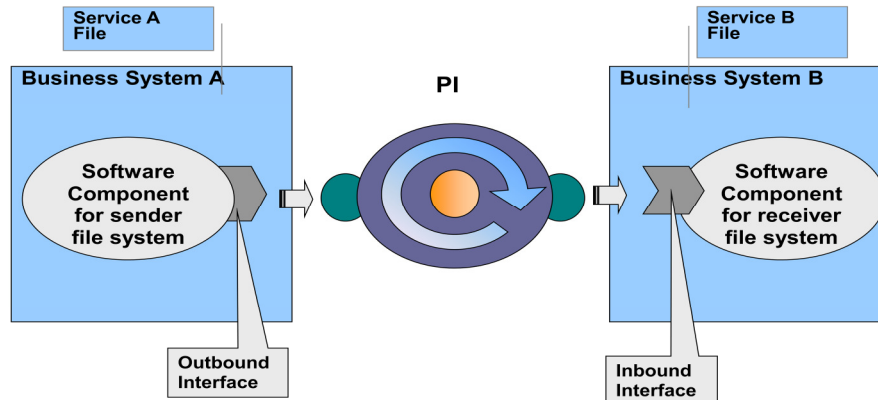


Exercise – File to File :



Pre- requisite to start this development-

Integration Repository - Software Components defined for Sender and Receiver File System in SLD

Business Logic behind this development

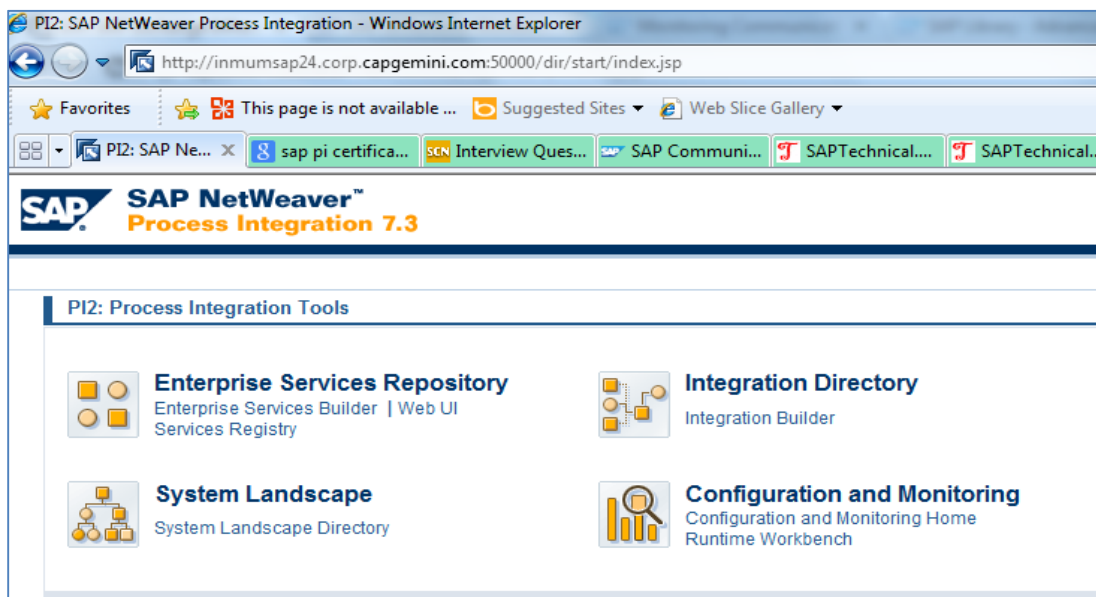
Sender File system will place their XML file on FTP server. SAP PI picks up the file and does require mapping. The result of mapping needs to be sent to the Receiver (File) system as XML File.

Design

1. Access to the Process Integration Tools

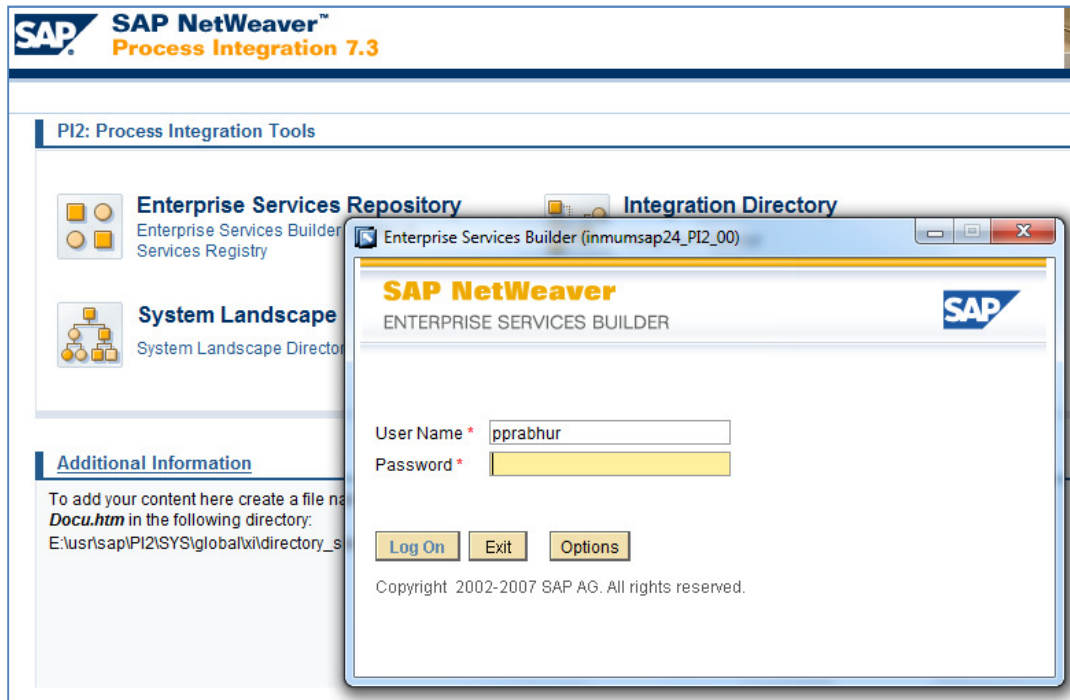
Link provided

For Eg -<http://inmumsap24.corp.capgemini.com:50000/dir/start/index.jsp>



2. Access Enterprise Service Repository(ESR)

Click on the link Enterprise Service Builder under Enterprise Service Repository. The following screen appears –



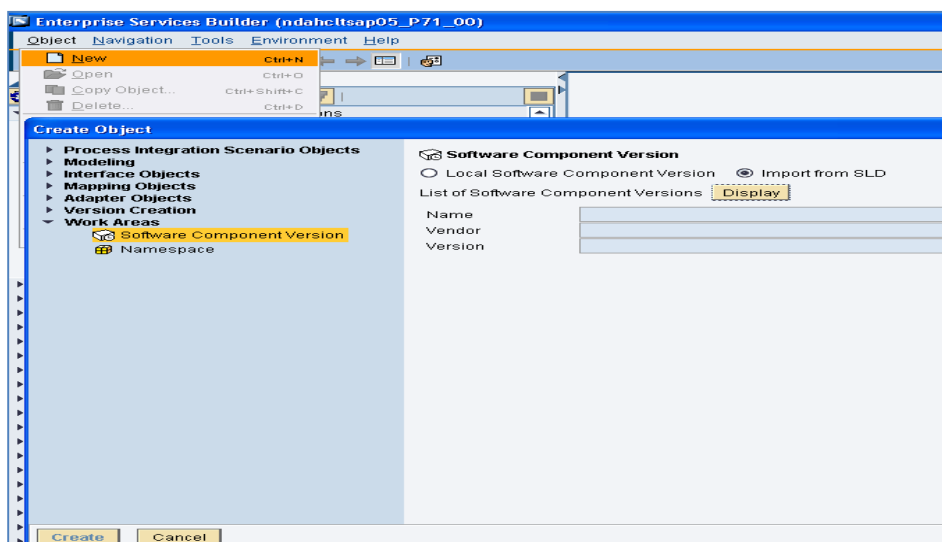
Login to PI using your user id and password.

3. Import the software component define in SLD

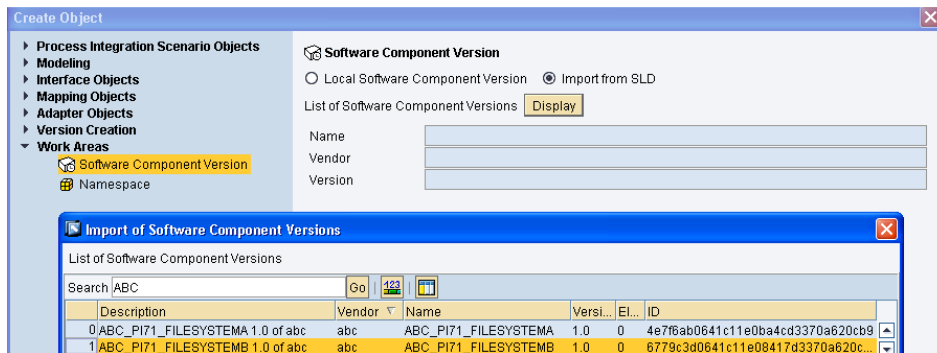
Import the following software component defined for the scenario into ESR

1. ABC_PI71_FILESYSTEMA 1.0 of abc
2. ABC_PI71_FILESYSTEMB 1.0 of abc

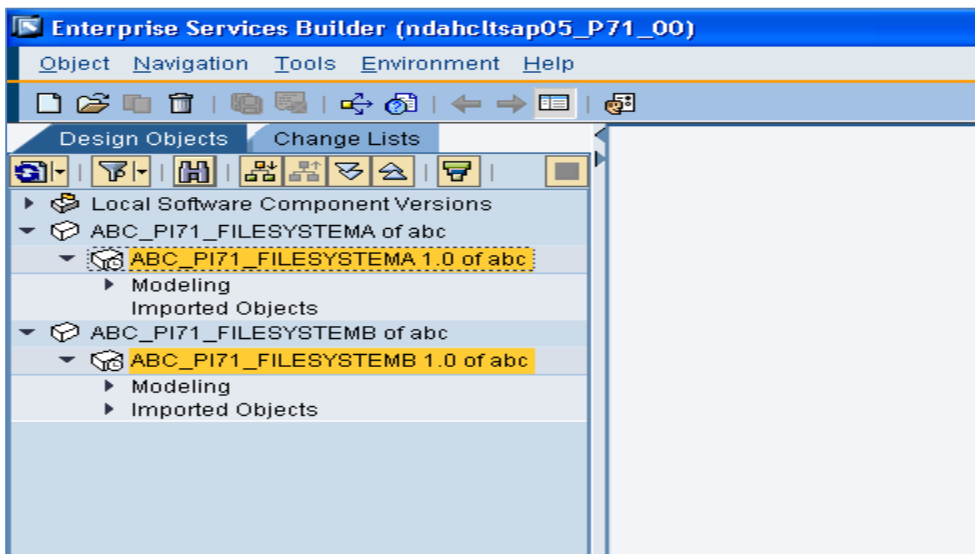
Click on new under object and select SWCV under Work Areas-



