the system on which you are working, for example `pwdf2638`.
9. Choose *Finish*.
10. Now create three new business systems. To create the first business system, on the SLD start page, choose *Business Systems*.
11. Choose *New Business System*.
12. Select the type *Standalone* by selecting the appropriate radio button.
13. Choose *Next*.
14. Select the technical system that you defined previously (dropdown list box in field *System*).
15. Choose *Next*.
16. Enter the name <SID>_FileSystem_XiPattern1.
17. Choose *Next*.
18. Specify the assigned Integration Server to be used for process integration (field *Related Integration Server*).
19. Choose *Finish*.
20. Following the same procedure, create two further business systems with the following names:
 - <SID>_FileSystem_XiPattern2
 - <SID>_FileSystem_XiPattern3
 - <SID>_FileSystem_XiPattern4
 - <SID>_FileSystem_XiPattern5

Defining Business System Components in the Integration Directory

In the following steps you define communication components for the business systems you just created. You need to create communication components to be able to address the business systems as senders/receivers of messages when you configure the process integration scenario in the Integration Directory.

To prepare the configuration of the process integration scenario, you must do the following:

- Define the involved business systems as communication components (business system components) in the Integration Directory
Only then can you address the business systems as senders or receivers of messages.
- Define communication channels for outbound message processing (for all relevant receiver business systems)

You define communication components in the Integration Directory.

Therefore, to execute the activities described below, call the Integration Builder (user interface of Integration Directory).

1. To call the Integration Builder, choose *Process Integration* → *Start Integration Builder* in the user menu.
2. On the initial screen, choose the hyperlink *Integration Builder* (under *Integration Directory*).
3. Log on using the XIDEMO user you created in 2.2.2.

The system opens the Integration Builder.

You use the steps below to define a communication component (business system component) in the Integration Directory for each business system.

1. Clear the SLD cache. To do this, select *Environment* → *Clear SLD Data Cache* in the menu of the Integration Builder.
2. In the Integration Builder navigation area, on the *Objects* tab page, position the cursor on the *Communication Components Without Party* node.
3. To call the wizard for assigning business systems, choose *Assign Business System* in the context menu.
4. Choose *Continue*.
5. On the next screen, choose *Continue*.

The business system components are not assigned to a party.

6. On the next screen, select the following business systems by using the corresponding checkboxes:
 - <SID>_FileSystem_XiPattern1
 - <SID>_FileSystem_XiPattern2
 - <SID>_FileSystem_XiPattern3
 - <SID>_FileSystem_XiPattern4
 - <SID>_FileSystem_XiPattern5
7. Make sure that the *Create Communication Channels Automatically* checkbox is **deselected**.



The relevant communication channels are created when the process integration scenario is configured.

8. Choose *Finish*.

The system displays the communication components in the Integration Builder navigation area, on the *Objects* tab page, under *Communication Components Without Party*.

Perform the following steps to activate the communication components.

1. In the navigation area, choose the *Change Lists* tab page.
2. Choose *Open* → *Standard Change List*.
3. Check that the three communication components you just created are contained in the change list.
4. Position the cursor on the *Standard Change List* node and choose *Activate* () in the context menu.

2.2.4 Special Prerequisites for Configuring Variant 5

Variant 5 describes message exchange based on the ABAP proxy runtime. In this document the following standard configuration is described:

- The same application system is used as sender and receiver of messages.
- The application system is implemented by a separate client of the SAP system where the Integration Server is installed: client 001 is used for the Integration Server and client 105 for the application system. Client 105 is called *application client* in the following. The application system is represented by one business system in the System Landscape Directory and by a business system component in the Integration Directory.

Generation of Different Clients

You generate the application client from the Integration Server client 001 by using client copy.

To do this, execute the following steps in the SAP system of the Integration Server:

1. Set up the logical systems (for travel agency and airlines).
2. Assign client and logical system.
3. Perform client copy.



The client for the Integration Server is set up during installation of SAP NetWeaver Process Integration.

The relevant configuration steps are described in detail in the document *SAP NetWeaver Process Integration - Demo Examples Configuration Guide* in the chapter *Setting up the System Landscape* (section 4.2).

To configure this variant of the use cases, you only have to consider the steps according to client 105. As a description of the client, choose *SenderReceiver*.



You can find this document in SAP Developer Network (SDN) at <https://www.sdn.sap.com/irj/sdn/soa-servicebus>. In the Knowledge Center on the *Service Bus-Based Integration* page choose *Getting Started*. Choose the *SAP NetWeaver Process Integration - Demo Examples Configuration* document.

Creating the Technical and Business Systems in the System Landscape Directory

Since the involved business system is simulated by a specific client of the SAP system of the Integration Server, you only need one technical system (the system of the Integration Server).



The system of the Integration Server registers itself as a technical system in the System Landscape Directory automatically during installation of SAP NetWeaver Process Integration (according to the *Installation Guide*).

You must assign the application client to this technical system.

1. On the initial screen of the System Landscape Directory, choose *Technical Systems*. The system displays all the technical systems in a table. The technical system of the Integration Server has the following name: <SID> on <Server Name>.



If you have called the system <SID>, the name of the server (<Server Name>) is displayed in the corresponding status field in the status bar of the SAP window (for the system <SID>) (for example, PWFDF1234).

2. Open the technical system <SID> on <Server Name>. To do this, click the corresponding entry.
3. Details of the system are displayed.
4. Assign the application client to the technical system. To do this, in the *Clients* tab, choose *Add New Client*.
5. Enter the following information:

Field	Description	Entry
Client Number	Client	105
Logical System Name	Name of the associated Logical System	<SID>CLNT105
Description	Description	SenderReceiver

6. To save your entries, choose *Save*.

Execute the following steps to define a business system for the sender/receiver:

1. On the initial screen of the System Landscape Directory, choose *Business Systems*.
2. Choose *New Business System*. The system opens the business system wizard.
3. Select the system type *AS ABAP* with the corresponding radio button.
4. Choose *Next*.
5. First specify the technical system. In the dropdown list box for the *System* field, select the name of the technical system. The name of the technical system is <SID> on <Server Name>.
6. In the *Client* field, select the client that implements the application system: 105.
7. Choose *Next*.
8. Enter the following name for the business system: <SID>_105.
9. Choose *Next*.
10. Select all the specified products. All software component versions resulting from the product selection are displayed.
11. Choose *Next*.
12. Enter the following information:
 - In the *Business System Role* field, specify the role of the business system. Specify the role *Application System* for the clients of the application systems.
 - In the *Related Integration Server* field, specify the technical system of the Integration Server (<SID> on <Server Name>).
13. To save your entries, choose *Finish*.

Configuring the Business System as Local Integration Engine

The relevant configuration steps are described in detail in the document *SAP NetWeaver Process Integration - Demo Examples Configuration Guide* in the chapter 4.4 *Configuring the Business System as Local Integration Engine*.

To configure this variant of the use cases, you only have to consider the steps according to client 105.

Defining Business System Components in the Integration Directory

In the following steps you define a communication component for the business system you just created.

To prepare the configuration of the process integration scenario, you must do the following:

- Define the involved business system as communication component (business system component) in the Integration Directory
Only then can you address the business system as sender or receiver of messages.
- Define a communication channel for outbound message processing (for the receiver business system)

You define the communication component in the Integration Directory.

Therefore, to execute the activities described below, call the Integration Builder (Integration Directory).

1. To call the Integration Builder (Integration Directory), choose *Process Integration* → *Start Integration Builder* in the user menu.
2. On the initial screen of Process Integration, choose the hyperlink *Integration Builder* (under *Integration Directory*).
3. Log on using the XIDEMO user you created in 2.2.2.
The system opens the Integration Builder (Integration Directory).

You use the steps below to define a communication component (business system component) in the Integration Directory for each business system.

1. Clear the SLD cache. To do this, select *Environment* → *Clear SLD Data Cache* in the menu of the Integration Builder.
2. In the Integration Builder navigation area, on the *Objects* tab page, position the cursor on the *Communication Components Without Party* node.
3. To call the wizard for assigning business systems, choose *Assign Business System* in the context menu.
4. Choose *Continue*.
5. On the next screen, choose *Continue*.
The business system components are not assigned to a party.
6. On the next screen, select the following business system by using the corresponding checkbox: <SID>_105.
7. Make sure that the *Create Communication Channels Automatically* checkbox is **selected**.
8. Choose *Finish*.

The system displays the communication component in the Integration Builder navigation area, on the *Objects* tab page, under *Communication Components Without Party* → *Business System*.

The communication channels that are generated automatically are displayed in the Integration Builder navigation area under *Communication Channel*. The communication channels with adapter type *XI* (for proxy-to-proxy communication) have the name *GeneratedReceiverChannel_XI*.

The name of the communication channel in the navigation area is preceded with the name of the business system the channel is assigned to.

You need a receiver communication channel with adapter type *XI*. To be able to use the communication channel, you have to edit it manually.

Editing the Communication Channels

You must execute the following steps to edit the communication channel *GeneratedReceiverChannel_XI*.

1. In the Integration Builder navigation area, under *Communication Channel*, open the channel **I<SID>_105/GeneratedReceiverChannel_XI*.
2. Switch to change mode. To do this, choose *Switch between display and edit mode* (.
3. Enter the following values for the communication channel attributes.



The table contains the correct entries for all fields in the communication channel. Note that some fields will be completed by the system when communication channels are created automatically. The *Specify Manually* column indicates which fields remain to be completed.

Entries for Communication Channels

Specify Manually	Attribute	Value
	Transport Protocol	HTTP 1.0
	Message Protocol	XI 3.0
	Addressing Type	URL Address
	Target Host	Name of the server of the target host (for example, <code>pwdf0005</code>)
X	Service Number	Number of the HTTP port  To find out the number of the HTTP port (service number), see the ICM Monitor of the relevant Integration Server (call transaction <i>SMICM</i> and then <i>Goto</i> → <i>Services</i>). Select the Service Name/Port for the Protocol HTTP.
	Path	<code>/sap/xi/engine?type=entry</code>

Authentication Data

Specify Manually	Attribute	Value
x	Authentication Type	Use Logon Data for SAP System
x	User Name	PIAPPLUSER
x	User Password	Use the password set during installation.
x	Logon Language	EN
x	Client	105

4. Save the communication channel (.



Sender agreements are **not** necessary for adapter type XI and XI 3.0 protocol. Therefore, no sender communication channels are necessary.

Perform the following steps to activate the communication component and communication channel.

1. In the navigation area, choose the *Change Lists* tab page.
2. Choose *Open* → *Standard Change List*.
3. Check that the three communication components you just created are contained in the change list.
4. Position the cursor on the *Standard Change List* node and choose *Activate* (🔧) in the context menu.

3 Variant 1: One Sender and One Receiver

This variant enables you to configure and execute the simplest kind of message exchange: one sender (file directory *XiPattern1*) sends a message to one receiver (file directory *XiPattern1*). No mapping is executed between the outbound and inbound interface in this variant.

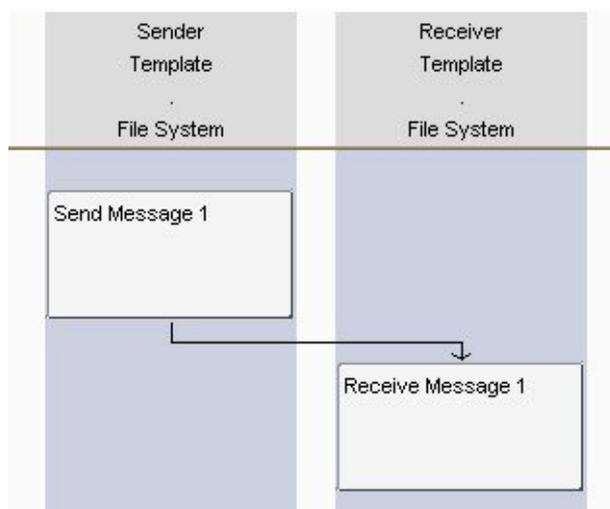
3.1 Design Objects Used



This section explains all the design objects that are used for describing this variant. There is a step-by-step description of how to access each object. SAP recommends that you follow these steps directly in the system.

The subsequent description of each variant will refer to the detailed step-by-step description in this section; the relevant design objects will only be listed briefly in each case.

You use the component view *File2File_NoMapping* in process integration scenario *XiPatternOneStepScenario* to configure this variant. The following graphic shows the component view:



The following describes all the objects that describe the message exchange in this variant in turn. Starting with the process integration scenario (component view), you can open all the objects that the component view uses.