Click on Display and select the required sender and receiver business system to be imported into ESR for further development



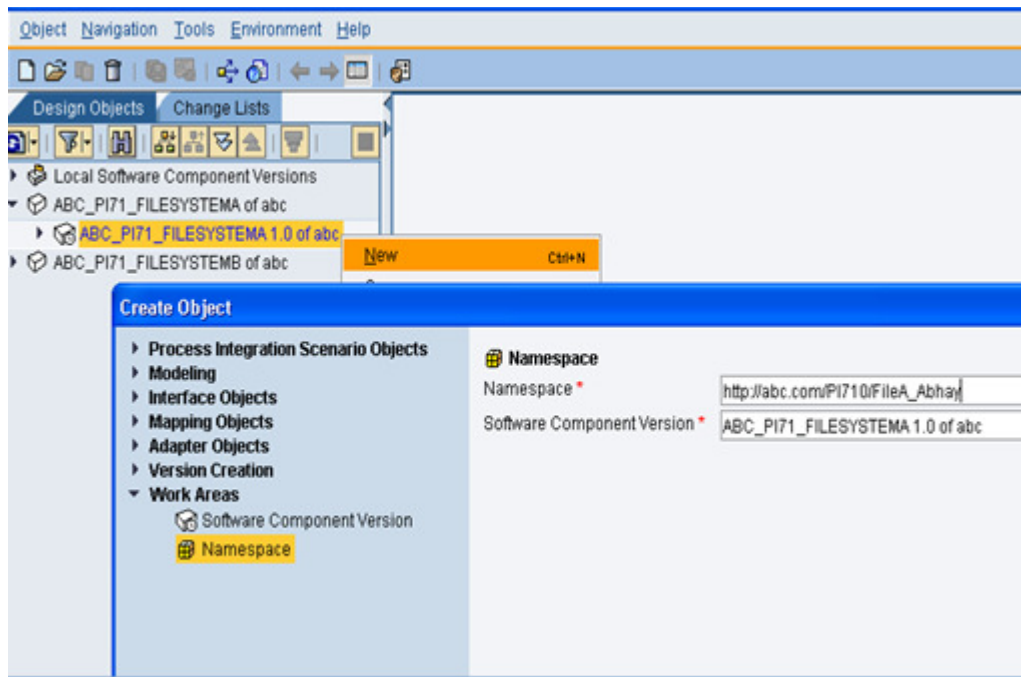
Below is how the SWCV look like after getting imported into ESR-



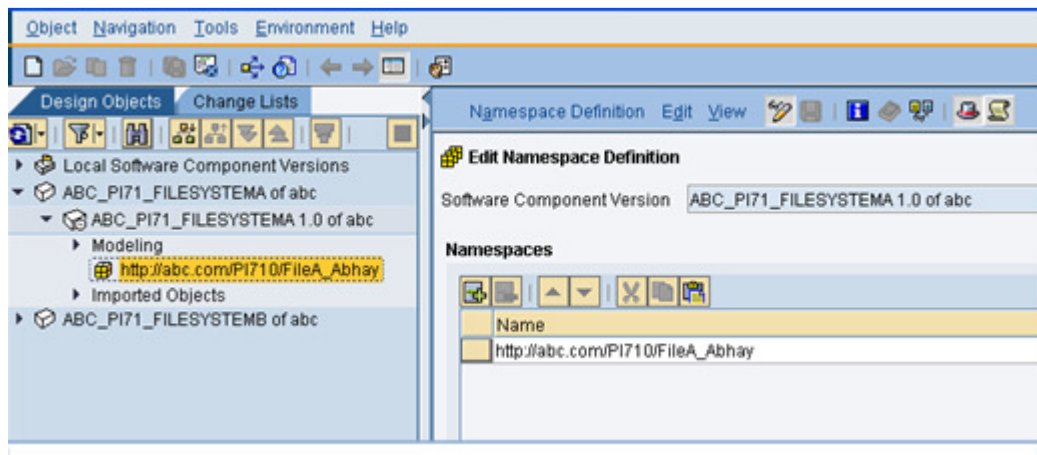
4. Define Namespace

1. Namespace (http://<CompanyName>.com/PI710/FileA_<Name/EmpNo>) for Sender SWCV
2. Namespace (http://<CompanyName>.com/PI710/FileB_<Name/EmpNo>) for Receiver SWCV

You Define a Namespace for you SWCV that uniquely identifies all the objects you create in Repository. Right click software component version and select Namespace from the context.

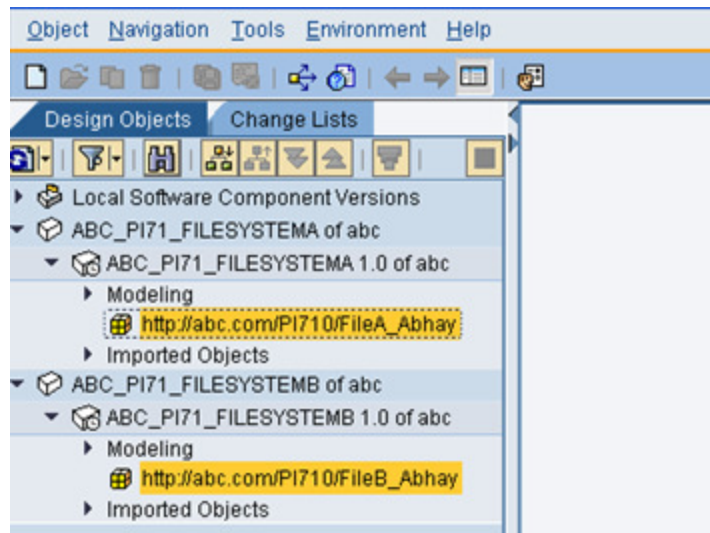


After creating and saving it appears like –



Activate the changes made in the objects by going to the Change Lists tab(left navigation bar). Click on the software component “`http://<CompanyName>.com/PI710/FileA_<Name/EmpNo>`” and click on the refresh icon. Similarly need to create namespace for receiver SWCV ““`http://<CompanyName>.com/PI710/FileB_<Name/EmpNo>`”.

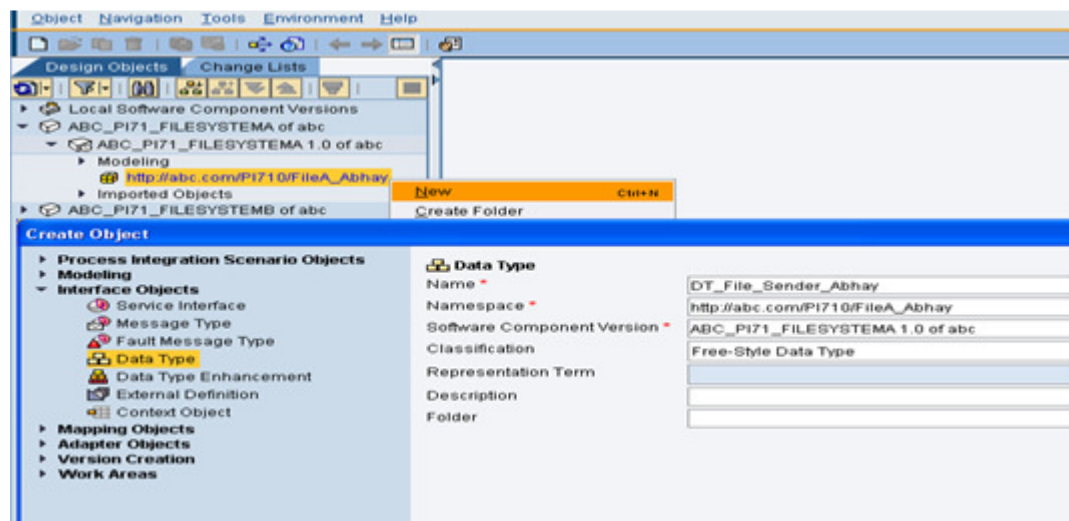
The ESR screen will look as below once the namespaces are created for sender and receiver system.



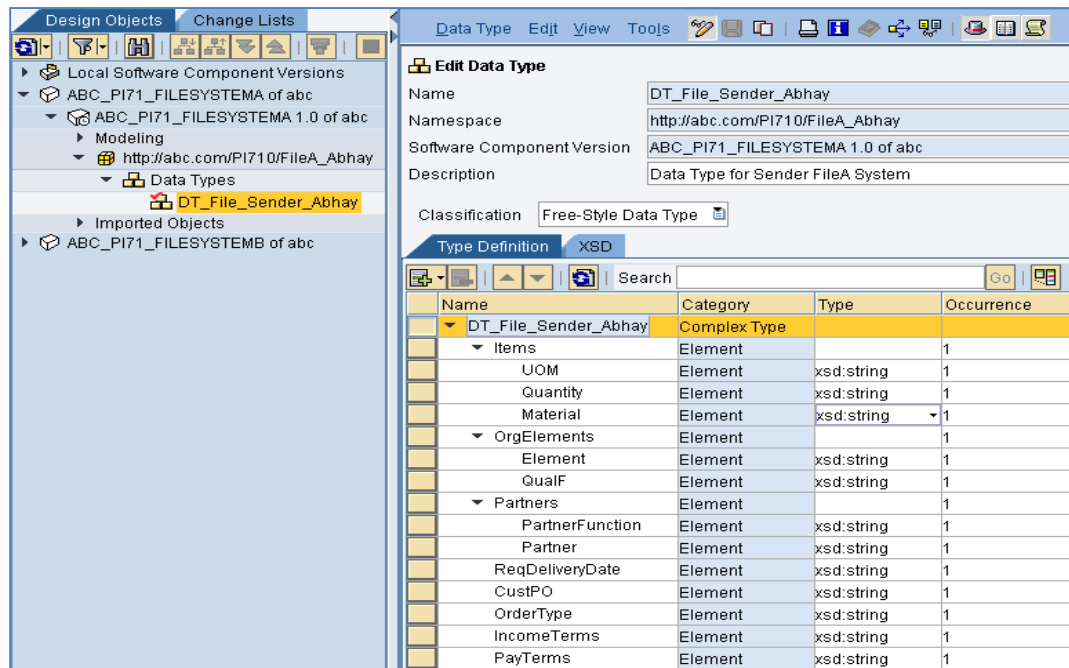
5. Define Data Type

1. Date type(DT_File_Sender_<Name/EmpNo> for Sender System (FileSystemA)
2. Date type(DT_File_Receiver_< Name/EmpNo >) for Receiver System (FileSystemB)

Create data types with required structure and type of elements as below. Right click on the namespace and add new Data type under Interface objects context. The following screen appears. Enter the name of the Data type.



Create DT_File_Sender_< Name/EmpNo > for the outbound message from the sender FileA System. Define the structure of your data type with nodes and fields and as depicted below and save upon completion if the data type.