1. To call the Enterprise Services Builder (user interface of the Enterprise Services Repository), choose *Process Integration* → *Start Integration Builder* in the user menu.
2. On the *Process Integration* initial screen, choose *Enterprise Services Builder* under (*Enterprise Services Repository*).
3. In the navigation area, open the software component *SAP BASIS*, software component version *SAP BASIS 7.10*, and namespace *http://sap.com/xi/XI/System/Patterns*.
4. Open the node *Process Integration Scenarios*.
5. Open process integration scenario *XiPatternOneStepScenario* by double-clicking.
The process integration scenario editor is opened.

Process Integration Scenario/Component View

A process integration scenario describes the process flow and represents the points in the process flow at which messages are exchanged between application components (in this case, the application components *Sender* and *Receiver*). Specific communication variants are represented by different component views. Component View *File2File_NoMapping* represents the three variants without a mapping. Component View *File2File_WithMapping* represents the variant that includes a mapping.

Ensure that component view *File2File_NoMapping* is displayed in the graphical editor. If it is not displayed, choose the component view in the preview area on the left by double-clicking. The component view comprises two actions *Send Message 1* and *Receive Message 2*, which are linked by an asynchronous communication step (angled arrow).

Interface Objects

Interface objects describe the communication mode and the data structure for exchanging messages. Interface objects comprise service interfaces, message types, and data types, which you can open by means of forward navigation, starting in the component view.

1. To open the interface objects used, double-click the connecting arrow in the component view.

The connection editor is displayed.

The service interfaces for the sender and receiver are displayed on the *Assign Interfaces* tab page: in the case of the sender, it is the outbound interface *XiPatternInterface1* and in the case of the receiver, it is the same inbound interface *XiPatternInterface1*. Since the interface is the same for both the sender and receiver, no mapping is required.

2. To open the service interface *XiPatternInterface1*, double-click the interface name.
3. In the *Service Interface* editor you can see that it is an abstract interface.

The interface is *abstract* because the interface description is not implemented in the relevant software component.

4. To open the message type, double-click the name of the message type *XiPatternMessage1* (in the service interface editor under *Messages*).

In the *Message Type* editor, you can display the data structure of the message. The data structure is defined by the data type *XiPatternDataType1*.

Information about a person is structured by different elements in the data structure of the exchanged message. Details such as the maximum character length are displayed in the *Details* column.

Mapping Objects

If the interfaces at the sender and receiver are different, a mapping is required.

Navigate to the connection editor and choose the *Assign Mapping* tab page. You can see that a mapping has not been assigned for the connection between the two (identical) service interfaces.

Communication Channel Templates

You use communication channel templates to define details, which were defined when the process integration scenario was defined, for the inbound or outbound processing of the message. You can then use the communication channel template to create the corresponding communication channel when configuring in the Integration Directory. You then need to specify all the details, which are only known at configuration time, in the communication channel.

1. In the connection editor, choose the *Assign Communication Channel Template* tab page. One communication channel is specified for both the sender and receiver: (*XiPatternChannel_FileSender* for the sender, *XiPatternChannel_FileReceiver* for the receiver).
2. To open a communication channel template (for example, for the sender), double-click the name. Further details are displayed in the *Display Communication Channel Template* editor.
3. Close all objects and the process integration scenario.

Summary of Design Objects Used

The design objects used are summarized once more in the following table.



For a general description of the various object types, see [Enterprise Services Repository](#).

Design Objects Used

Object Type (Name)	Description
Process Integration Scenario (<i>XiPatternOneStepScenario</i>)	Describes the process flow and indicates the points at which messages are exchanged between the different components. The component view <i>File2File_NoMapping</i> in particular describes all variants in which no mapping is executed (interfaces at the sender and receiver are identical).
Service Interface (<i>XiPatternInterface1</i>)	Specifies the communication mode (asynchronous) and references the message type used.
Message Type (<i>XiPatternMessage1</i>)	Describes the message sent at runtime and references the data type used.
Data Type (<i>XiPatternDataType1</i>)	Describes the data structure of the message
Communication Channel Templates for Sender and Receiver (<i>XiPatternChannel_FileSender</i> , <i>XiPatternChannel_FileReceiver</i>)	Specifies the sender (and receiver) adapter configuration details, which are defined at design time. Communication channel templates enable a communication channel to be preconfigured at design time.

3.2 Configuring the Process Integration Scenario

You perform the following configuration steps in the Integration Directory.



Note the prerequisite steps described in section 2.2.

In configuration, you translate the process described by the process integration scenario (component view) to an actual system landscape. You have already described the system landscape in the SLD. The communication components you defined earlier enable you to address the involved business systems as senders/receivers of messages in the Integration Directory.

You use the model configurator for the whole configuration procedure. This tool enables you to use an process integration scenario from the Enterprise Services Repository as a configuration template and to add further information (which is not known until configuration time).

Calling the Integration Directory

First, call the Integration Directory. On the *Process Integration* initial screen, choose hyperlink *Integration Builder* (under *Integration Directory*).

Selecting the Process Integration Scenario

Perform the following steps to select the process integration scenario that you want to configure from the Enterprise Services Repository. Also, enter the name of the *configuration scenario* (in the Integration Directory) in which you want to group the configuration objects.

1. To call the model configurator, choose *Tools* → *Transfer Model from Enterprise Services Repository* in the Integration Builder menu bar.

This calls a wizard, which you use to select a model (in this case, a process integration scenario) and create a configuration scenario in the Integration Directory.

2. Make sure that the model type *Process Integration Scenario* is selected (checkbox under *Type of ES Repository Model*).

3. Choose the input help (for the *Name* field) () and select *Process Integration Scenario* in the dropdown listbox.

This calls an input help, which displays all process integration scenarios from the Enterprise Services Repository in a list.

4. Select the process integration scenario *XiPatternOneStepScenario* (software component version *SAP BASIS 7.10*, namespace <http://sap.com/xi/XI/System/Patterns>) from the Enterprise Services Repository and choose *Apply*.

5. Choose *Continue*.

6. Enter the name of the configuration scenario as: `XiPatternOneStepScenario`.

7. Choose *Finish* and then *Close*.

The system calls the model configurator.