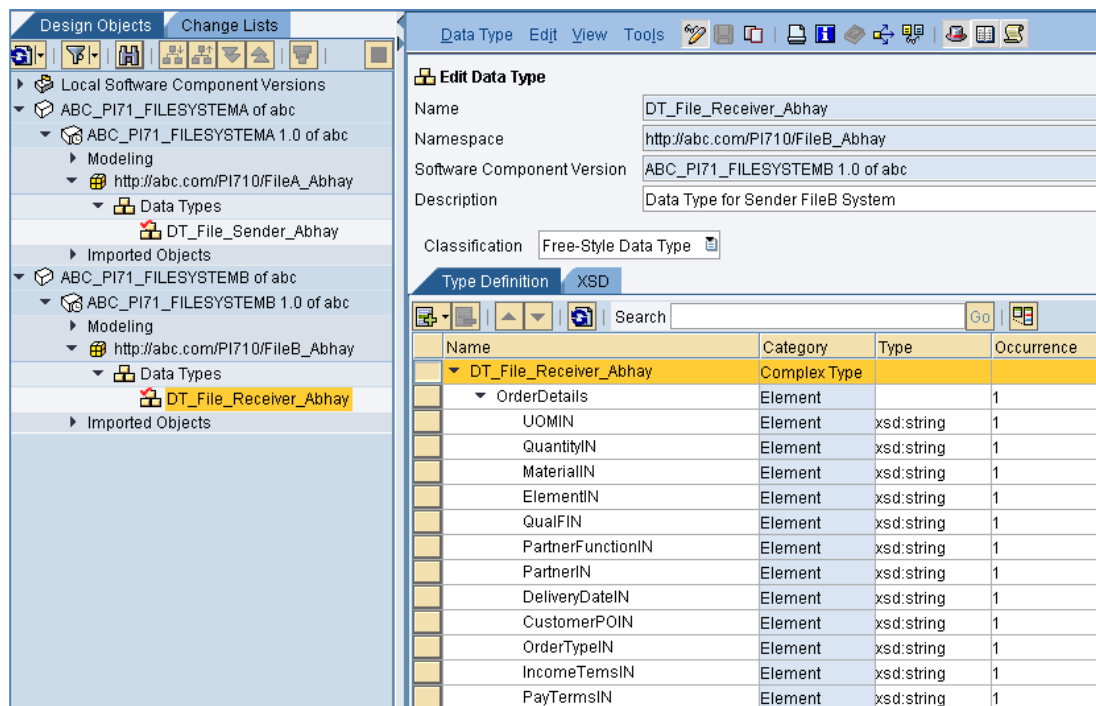
Edit Data Type

Name: DT_File_Sender_Abhay
 Namespace: http://abc.com/PI710/FileA_Abhay
 Software Component Version: ABC_PI71_FILESYSTEMA 1.0 of abc
 Description: Data Type for Sender FileA System
 Classification: Free-Style Data Type

Type Definition | XSD

Name	Category	Type	Occurrence
DT_File_Sender_Abhay	Complex Type		
Items	Element		1
UOM	Element	xsd:string	1
Quantity	Element	xsd:string	1
Material	Element	xsd:string	1
OrgElements	Element		1
Element	Element	xsd:string	1
QualF	Element	xsd:string	1
Partners	Element		1
PartnerFunction	Element	xsd:string	1
Partner	Element	xsd:string	1
ReqDeliveryDate	Element	xsd:string	1
CustPO	Element	xsd:string	1
OrderType	Element	xsd:string	1
IncomeTerms	Element	xsd:string	1
PayTerms	Element	xsd:string	1

Similarly create DT_File_Receiver_< Name/EmpNo > for the inbound message to the receiver FileB System. Save after completion of the data type.



Edit Data Type

Name: DT_File_Receiver_Abhay
 Namespace: http://abc.com/PI710/FileB_Abhay
 Software Component Version: ABC_PI71_FILESYSTEMB 1.0 of abc
 Description: Data Type for Sender FileB System
 Classification: Free-Style Data Type

Type Definition | XSD

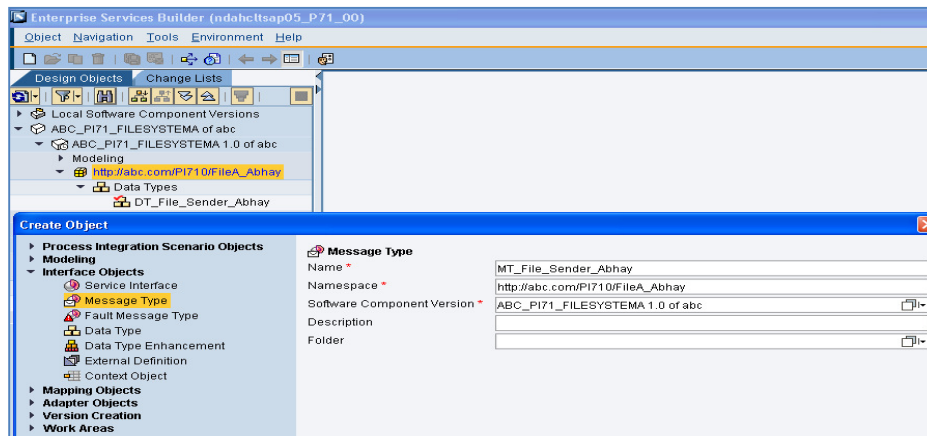
Name	Category	Type	Occurrence
DT_File_Receiver_Abhay	Complex Type		
OrderDetails	Element		1
UOMIN	Element	xsd:string	1
QuantityIN	Element	xsd:string	1
MaterialIN	Element	xsd:string	1
ElementIN	Element	xsd:string	1
QualFIN	Element	xsd:string	1
PartnerFunctionIN	Element	xsd:string	1
PartnerIN	Element	xsd:string	1
DeliveryDateIN	Element	xsd:string	1
CustomerPOIN	Element	xsd:string	1
OrderTypeIN	Element	xsd:string	1
IncomeTemsIN	Element	xsd:string	1
PayTermsIN	Element	xsd:string	1

6. Define Message Type

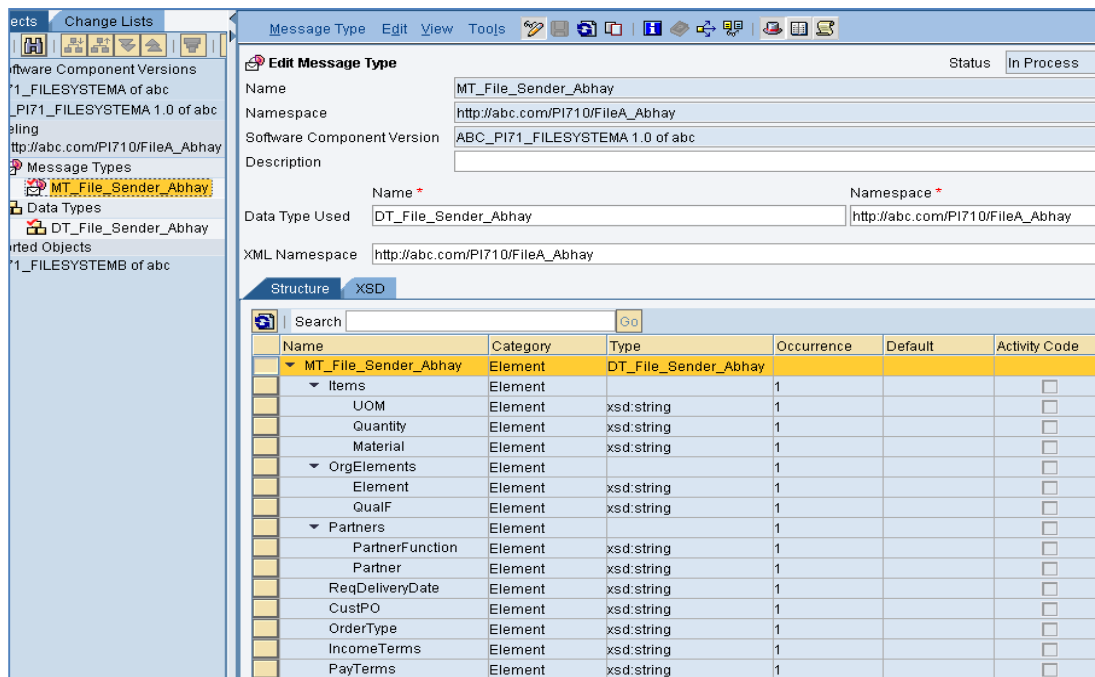
1. Message type(MT_File_Sender_<Name/EmpNo> for Sender System (FileSystemA).
2. Message type(MT_File_Receiver_< Name/EmpNo >) for Receiver System (FileSystemB)

Create MT_File_Sender_<Name/EmpNo> for the message coming from sender FileSytemA and MT_File_Receiver_< Name/EmpNo > for the message to be sent to receiver FileSystemB based on the following steps-

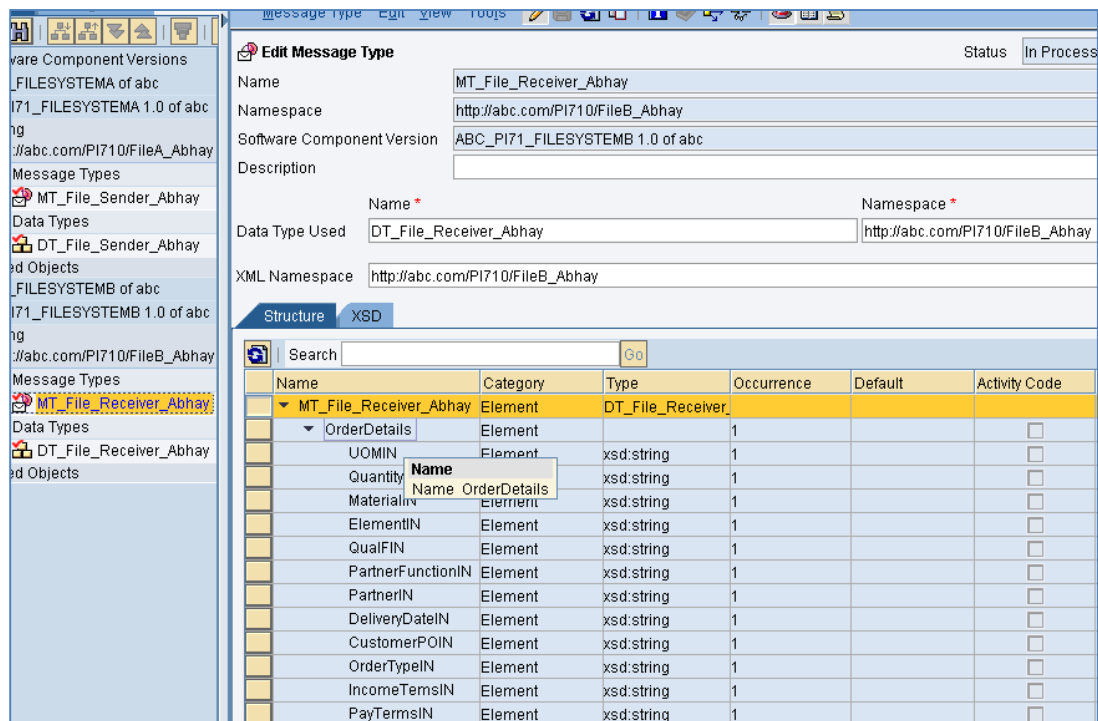
Right click on the namespace and add new Message type under Interface objects context.



Select the Data type (created in the previous step) for the message type. The Namespace automatically gets populated. This is the message type for sender system. Save after completing.



Similarly create MT_File_Receiver_< Name/EmpNo > for the receiver FileB System as below-



Edit Message Type Status: In Process

Name: MT_File_Receiver_Abhay
 Namespace: http://abc.com/PI710/FileB_Abhay
 Software Component Version: ABC_PI71_FILESYSTEMB 1.0 of abc
 Description:

Data Type Used: DT_File_Receiver_Abhay Namespace: http://abc.com/PI710/FileB_Abhay
 XML Namespace: http://abc.com/PI710/FileB_Abhay

Structure XSD

Name	Category	Type	Occurrence	Default	Activity Code
MT_File_Receiver_Abhay	Element	DT_File_Receiver	1		
OrderDetails	Element		1		
UOMIN	Element	xsd:string	1		
Quantity	Element	xsd:string	1		
MaterialIN	Element	xsd:string	1		
ElementIN	Element	xsd:string	1		
QualFIN	Element	xsd:string	1		
PartnerFunctionIN	Element	xsd:string	1		
PartnerIN	Element	xsd:string	1		
DeliveryDateIN	Element	xsd:string	1		
CustomerPOIN	Element	xsd:string	1		
OrderTypeIN	Element	xsd:string	1		
IncomeTermsIN	Element	xsd:string	1		
PayTermsIN	Element	xsd:string	1		

7. Create Service Interfaces

1. Service Interface(SI_File_Sender_Out_< Name/EmpNo>) for Sender System (FileSystemA).
2. Service Interface(SI_File_Receiver_In_< Name/EmpNo>) for Sender System (FileSystemB).

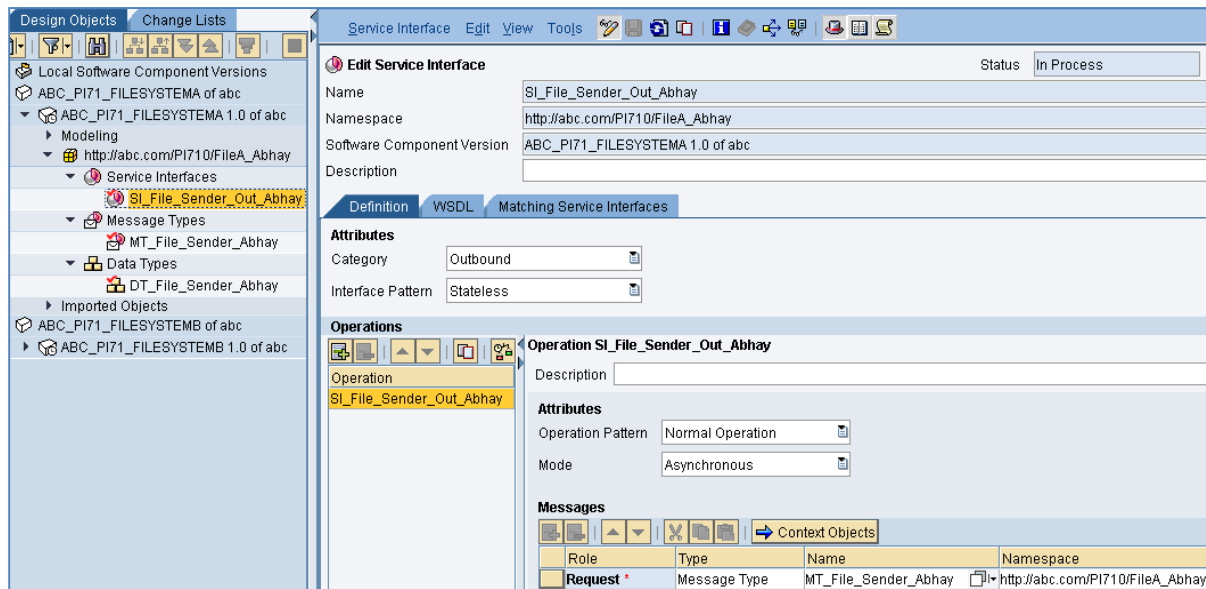
Service Interfaces specify the Mode(Sync/Async), direction(inbound/outbound) and the corresponding Message Type. Define two Service Interfaces using the attributes given below. Save after creating both service interfaces.

Service Interface	Message Type	Category	Mode	Operation Pattern
SI_File_Sender_Out_< Name/EmpNo>	MT_File_Sender_<Name/EmpNo>	Outbound	Asynchronous	Stateless
SI_File_Receiver_In_< Name/EmpNo>	MT_File_Receiver_<Name/EmpNo>	Inbound	Asynchronous	Stateless

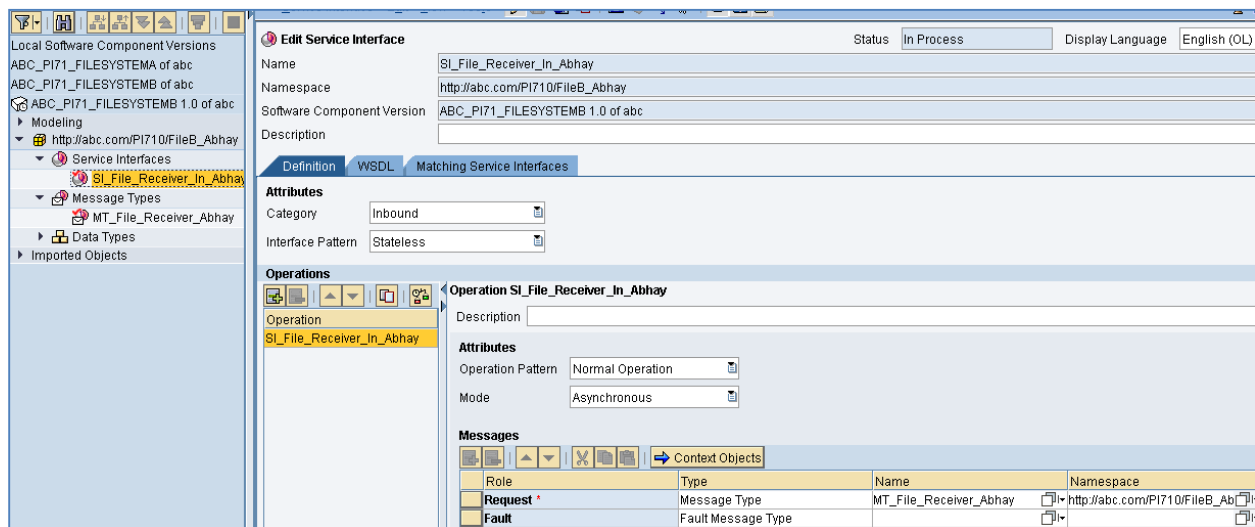
Note : Inbound and Outbound Category is the point of view of the Business Systems being integrated and not that of PI.

The outbound and inbound Service Interfaces are shown below -

This the service interface for the sender system. Save after completing



This is the service interface for the receiver system. Save after completing

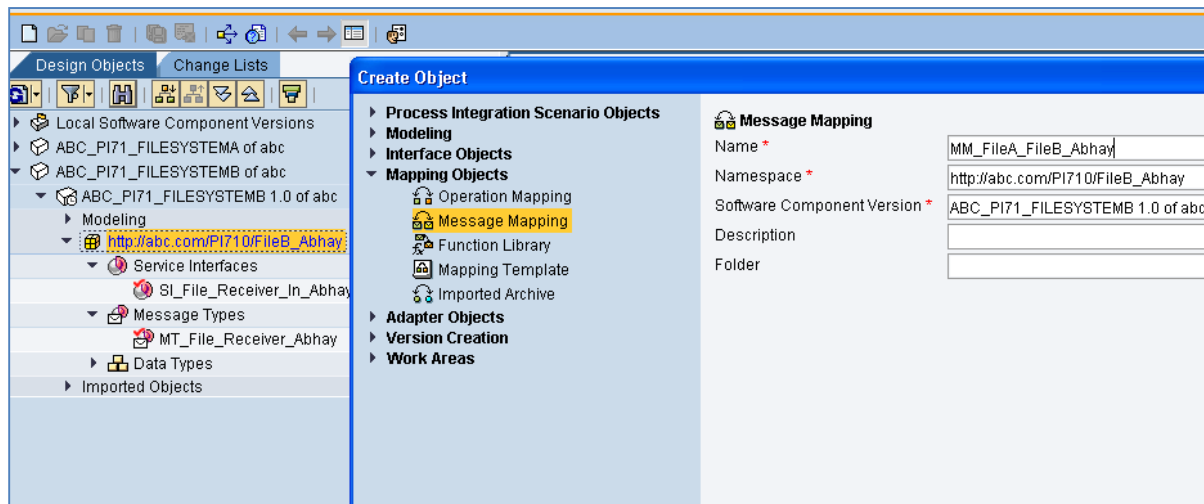


Leave the Fault Message Type as empty as we are not doing any error handling. Activate the changes made in objects by going to change lists tab.