Selecting the Component View

Check whether the component view *File2File_NoMapping* is displayed. If it is not, choose the menu option *Select Component View* (). On the subsequent screen, select the component view (in the preview area) by double-clicking.

Assigning Communication Components

You perform the following steps to assign communication components to the application components of the process integration scenario. You use the business system components that you created before (see 2.2.2).

1. In the graphical editor, click on the lane representing the *Sender* application component.
This calls the *Assign Business System Component* dialog box for the *Sender* application component (below the graphical representation of the process integration scenario).
2. To assign a communication component, on the *Business System Components for A2A* tab page, choose *Insert Line After Selection* (.
3. Click in the input field in the *Communication Component* column and call the input help (.
4. In the subsequent screen, select the communication component `<SID>_FileSystem_XiPattern1` and then choose *Apply*.
5. Choose *Next Role* () to navigate to the configuration dialog for the application component *Receiver*.
6. Assign the communication component `<SID>_FileSystem_XiPattern2` to the application component *Receiver*. Follow the same procedure as when assigning the communication component to the application component *Sender*.
7. Choose *Assign*.

Configuring the Connection

You perform the following steps to configure the connection of the process integration scenario. You activate the relevant sender-receiver relation (between the sender and receiver component) and assign both the sender and receiver a (sender or receiver) communication channel. You create the communication channel on the basis of the communication channel templates that the process integration scenario already contains.

1. In the graphical editor, click on the connection between the lanes representing the two application components.
2. This calls the *Configure Connection* dialog box for the connection between the actions (below the graphical representation of the process integration scenario).
3. On the *Configure Connections* screen, choose the tab page *Connections from Component Assignment*. The communication components that you assigned in the previous configuration step are displayed.



The remaining tab pages contain further information about the process integration scenario, for example, the communication channel templates that are defined for the process integration scenario in the Enterprise Services Repository.

4. Assign the sender a communication channel. To do so, in the push button bar choose  → *Create Communication Channel With Template* (). In the subsequent wizard you can create a communication channel based on a communication channel template.
5. Choose *Continue*.
The default communication channel template on the subsequent screen is that which was assigned to the sender application component in the process integration scenario (name `XiPatternChannel_FileSender`, software component version `SAP BASIS 7.10`, namespace `http://sap.com/xi/XI/System/Patterns`).
6. Choose *Continue*.
The default name for the communication channel on the next screen is `XiPatternChannel_FileSender`.
7. Confirm your selection by choosing *Finish*.

8. Following the same procedure (see step 4-7), assign the application component of the receiver a communication channel (*XiPatternChannel_FileReceiver*) based on the default communication channel template (name *XiPatternChannel_FileReceiver*, software component version *SAP BASIS 7.10*, namespace *http://sap.com/xi/XI/System/Patterns*).
9. Choose *Apply*.

Generating the Configuration Objects

In the following steps, you generate the relevant configuration objects based on the settings made in previous steps.

1. Choose the push button *Create Configuration Objects* ()
2. Under *General*, select the radio button *Generation*.
3. Choose *Start*.
The configuration objects are generated based on the settings you made. Once generation is complete, the generation log is opened. The generation log displays the results of generation. The generation log enables you to analyze any errors that occurred during generation.
4. Close the generation log.
5. Close the model configurator. Choose *Accept changes*.
6. Save the configuration scenario.

Manual Editing

In the following steps, you edit the generated configuration objects manually and enhance the configuration settings made during generation. In this variant, you must edit the communication channels after generation.

1. In the editor *Display Configuration Scenario* (for the open scenario *XiPatternOneStepScenario*), on the tab page *Configuration Scenario Objects* under *Communication Profile Objects*, open the communication channel () *XiPatternChannel_FileSender* by double-clicking.
2. Under *File Access Parameter* in the *Source Directory* field, enter the path in the file directory that you defined previously (see 2.2.2). In the case of the sender communication channel, enter the whole path for the file directory *XiPattern1*.
Example: C:/<selected path>/XiPattern1
3. Save the communication channel.
4. Following the same procedure, edit the communication channel *XiPatternChannel_FileReceiver* (communication component <SID>_FileSystem_XiPattern2). Enter the path for the file directory *XiPattern2*.
5. Save the communication channel.

Activating the Change List

1. To do this, select the *Change Lists* tab page in the Integration Directory navigation area.
2. Choose *Open* → *Standard Change List*.
3. Position the cursor on the *Standard Change List* node and choose *Activate* () in the context menu.

Summary of the Configuration Objects Used

The generated configuration objects used are summarized once more in the following table.



SAP recommends that you look at the configuration objects directly in the system and that you understand the configuration settings resulting from these objects.



For a general description of the various object types, see [Integration Directory](#).

Configuration Objects Used

Object Type (Name)	Description
Configuration Scenario (<i>XiPatternOneStepScenario</i>)	Groups all configuration objects that are relevant for the execution of the process integration scenario.  <p>This documentation recommends that you group the configuration objects for all variants of the process integration scenario in the same configuration scenario.</p>
Business System Component (<i><SID>_FileSystem_XiPattern1</i> for sender and <i><SID>_FileSystem_XiPattern2</i> for receiver)	Identifies a business system in the Integration Directory that is to be addressed as the sender or receiver of messages.
Sender Communication Channel (<i>XiPatternChannel_FileSender</i>)	Contains the details for configuring a sender adapter that is used to process the message on the inbound side. In a real installation, the sender file/FTP adapter is used to write a file from a source directory to the pipeline.
Receiver Communication Channel (<i>XiPatternChannel_FileReceiver</i>)	Contains the details for configuring a receiver adapter that is used to process the message on the outbound side. The receiver file/FTP adapter writes the file to one or more target directories.
Receiver Determination (<i>! <SID>_FileSystem_XiPattern1 XiPatternInterface1</i>)	Specifies the receiver of the message for the sender and the outbound interface <i>XiPatternInterface1</i> . The receiver <i><SID>_FileSystem_XiPattern2</i> must be entered in the receiver determination as a configured receiver.
Interface Determination (<i>! <SID>_FileSystem_XiPattern1 XiPatternInterface1 <SID>_FileSystem_XiPattern2</i>)	Specifies the inbound interface <i>XiPatternInterface1</i> for the sender, outbound interface, and the receiver. The inbound interface must be entered in the interface determination as a configured inbound interface.
Sender Agreement (<i>! <SID>_FileSystem_XiPattern1 XiPatternInterface1</i>)	Specifies the sender communication channel <i>XiPatternChannel_FileSender</i> to be used for the sender and the outbound interface.

Receiver Agreement (1 <SID>_FileSystem_XiPattern1 <SID>_FileSystem_XiPattern2 XiPatternInterface1)	Specifies the receiver communication channel <i>XiPatternChannel_FileReceiver</i> to be used for the sender, the receiver, and the inbound interface.
---	---

3.3 Execution

You perform the following steps to check that the process integration scenario is executed without errors.

1. On the host of the Integration Server, open the file directory *XiPattern1*.
2. Rename the file **ZXiPatternSenderFile1.xml** as follows: *XiPatternSenderFile1.xml*.
The File/FTP adapter is designed so that a file with exactly this name is recognized and moved from the source directory *XiPattern1* to the target directory *XiPattern2* every 60 seconds. If the configuration was executed without errors, the file will be moved from the source directory 60 seconds after it is renamed.
3. Open the directory *XiPattern2* and check that the file has arrived.
4. Open the file and check whether the content has changed (compare with the original file *ZXiPatternSenderFile1.xml*).



The file must not be altered when this variant is executed.

Furthermore, you can check the processing of the message in monitoring. To do this, proceed as follows:

1. Start the Runtime Workbench by choosing *Runtime Workbench* on the Process Integration start page.
2. Choose *Message Monitoring*.
3. Select *Integration Engine <SID>* as the component and then choose *Display*.
4. Use appropriate filter criteria to restrict the number of XML messages displayed (for example, the processing period).
5. Choose *Start*.
The system displays the selected messages in a table. You want to locate the messages that were sent during execution.
6. Select the message to be displayed by selecting the appropriate radio button and then choose *Details*.
The tab page *Message Contents* displays the message contents including all message versions for the various phases of message processing.

4 Variant 2: One Sender and Two Receivers

This variant extends variant 1 by adding an additional receiver. The message is sent from the sender to two receivers, without a mapping being executed at runtime.

4.1 Design Objects Used

The same component view and design objects are used as in variant 1 (see 3.1).

4.2 Configuring the Process Integration Scenario

When you configure this variant, you will partially overwrite the configuration settings from variant 1. You use the same configuration scenario *XiPatternOneStepScenario* as in variant 1.

Opening the Configuration Scenario

Perform the following steps to open the configuration scenario that you created in variant 1.

1. In the Integration Directory, open configuration scenario *XiPatternOneStepScenario*. To do so, in the Integration Directory navigation area, choose the tab page *Scenarios* and open the configuration scenario *XiPatternOneStepScenario* by double-clicking.



If you followed the configuration sequence suggested in this document, the configuration objects that were generated in variant 1 are displayed in the configuration scenario.

2. Start the model configurator. To do so, in the menu bar of the configuration scenario object editor, choose *Configuration Scenario* → *Model Configurator*.



The model configurator loads the same process integration scenario as in variant 1.

Selecting the Component View

Check that the component view *File2File_NoMapping* is selected. If it is not, under *Configuration Steps*, choose the menu option *Select Component View*. On the subsequent screen, select the component view (in the preview area) by double-clicking.

Assigning Communication Components

Additionally, assign the application component *Receiver* the business system component *<SID>_FileSystem_XiPattern3*.

Configuring the Connection

The communication channels selected for variant 1 are displayed for the relation from sender communication component *<SID>_FileSystem_XiPattern1* and receiver component *<SID>_FileSystem_XiPattern2*. If this is not the case, select them again by using the input help.

For the relation from the sender component *<SID>_FileSystem_XiPattern1* and receiver component *<SID>_FileSystem_XiPattern3*, proceed as follows:

- Assign the communication channel *XiPatternChannel_FileSender* (communication component *<SID>_FileSystem_XiPattern1*) that you created previously to the sender component. To do so, position the cursor on the input field in the column *Communication Channel* (in the composite column *Sender Business System Components for A2A*) and call the input help ().
- Using the communication channel template, create a new communication channel for the receiver communication component *<SID>_FileSystem_XiPattern3*. To do so, proceed as described in variant 1.

Generating the Configuration Objects

Proceed as described in variant 1. Once you have checked the generation log, close it and save the configuration scenario.

Manual Editing

Manually edit the new communication channel *XiPatternChannel_FileReceiver* for communication component *<SID>_FileSystem_XiPattern3* as in variant 1. Under *File Access Parameter* in field *Source Directory*, enter the path for the file directory *XiPattern3* (see 2.2.2).

Summary of the Configuration Objects Used

The same configuration objects are used as in variant 1.

The following differences arise as a result of the additional receiver component:

- Two receiver components are configured in the receiver determination (as configured receivers). At runtime, the message is sent simultaneously from sender *<SID>_FileSystem_XiPattern1* to the two receivers *<SID>_FileSystem_XiPattern2* and *<SID>_FileSystem_XiPattern3*.
- There is an additional communication channel *XiPatternChannel_FileReceiver* for the additional receiver *<SID>_FileSystem_XiPattern3*.
- There is a further receiver agreement *I <SID>_FileSystem_XiPattern1 | <SID>_FileSystem_XiPattern3 | XiPatternInterface1*. The receiver agreement specifies the receiver communication channel *XiPatternChannel_FileReceiver* to be used for the sender, additional receiver *<SID>_FileSystem_XiPattern3*, and the inbound interface.

4.3 Execution

You perform the following steps to check that the process integration scenario is executed without errors.

As for the execution of variant 1, in directory *XiPattern1*, copy the file *ZXiPatternSenderFile1.xml* and rename it *XiPatternSenderFile1.xml*. The file will be moved from the directory within the next 60 seconds.

Since this variant has two target directories (receivers), check in both directory *XiPattern2* and *XiPattern3*, whether the file has arrived.

Message monitoring will display three messages (see 3.3).



In this variant, two messages are created from the outbound message (one for each receiver).

5 Variant 3: One Sender and Two Receivers – with Routing Condition

This variant extends variant 2 by adding a routing condition. Depending on the condition, the message is sent to one of the two receivers.

5.1 Design Objects Used

The same component view and design objects are used as in variant 1 (see 3.1).

5.2 Configuring the Process Integration Scenario

To configure this variant you use the configuration settings from variant 2. The only enhancement you need to make is to add a routing condition to the receiver determination.