8. Create the Message Mapping

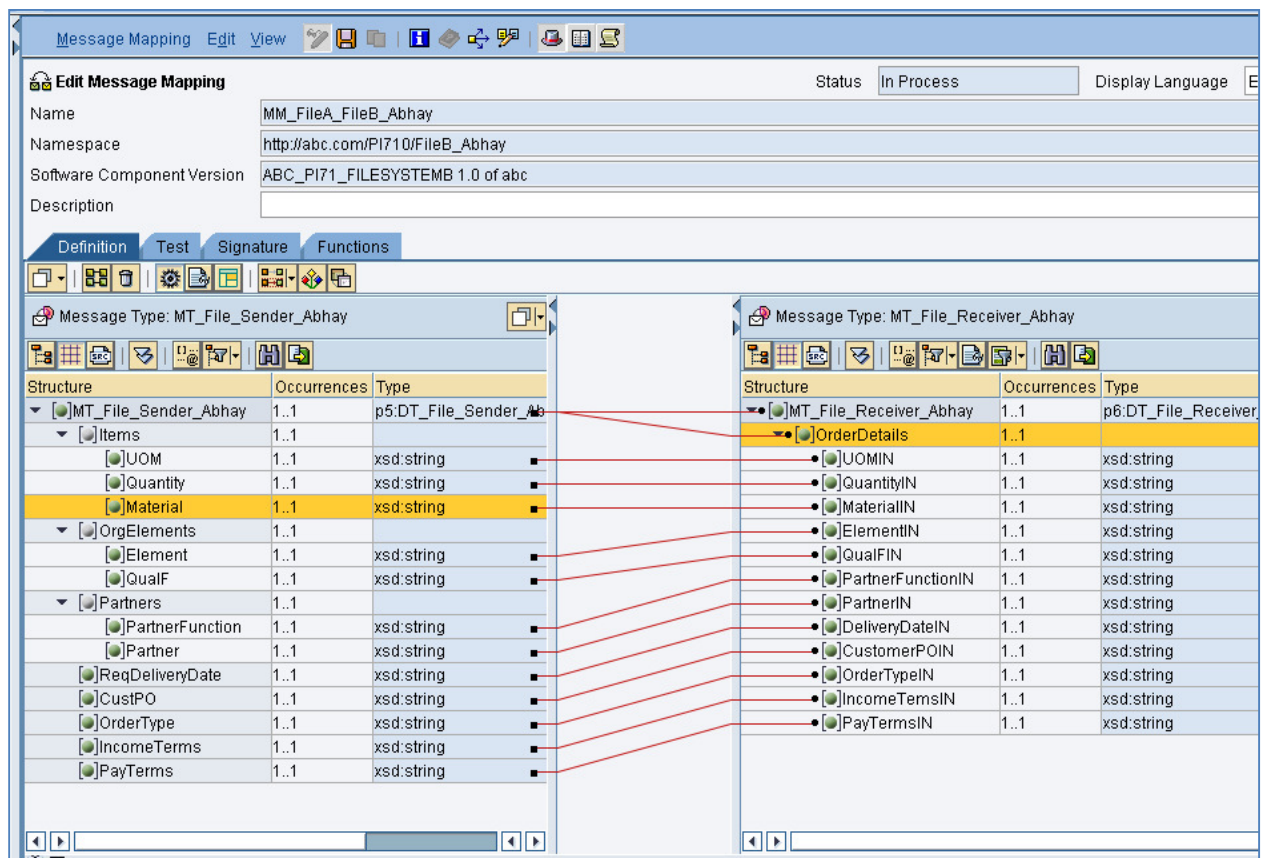
1. MM_FileA_FileB_<Name/EmpNo> in the receiver File SystemB

Create a new Message mapping by right clicking the receiver SWCV and adding selecting message mapping under Mapping objects context as below –



Mapping object transforms Data from one Message Type to another message type.

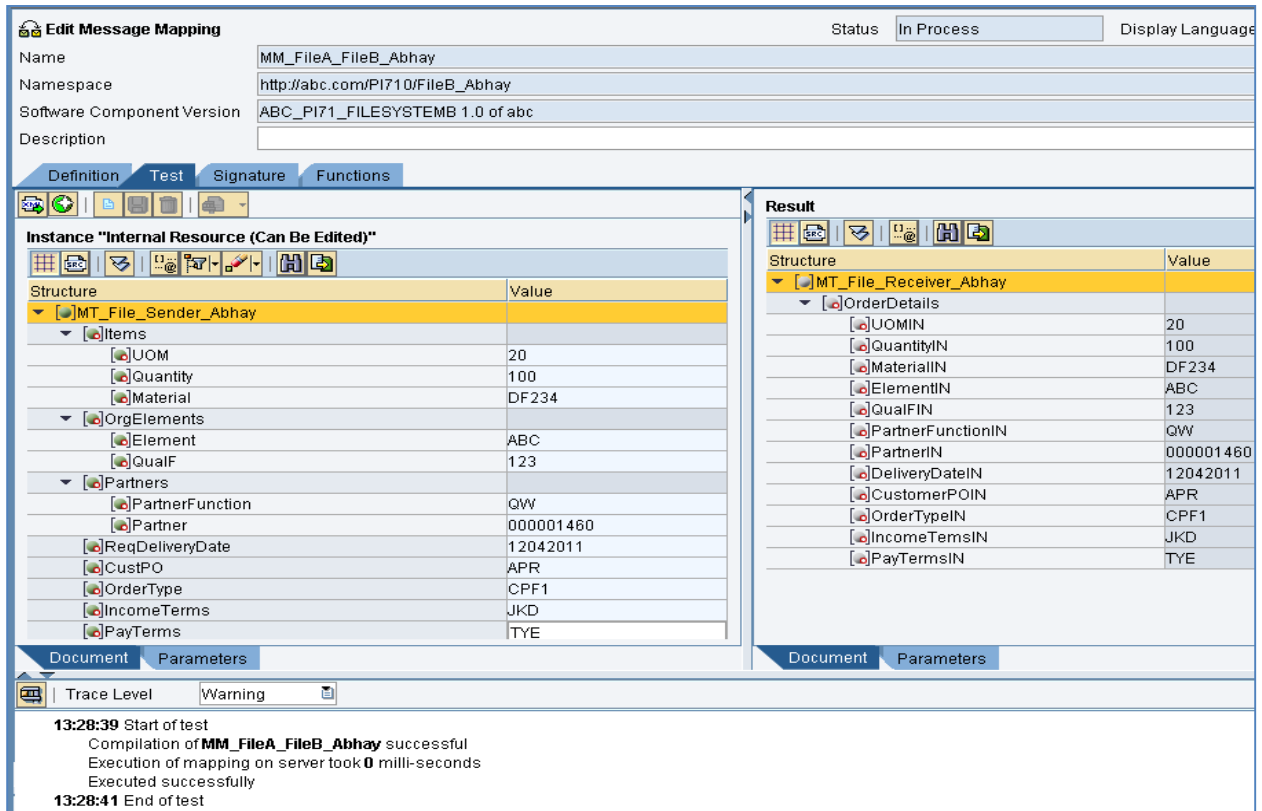
The mapping in this scenario is one-to-one. The names of the source field and target fields are identical. To perform a mapping, drag the source fields and drop on the target field. The mapping would be displayed in the graphical display.



You can test your mapping by going to test Tab. You can select either of the two buttons from the tool bar to test your scenario.

1. **Load XM Instance** – load an XML file that contains your test data.
2. **Generate Instance** – By specifying the value manually.

Now you can test the scenario by clicking the Execute Mapping.



Edit Message Mapping Status: In Process Display Language

Name: MM_FileA_FileB_Abhay
 Namespace: http://abc.com/PI710/FileB_Abhay
 Software Component Version: ABC_PI71_FILESYSTEMB 1.0 of abc
 Description:

Definition Test Signature Functions

Instance "Internal Resource (Can Be Edited)"

Structure	Value
MT_File_Sender_Abhay	
Items	
UOM	20
Quantity	100
Material	DF234
OrgElements	
Element	ABC
QualF	123
Partners	
PartnerFunction	QW
Partner	000001460
ReqDeliveryDate	12042011
CustPO	APR
OrderType	CPF1
IncomeTerms	JKD
PayTerms	TYE

Result

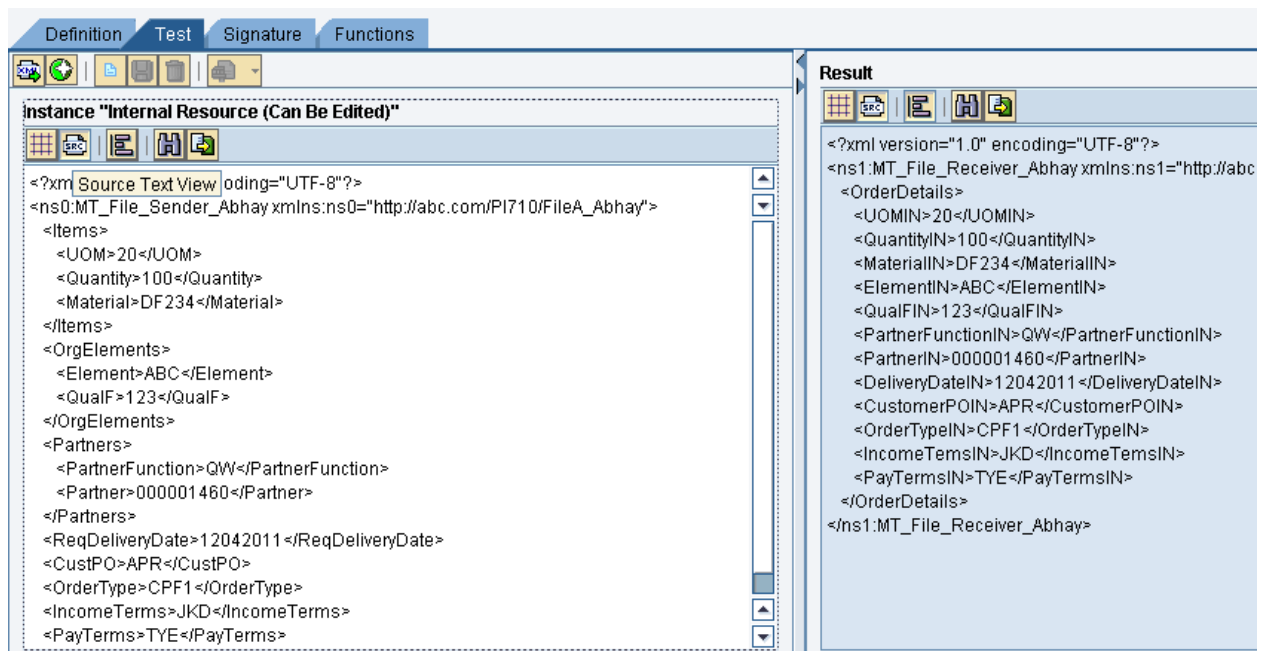
Structure	Value
MT_File_Receiver_Abhay	
OrderDetails	
UOMIN	20
QuantityIN	100
MaterialIN	DF234
ElementIN	ABC
QualFIN	123
PartnerFunctionIN	QW
PartnerIN	000001460
DeliveryDateIN	12042011
CustomerPOIN	APR
OrderTypeIN	CPF1
IncomeTermsIN	JKD
PayTermsIN	TYE

Document Parameters

Trace Level: Warning

13:28:39 Start of test
 Compilation of MM_FileA_FileB_Abhay successful
 Execution of mapping on server took 0 milli-seconds
 Executed successfully
 13:28:41 End of test

The following is the source xml that will be generated as the result of your test values. We can view it by clicking on the **src** icon.



Definition Test Signature Functions

Instance "Internal Resource (Can Be Edited)"

```
<?xml Source Text Viewoding="UTF-8"?>
<ns0:MT_File_Sender_Abhay xmlns:ns0="http://abc.com/PI710/FileA_Abhay">
  <Items>
    <UOM>20</UOM>
    <Quantity>100</Quantity>
    <Material>DF234</Material>
  </Items>
  <OrgElements>
    <Element>ABC</Element>
    <QualF>123</QualF>
  </OrgElements>
  <Partners>
    <PartnerFunction>QW</PartnerFunction>
    <Partner>000001460</Partner>
  </Partners>
  <ReqDeliveryDate>12042011</ReqDeliveryDate>
  <CustPO>APR</CustPO>
  <OrderType>CPF1</OrderType>
  <IncomeTerms>JKD</IncomeTerms>
  <PayTerms>TYE</PayTerms>
</ns0:MT_File_Sender_Abhay>
```

Result

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:MT_File_Receiver_Abhay xmlns:ns1="http://abc.com/PI710/FileB_Abhay">
  <OrderDetails>
    <UOMIN>20</UOMIN>
    <QuantityIN>100</QuantityIN>
    <MaterialIN>DF234</MaterialIN>
    <ElementIN>ABC</ElementIN>
    <QualFIN>123</QualFIN>
    <PartnerFunctionIN>QW</PartnerFunctionIN>
    <PartnerIN>000001460</PartnerIN>
    <DeliveryDateIN>12042011</DeliveryDateIN>
    <CustomerPOIN>APR</CustomerPOIN>
    <OrderTypeIN>CPF1</OrderTypeIN>
    <IncomeTermsIN>JKD</IncomeTermsIN>
    <PayTermsIN>TYE</PayTermsIN>
  </OrderDetails>
</ns1:MT_File_Receiver_Abhay>
```

Copy the file and save it on your local machine. This will be later used for testing the scenario end to end.

9. Create the Operation Mapping

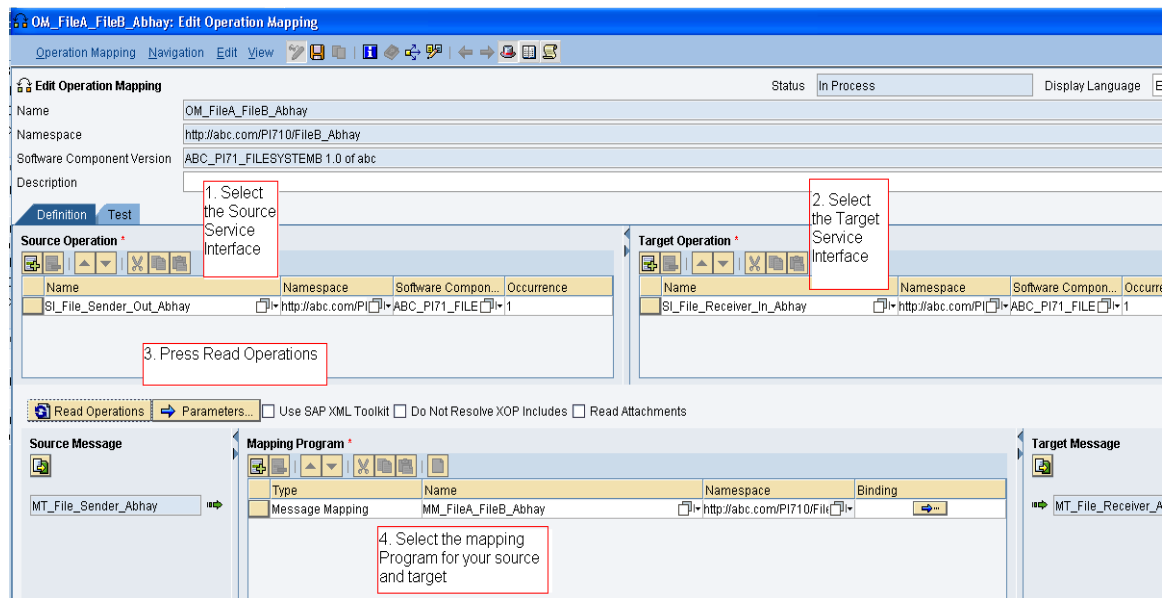
1. OM_FileA_FileB_<Name/EmpNo> in the receiver File SystemB.

In the operation mapping,

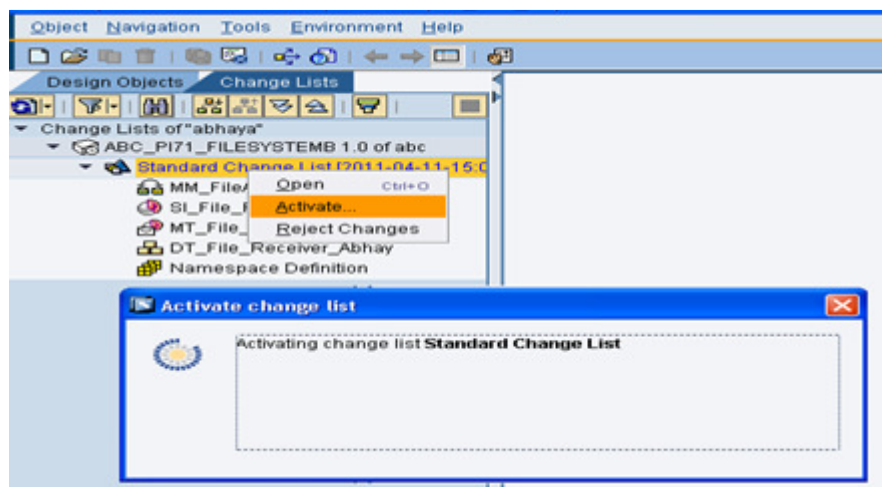
Select the source interface as “SI_File_Sender_Out_< Name/EmpNo>” and target interface as “SI_File_Receiver_In_< Name/EmpNo>”.

Click on the Read Interfaces tab and source/target message types automatically get populated.

Select the message mapping as “MM_FileA_FileB_<Name/EmpNo> “



Activate the changes made in the objects by going to the change list tab. You can see the change list under your SWCV. Right click on the standard change list and select Activate from the context.

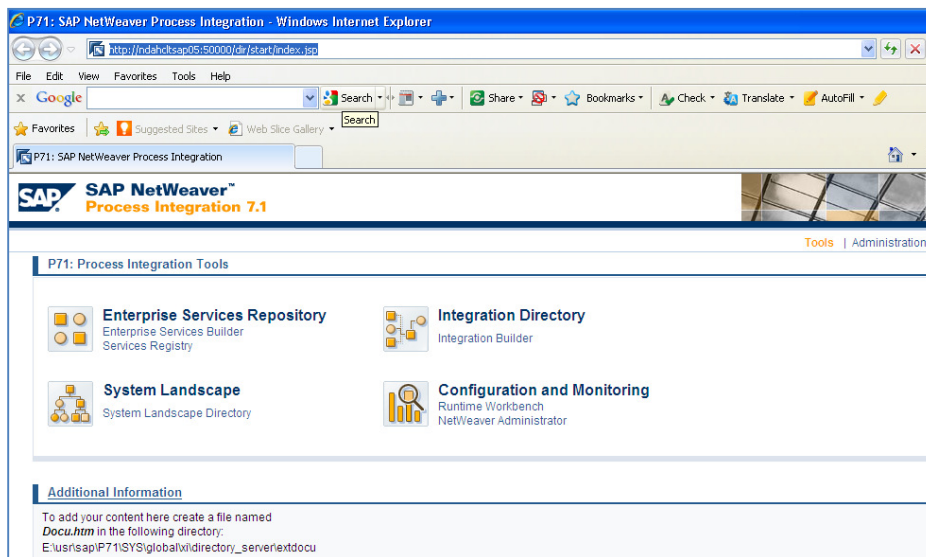


Configuration

1. Access to the Process Integration Tools

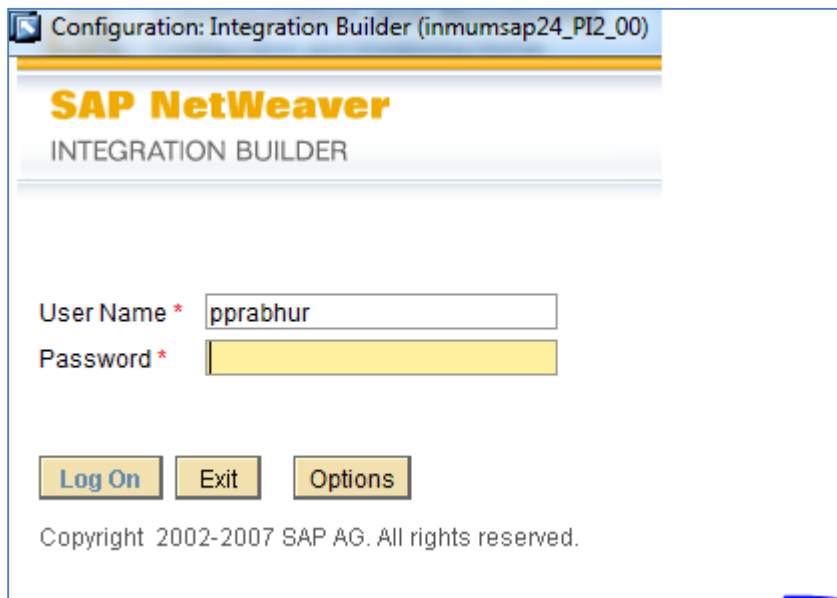
Link provided.

For Eg - <http://inmumsap24.corp.capgemini.com:50000/dir/start/index.jsp>



2. Access Integration Directory

Click on the link Integration Builder under Integration Directory. The following screen appears –

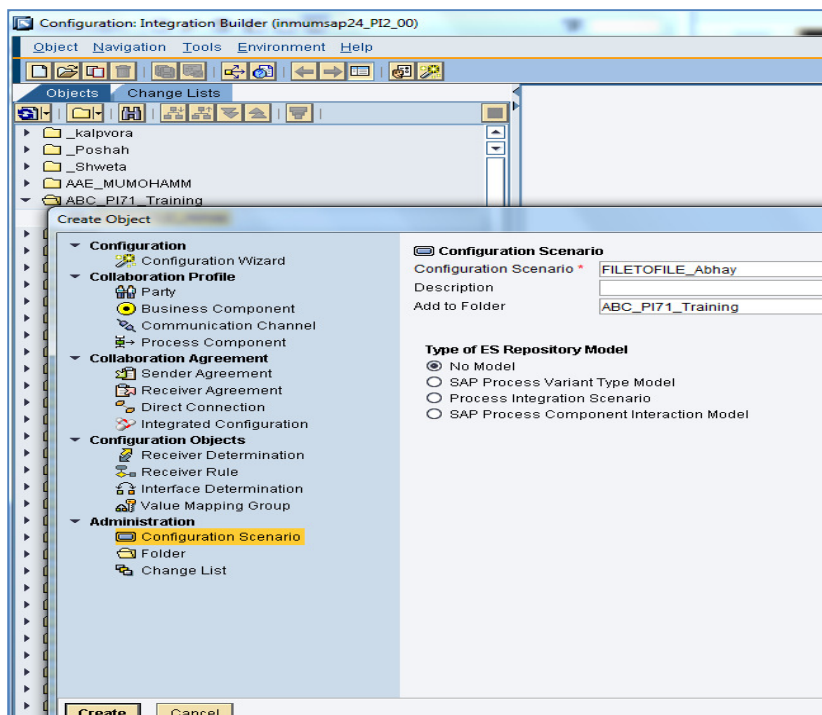
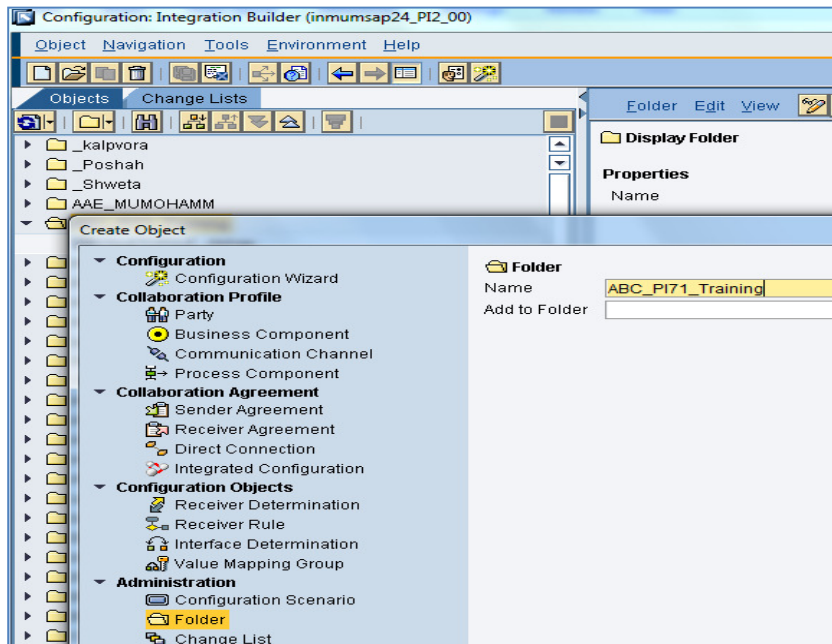


Login to PI using your user id and password.

3. Create a new Scenario.

scenario is a place holder for the configuration of interfaces. A scenario can hold the configuration for multiple interfaces.

Create a Configuration Scenario FILETOFILE_Abhay and add it to the Folder ABC_Pi71_Training.



4. Create Business Component for Sending and receiving File system

Business component is same as Business Service is PI7.0.

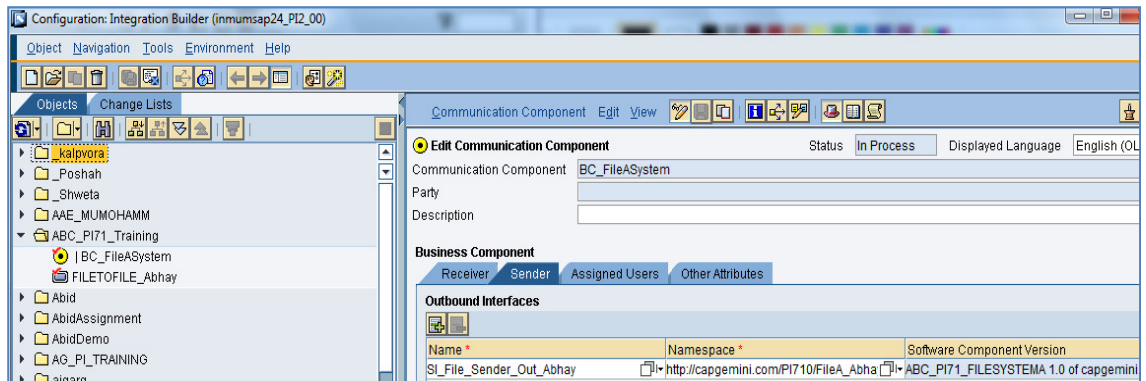
for this exercise we are going to define Business components.

5. Assign the Service Interfaces to Business Component.

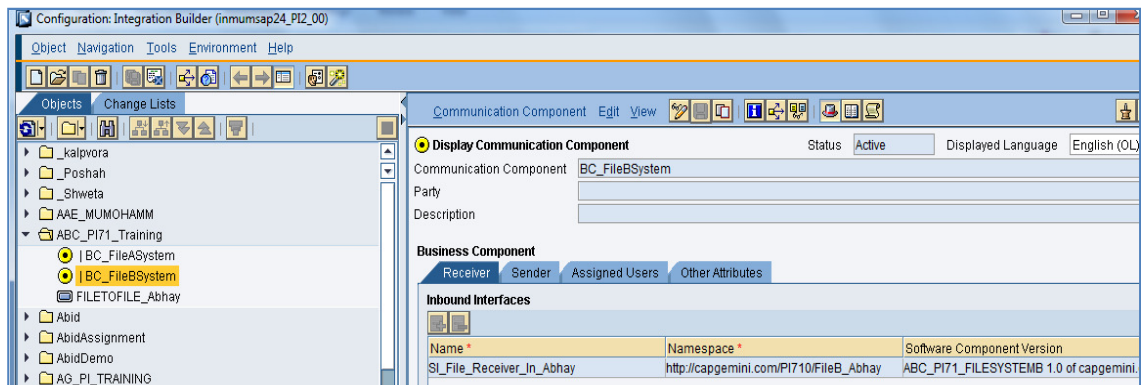
Note : This step is required only when we are using business component and not the business systems. In case of Business systems, there is reference to Software components defined in SLD.

But if we are using Business components, we need to explicitly add respective inbound/outbound service interfaces to the business components created.

Sender Service Component -



Receiver Business component –



6. Create Communication Channels