1. Open configuration scenario *XiPatternOneStepScenario*.
2. On the tab page *Configuration Scenario Objects* under *Logical Routing and Collaboration Agreement Objects*, open the receiver determination *I <SID>_FileSystem_XiPattern1 I XiPatternInterface1* () by double-clicking.
3. Switch to change mode ()
4. Under *Configured Receivers*, select the cell for the receiver component *<SID>_FileSystem_XiPattern2* in the column *Condition*.
5. Call the condition editor by choosing the input help icon ()
6. Choose *Insert Expression* () . A line is inserted for the condition.
7. In the column *Left Operand*, call the expression editor by choosing the input help icon ()
8. In the expression editor, select the *XPath* radio button.
The data structure of the message is displayed.
9. Expand the structure and select the element *CountryCode* by using the cursor.
10. Choose *Apply*.
11. In the condition editor, choose the operator = in the column *Op*.
12. In the column *Right Operand*, enter *US*.
13. Choose *Apply*.

You have specified the routing condition `CountryCode = US`.



At runtime, messages that have the value *US* entered for the element *CountryCode* are sent to receiver *<SID>_FileSystem_XiPattern2*.

14. Following this procedure, create the routing condition `CountryCode = DE` for the configured receiver *<SID>_FileSystem_XiPattern3*.
15. Save the receiver determination.
16. Activate your change lists.

5.3 Execution

You perform the following steps to check that the process integration scenario is executed without errors.

In directory *XiPattern1*, copy the file *ZXiPatternSenderFile1.xml* and rename the copied file *XiPatternSenderFile1.xml*. The file will be moved from the directory within the next 60 seconds.

Check whether the file has arrived in directory *XiPattern2*. Open the file and check that the country code *US* is entered in the personal data.

There should be no file in directory *XiPattern3*.

Monitoring contains an entry for this exchange of messages (see 3.3).



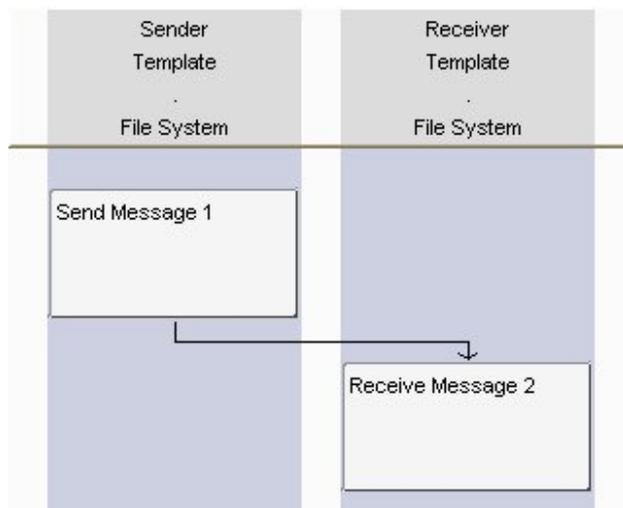
You can repeat this procedure for file *ZXiPatternSenderFile2.xml*. This file should arrive in directory *XiPattern3* (contains personal data with the country code *DE*).

6 Variant 4: One Sender and Two Receivers – with Mapping

In this variant, the message is sent to two receivers. A mapping is executed when the message is forwarded to one of the two receivers.

6.1 Design Objects Used

You use the component view *File2File_WithMapping* in process integration scenario *XiPatternOneStepScenario* to configure this variant. The following graphic shows the component view:



As described under 3.1, you can navigate to all design objects used from the component view. The following explains the design objects that play an **additional** role when using this component view.

Interface Objects

You can see that the outbound interface entered in the connection editor is the same as the interface in component view *File2File_NoMapping*. If you navigate further to the message type and data type used, you will see that the data structure on the sender side is the same as in the previous variants (defined by the data type *XiPatternDataType1*).

However, the asynchronous service interface *XiPatternInterface2* is entered as the inbound interface at the receiver. This service interface references the message type *XiPatternMessage2*. The data structure of the message is defined by the data type *XiPatternDataType2*. The structure of data type *XiPatternDataType2* is different to the structure of data type *XiPatternDataType1* (which is used in the component view) in the following ways:

- In data type *XiPatternDataType1*, two separate elements are used to specify the person's first and second name, whereas in data type *XiPatternDataType2*, only **one** element is used (*FullName*).

- In data type *XiPatternDataType1*, the name of the element that specifies the telephone number is *TelephonNumber*, whereas in data type *XiPatternDataType2*, the name is *PhoneNumber*.

Mapping Objects

A mapping between the sender and receiver data structures must therefore be executed for the message to be processed.

In the connection editor, navigate to the relevant mapping objects by choosing the tab page *Assign Mapping*.

Double-click on the name of the operation mapping (*XiPatternInterface1ToInterface2*) to open it. The names of the source and target interface, as well as the source and target message are displayed. The message mapping used is displayed under *Mapping Program*. Double-click to open the message mapping (*XiPatternMessage1ToMessage2*).

In the message mapping editor, the tab page *Design* displays the source and target structures.



You have the option of displaying for which elements of the source and target structure mappings are to be executed. To do so, choose *Display Dependencies* → *Display All* ().

Summary of Additional Design Objects Used

The following table summarizes the additional design objects used.

Additional Design Objects Used

Object Type (Name)	Description
Component View <i>File2File_WithMapping</i> (Process Integration Scenario <i>XiPatternOneStepScenario</i>)	Describes the process flow and the message exchange if a mapping is to be executed during message processing (the interfaces on the sender and receiver sides are not identical)
Service interface (<i>XiPatternInterface2</i>)	Describes the interface and data structure of the receiver. The data structure is different to the data structure of the sender.
Message Type (<i>XiPatternMessage2</i>)	
Data Type (<i>XiPatternDataType2</i>)	
Operation mapping (<i>XiPatternInterface1ToInterface2</i>)	Describes the mapping between the source and target interface
Message Mapping (<i>XiPatternMessage1ToMessage2</i>)	Describes the mapping between the source and target structure in detail

6.2 Configuring the Process Integration Scenario

This variant uses the configuration settings from the previous variant.

Open configuration scenario *XiPatternOneStepScenario* and call the model configurator from the configuration editor.

Selecting an Process Integration Scenario (Component View)

Select component view *File2File_WithMapping*.



Any settings previously made in the model configurator will be overwritten.

Assigning Communication Component

Assign the communication component `<SID>_FileSystem_XiPattern1` to the application component *Sender*, and the communication component `<SID>_FileSystem_XiPattern3` to the application component *Receiver*.



In this variant, only the connection between sender `<SID>_FileSystem_XiPattern1` and receiver `<SID>_FileSystem_XiPattern3` is reconfigured. The configuration of the connection between sender `<SID>_FileSystem_XiPattern1` and receiver `<SID>_FileSystem_XiPattern2` (defined in the previous variants) remains unchanged.

Configuring the Connection

When you configure the connection, you select the communication channels that were selection in the previous variants.

Generating the Configuration Objects

To do so, proceed as described in variant 1.

Under *Change to Existing Interface Determinations*, select the checkbox *Permit Change*.



Generation reuses the interface determination from variant 1. You must select this option to enter the mapping to be used in variant 4 in the existing interface determination.

When you have checked the generation log, close it and save the configuration scenario. Activate your change lists.

6.3 Execution

You perform the following steps to check that the process integration scenario is executed without errors.

In directory *XiPattern1*, copy the file *ZXiPatternSenderFile2.xml*. Rename the directory *XiPatternSenderFile2.xml*. The file will be moved from the directory within the next 60 seconds.

Check whether a file has arrived in directory *XiPattern3*. Open the file. The structure will have changed: the first and second name of the person have been concatenated; the field for the telephone number has a new name.

Monitoring contains an entry for this exchange of messages (see 3.3).



You can repeat this procedure for file *ZXiPatternSenderFile1.xml*. This file is expected in directory *XiPattern2* (no mapping executed).

7 Variant 5: One Sender and One Receiver - Proxy-to-Proxy Communication

In this variant, the message is processed via the ABAP proxy runtime. Following the standard configuration suggested below one application system is used both as sender **and** receiver of messages.

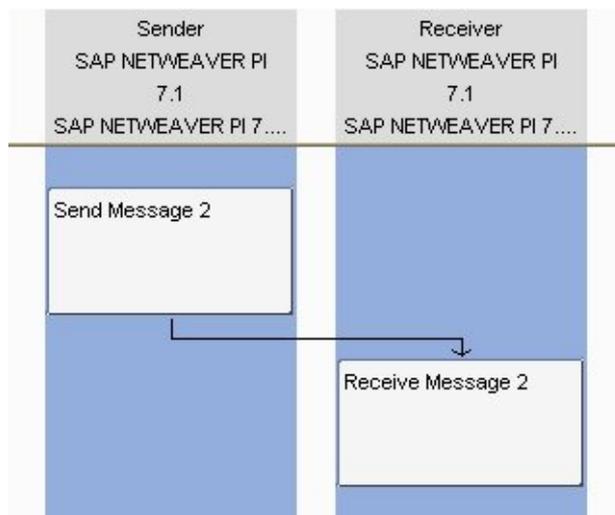


You can use this variant to test if an SAP system to be used as business system is configured correctly with regard to the connection from the SAP system to the Integration Server. In particular, you can test if an SAP system works correctly both as sender and receiver of messages.

You can use this use case to check if the installation of SAP NetWeaver Process Integration is correct.

7.1 Design Objects Used

You use the component view *Proxy2Proxy_NoMapping* in process integration scenario *XiPatternOneStepScenario* to configure this variant. The following graphic shows the component view:



As described under 3.1, you can navigate to all design objects used from the component view. The following explains the design objects that play a role when using this component view.

Interface Objects

In this variant, the data structures at both the sender and receiver side are the same and no mapping is required. Nevertheless, two different service interfaces are used: outbound interface *XiPatternInterface2_Out* for the sender and inbound interface *XiPatternInterface2_In* for the receiver. This is due to the fact that the sender and receivers systems are SAP systems that communicate with each other by the ABAP proxy runtime. This communication type requires an outbound proxy and an inbound proxy (both generated resp. derived from the outbound resp. inbound interface).

Summary of Design Objects Used

The following table summarizes the additional design objects used.

Design Objects Used

Object Type (Name)	Description
Component View <i>Proxy2Proxy_NoMapping</i> (Process Integration Scenario <i>XiPatternOneStepScenario</i>)	Describes the process flow and the message exchange for this variant.
Service interface (<i>XiPatternInterface2_Out</i>)	Describes the interface and data structure of the receiver. The data structure is the same for sender and receiver. The fault message type is used to handle application-specific errors that occur on the inbound side and that are reported back to the sender.
Service Interface (<i>XiPatternInterface2_In</i>)	
Message Type (<i>XiPatternMessage2</i>)	
Fault Message Type (<i>XiPatternFaultMessage</i>)	
Data Type (<i>XiPatternDataType2</i>)	

7.2 Configuring the Process Integration Scenario

To configure this variant, a new configuration scenario is created.

Selecting the Process Integration Scenario

Perform the following steps to select the process integration scenario that you want to configure from the Enterprise Services Repository. Also, enter the name of the configuration scenario (in the Integration Directory) in which you want to group the configuration objects.

- To call the model configurator, choose *Tools* → *Transfer Model from Enterprise Services Repository* in the Integration Builder menu bar.

This calls a wizard, which you use to select a model (in this case, a process integration scenario) and create a configuration scenario in the Integration Directory.
- Make sure that the model type *Process Integration Scenario* is selected (checkbox under *Type of ES Repository Model*).
- Choose the input help (for the *Name* field) () and select *Process Integration Scenario* in the dropdown listbox.

This calls an input help, which displays all process integration scenarios from the Enterprise Services Repository in a list.
- Select the process integration scenario *XiPatternOneStepScenario* (software component version *SAP BASIS 7.10*, namespace *http://sap.com/xi/XI/System/Patterns*) from the Enterprise Services Repository and choose *Apply*.
- Choose *Continue*.
- Enter the name of the configuration scenario as:
XiPatternOneStepScenario_Proxy2Proxy.
- Choose *Finish* and then *Close*.

The system calls the model configurator.

Selecting the Component View

Select component view *Proxy2Proxy_NoMapping*.

Assigning Communication Component

Assign the communication component <SID>_105 to the application component *Sender* and to the application component *Receiver*.

Configuring the Connection

When you configure the connection, assign the communication channel *GeneratedReceiverChannel_XI* (see 2.2.4) to the application component of the receiver.