This communication channel will enable the business component/business system and integration server to communicate to each other.

Sender Communication Channel –

Communication Channel			
<div> Display Communication Channel Status: Active Displayed Language: English (OL) </div>			
Communication Channel	CC_File_Sender_Abhay		
Party	*		
Communication Component	BC_FileASystem		
Description	Communication Channel For File Sender		
<div> Parameters Identifiers Module </div>			
Adapter Type *	File	http://sap.com/xi/XI/System	SAP BASIS 7.11
<input checked="" type="radio"/> Sender <input type="radio"/> Receiver			
Transport Protocol *	File Transfer Protocol (FTP)		
Message Protocol *	File		
Adapter Engine *	Central Adapter Engine		
<div> Source Processing Advanced </div>			
File Access Parameters			
Source Directory *	\\PITraining\OUT\		
File Name *	xi_input_abhay.xml		
<input type="checkbox"/> Advanced Selection for Source File			
FTP Connection Parameters			
Server *	10.75.65.148		
Port *	21		
Data Connection	Passive		
Timeout (secs)			
Connection Security *	None		
<input type="checkbox"/> Anonymous Login			
User Name	trainee1		
Password	<div> <div>.....</div> <div>=</div> <div>.....</div> </div>		
Connect Mode *	Permanently		
Transfer Mode *	Binary		

Receiver Communication Channel-

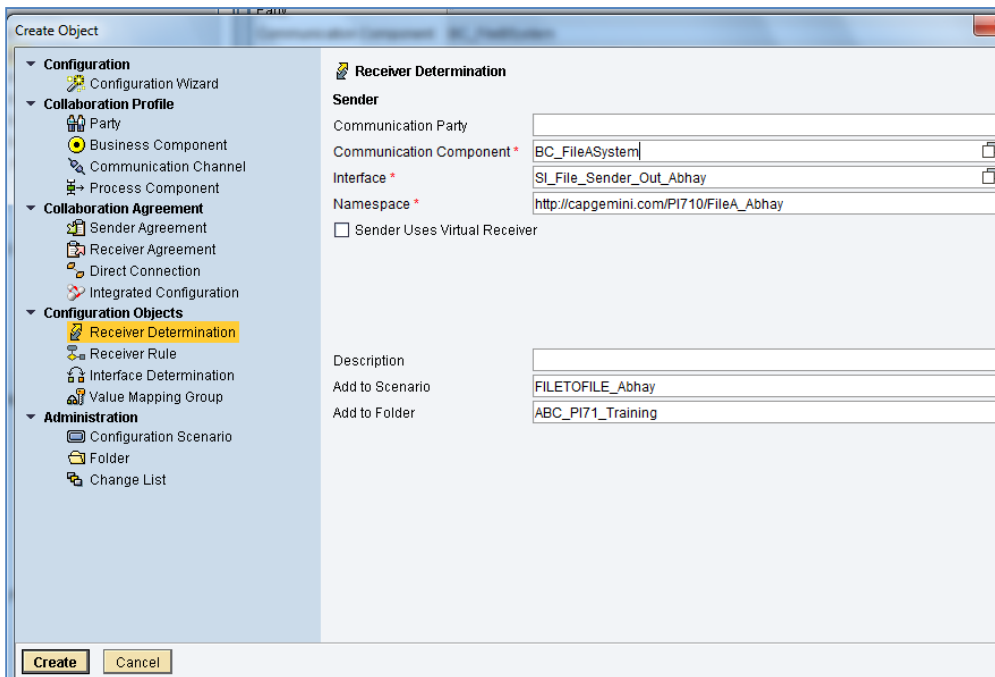
Communication Channel			
<div> Communication Channel Edit View </div>			
<div> Display Communication Channel Status: Active Displayed Language: English (OL) </div>			
Communication Channel	CC_File_Receiver_Abhay		
Party	*		
Communication Component	BC_FileBSysSystem		
Description	Communication Channel For File Receiver		
<div> Parameters Identifiers Module </div>			
Adapter Type *	File	http://sap.com/xi/XI/System	SAP BASIS 7.11
<div> <input type="radio"/> Sender <input checked="" type="radio"/> Receiver </div>			
Transport Protocol *	File Transfer Protocol (FTP)		
Message Protocol *	File		
Adapter Engine *	Central Adapter Engine		
<div> Target Processing Advanced </div>			
File Access Parameters			
Target Directory *	\\PITraining\OUT\		
<input checked="" type="checkbox"/> Create Target Directory			
File Name Scheme *	xi_output_abhay.xml		
FTP Connection Parameters			
Server *	10.75.65.148		
Port *	21		
Data Connection	Passive		
Timeout (secs)			
Connection Security *	None		
<input checked="" type="checkbox"/> Anonymous Login			
Connect Mode *	Permanently		
Transfer Mode *	Binary		

Note: Target Directory will depend on the system to which we are connecting. File Name Scheme will can also change as per requirement.

7. Receiver Determination

A receiver determination defines one or more receivers for a sender and a outbound Service interface.

Create a new receiver determination, by right the scenario and then select Receiver Determination under Configuration Objects –



Create Object

- Configuration
 - Configuration Wizard
- Collaboration Profile
 - Party
 - Business Component
 - Communication Channel
 - Process Component
- Collaboration Agreement
 - Sender Agreement
 - Receiver Agreement
 - Direct Connection
 - Integrated Configuration
- Configuration Objects
 - Receiver Determination**
 - Receiver Rule
 - Interface Determination
 - Value Mapping Group
- Administration
 - Configuration Scenario
 - Folder
 - Change List

Receiver Determination

Sender

Communication Party: _____

Communication Component *: BC_FileASystem

Interface *: SI_File_Sender_Out_Abhay

Namespace *: http://capgemini.com/PI710/FileA_Abhay

☐ Sender Uses Virtual Receiver

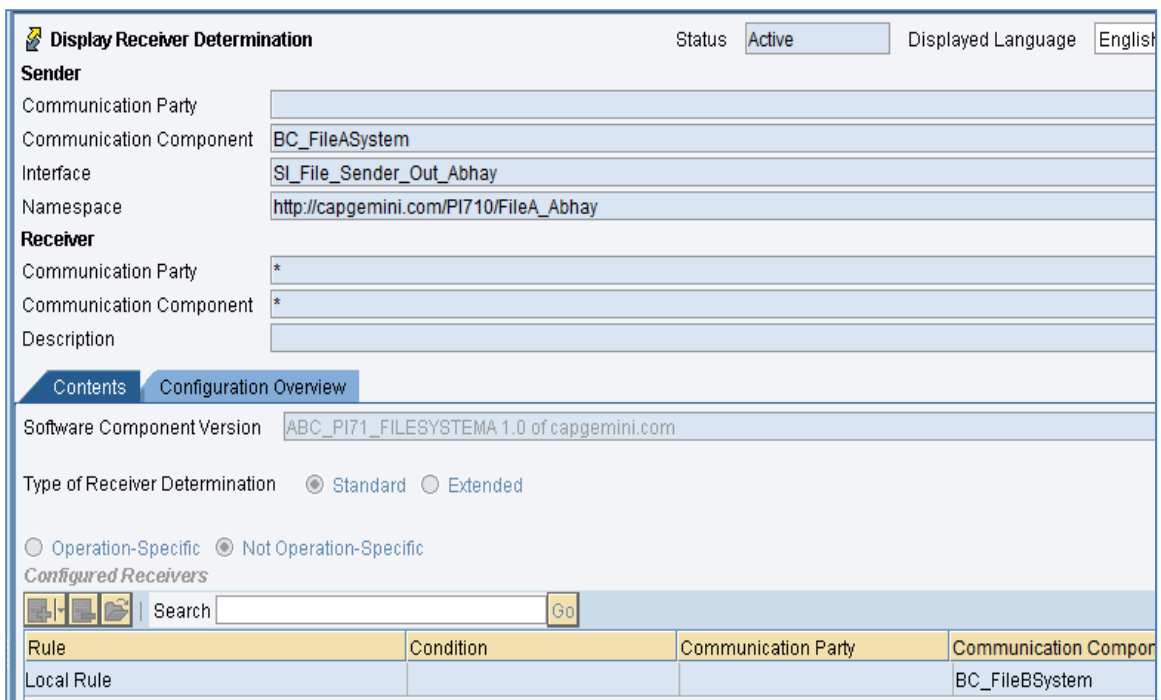
Description: _____