Generating the Configuration Objects

To do so, proceed as described in variant 1.

When you have checked the generation log, close it and save the configuration scenario. Activate your change lists.

7.3 Execution

You perform the following steps to check that the process integration scenario is executed without errors.

To run the process integration scenario, you can use a transaction that is available in the SAP system of the Integration Server.

1. Logon to the SAP system of the Integration Server, client 105.
2. Choose transaction *SXIPATT1*.
3. Enter values for the fields *Name*, *Country* and *Tel*.
4. Choose *Send*.
5. To check whether the data are transferred correctly from the sender to the receiver, switch to the receiver view (pushbutton *Receiver View*).
When the message was sent without errors, the data are displayed that you entered in the sender view.



If you switch fast enough from sender to receiver view after having chosen *Send*, you can see the former data. Choose *Refresh* to check if the message was already processed and the data is updated accordingly.

8 Variant 6: One Sender and One Receiver using Advanced Adapter Engine

This variant enables you to configure and execute the simplest kind of message exchange:

One sender (file directory *XiPattern4*) sends a message to one receiver (file directory *XiPattern5*). In this way the message is exchanged between sender and receiver adapter using one single Adapter Engine and without using the ABAP proxy runtime. No mapping is executed between the outbound and inbound interface in this variant.

8.1 Design Objects Used

In this variant the same interfaces are used as in variant 1 (see 3.1). Note that no process integration scenario exists for this variant.

8.2 Configuring the Process Integration Scenario

You perform the following configuration steps in the Integration Directory.

Note that this scenario can run in parallel with variant 1.

For this scenario, the integrated configuration object is used for configuration. The model configurator is not used in this case.



For more information, see [Executing Integrated Configuration](#).

Calling the Integration Directory

First, call the Integration Directory. On the *Process Integration* initial screen, choose *Integration Directory*.

Creating a Configuration Scenario

Perform the following steps to create a configuration scenario to group the configuration objects.

1. In the Integration Directory main menu, choose *Object* → *New*.
2. Under *Administration* select *Configuration Scenario*.
3. Enter the name `XiPatternOneStepScenario_AAE2AAE`.
4. Choose *Create*.
5. Save the configuration scenario.

Creating the Communication Channels for Sender and Receiver

Perform the following steps to create the communication channels.

1. In the Integration Directory main menu, choose *Object* → *New*.
2. Under *Collaboration Profile* select *Communication Channel*.
3. Call the input help () for the Communication Component.
4. Select the communication component `<SID>_FileSystem_XiPattern4` and choose *Apply*.
5. In field *Communication Channel* enter `XiPatternChannel_FileSender2`.
6. Under *Add to Scenario* select `XiPatternOneStepScenario_AAE2AAE` (input help).
7. Choose *Create*.
The newly created communication channel is opened.
8. In the menu of the communication channel object editor select *Communication Channel* → *Apply Template*.
9. Select the template `XiPatternChannel_FileSender`.
10. Choose *Apply*.
11. Under *File Access Parameter* in the *Source Directory* field, enter the path in the file directory that you defined under 2.2.2. In the case of the sender communication channel, enter the whole path for the file directory `XiPattern4`.



Example: C:/<selected path>/XiPattern4

12. Save the communication channel.
13. Following the same procedure, create the communication channel *XiPatternChannel_FileReceiver2* (communication component <SID>_FileSystem_XiPattern5) using the template *XiPatternChannel_FileReceiver*. Enter the path for the file directory *XiPattern5*.
14. Save the communication channel.
15. Activate the change list.

Creating the Integrated Configuration

1. In the Integration Directory main menu, choose *Object* → *New*.
2. Under *Collaboration Agreement* select *Integrated Configuration*.
3. Call the input help () in the field *Communication Component*.
4. Select the communication component <SID>_FileSystem_XiPattern4 and choose *Apply*.
5. In the fields *Interface* and *Namespace* enter *XiPatternInterface1* and <http://sap.com/xi/XI/System/Patterns>, respectively.
6. Under *Add to Scenario* select *XiPatternOneStepScenario_AAE2AAE* (input help).
7. Choose *Create*.
8. In tab *Inbound Processing* select the communication channel *XiPatternChannel_FileSender2*.
9. In the tab *Receiver Interfaces* select the communication component <SID>_FileSystem_XiPattern5.
10. For *Name* and *Namespace* enter *XiPatternInterface1* and <http://sap.com/xi/XI/System/Patterns>, respectively.
11. In the tab *Outbound Processing* select the communication channel *XiPatternChannel_FileReceiver2*.
12. Save the integrated configuration.

Activating the Change List

1. To do this, select the *Change Lists* tab in the Integration Directory navigation area.
2. Choose *Open* → *Standard Change List*.
3. Position the cursor on the *Standard Change List* node and choose *Activate* () in the context menu.

Executing the Use Case

You perform the following steps to check that the use case is executed without errors.

1. On the host of the Integration Server, open the file directory *XiPattern4*.
2. Rename the file `ZXiPatternSenderFile1.xml` as `XiPatternSenderFile1.xml`.

The File/FTP adapter is designed in a way that a file with exactly this name is recognized and moved from the source directory *XiPattern4* to the target directory

XiPattern5 every 60 seconds. If the configuration was executed without errors, the file will be moved from the source directory 60 seconds after it is renamed.

3. Open the directory *XiPattern5* and check that the file has arrived.
4. Open the file and check whether the content has changed (compare with the original file *ZXiPatternSenderFile1.xml*).

The file must not have been changed when this variant is executed.

Furthermore, you can check the processing of the message in monitoring. To do this, proceed as follows:

1. Start the Runtime Workbench by choosing *Runtime Workbench* on the Process Integration start page.
2. Choose *Message Monitoring*.
3. Select Adapter Engine <SID> as the component and then choose *Display*.
4. Use appropriate filter criteria to restrict the number of XML messages displayed (for example, the processing period).
5. Choose *Start*.

The system displays the selected messages in a table. You want to locate the messages that were sent during execution.

6. Select the message to be displayed by selecting the appropriate radio button and then choose *Details*.

The tab *Message Contents* displays the message contents including all message versions for the various phases of message processing.

Note that the message can not be seen in the Integration Server as the message is sent directly from the sender adapter to the receiver adapter.