Add to Scenario: FILETOFILE_Abhay

Add to Folder: ABC_PI71_Training

Create **Cancel**

Now add receiver business component –



Display Receiver Determination Status: **Active** Displayed Language: **English**

Sender

Communication Party: _____

Communication Component: BC_FileASystem

Interface: SI_File_Sender_Out_Abhay

Namespace: http://capgemini.com/PI710/FileA_Abhay

Receiver

Communication Party: *

Communication Component: *

Description: _____


Contents **Configuration Overview**

Software Component Version: ABC_PI71_FILESYSTEMA 1.0 of capgemini.com

Type of Receiver Determination: ☒ Standard ☐ Extended

☐ Operation-Specific ☒ Not Operation-Specific

Configured Receivers

 Search: _____ **Go**

Rule	Condition	Communication Party	Communication Component
Local Rule			BC_FileBSystem

8. Interface Determination

In this step, we need to determine the inbound service interface using which the receiver system is expected to receive data.

Create a new interface determination, by right the scenario and then select Interface Determination under Configuration Objects –

Create Object

- ▼ **Configuration**
 - Configuration Wizard
- ▼ **Collaboration Profile**
 - Party
 - Business Component
 - Communication Channel
 - Process Component
- ▼ **Collaboration Agreement**
 - Sender Agreement
 - Receiver Agreement
 - Direct Connection
 - Integrated Configuration
- ▼ **Configuration Objects**
 - Receiver Determination
 - Receiver Rule
 - Interface Determination**
 - Value Mapping Group
- ▼ **Administration**
 - Configuration Scenario
 - Folder
 - Change List

Interface Determination

Sender

Communication Party: _____

Communication Component *: BC_FileASystem

Interface *: SI_File_Sender_Out_Abhay

Namespace *: http://capgemini.com/PI710/FileA_Abhay

Receiver

Communication Party: _____

Communication Component *: BC_FileBSystem

Description: _____

Add to Scenario: FILETOFILE_Abhay

Add to Folder: ABC_PI71_Training

Create **Cancel**

Now add the inbound interface and operation mapping to be used by this scenario -

Configuration: Integration Builder (inmumsap24_P12_00)

Object Navigation Tools Environment Help

Objects Change Lists

- kalpvora
- _Poshah
- _Shweta
- AAE_MUMOHAMM
- ABC_PI71_Training
 - BC_FileASystem
 - BC_FileBSystem
 - * | BC_FileASystem | CC_File_Sender_Abhay
 - * | BC_FileBSystem | CC_File_Receiver_Abhay
 - BC_FileASystem | SI_File_Sender_Out_Abhay
 - BC_FileASystem | SI_File_Sender_Out_Abhay
 - FILETOFILE_Abhay
- Abid
- AbidAssignment
- AbidDemo
- AG_PL_TRAINING
- algarg
- Alok
- Amith
- Anjiti_Demo
- Arpi_T

Edit Interface Determination Status: In Process Displayed Language: English (OL)

Sender

Communication Party: _____

Communication Component: BC_FileASystem

Interface: SI_File_Sender_Out_Abhay

Namespace: http://capgemini.com/PI710/FileA_Abhay

Receiver

Communication Party: _____

Communication Component: BC_FileBSystem

Description: _____

Software Component Version of Sender Interface: ABC_PI71_FILESYSTEMA 1.0 of capgemini.com

☒ Maintain Order at Runtime

Operation

SI_File_Sender_Out_Abhay

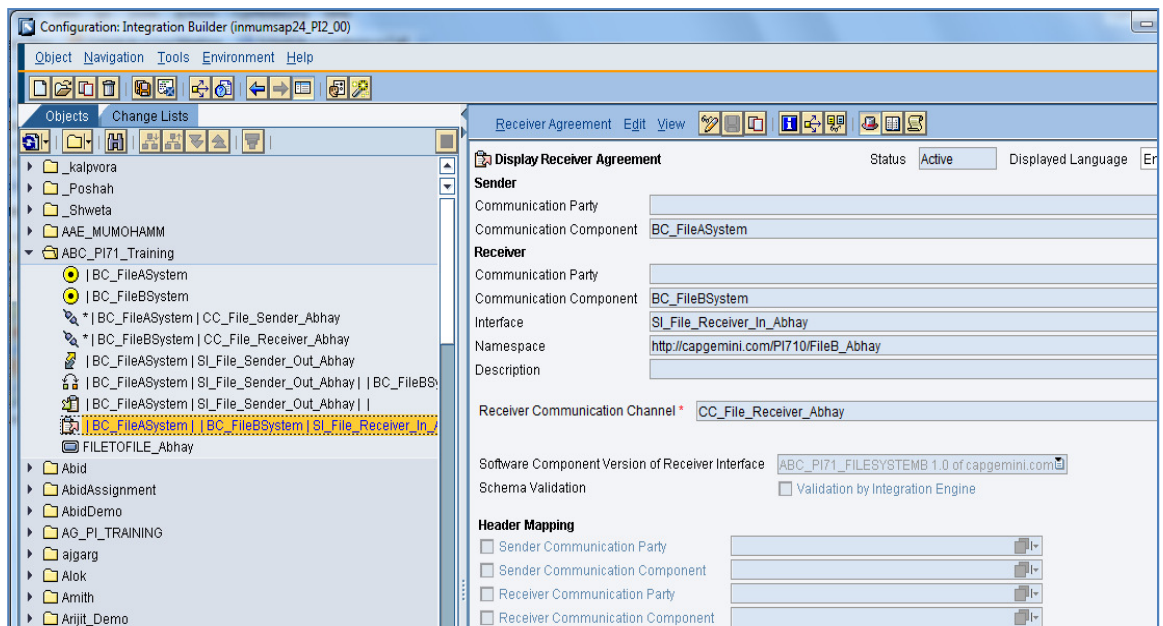
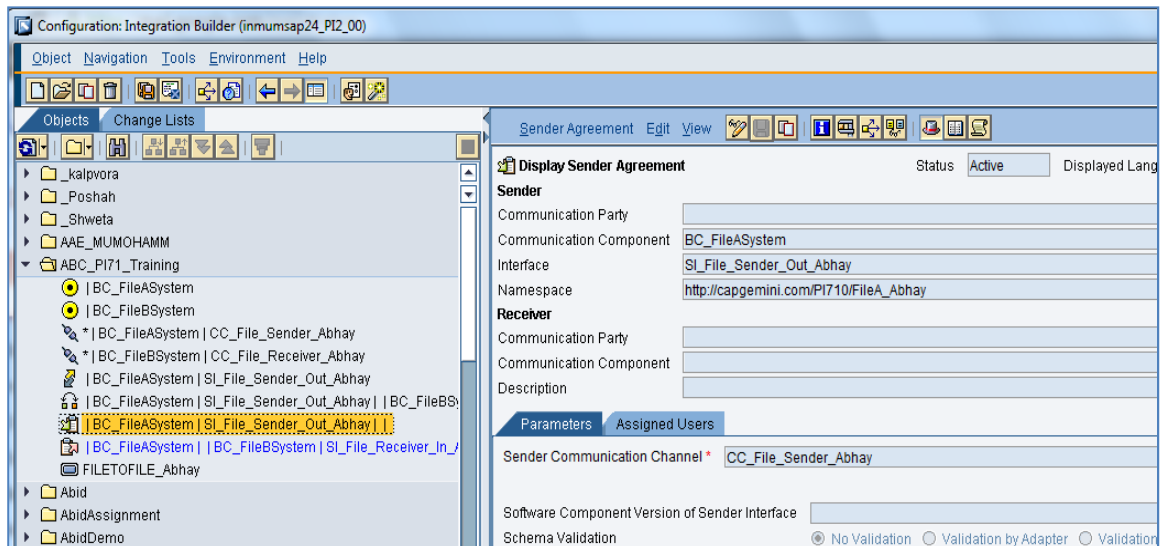
Receiver Interfaces *

Condition	Operation Mapp.	Name *	Namespace *	Multiplicity
OM_FileA_Fil	SI_File_Rece		http://capgem	1

9. Sender and Receiver Agreement

The agreements we need to specify the communication channel to be used by sending/receiving systems to actually connect to Integration Engine.

Create the sender and receiver receiver agreement by specifying sender and receiver business component and adding corresponding communication channel.



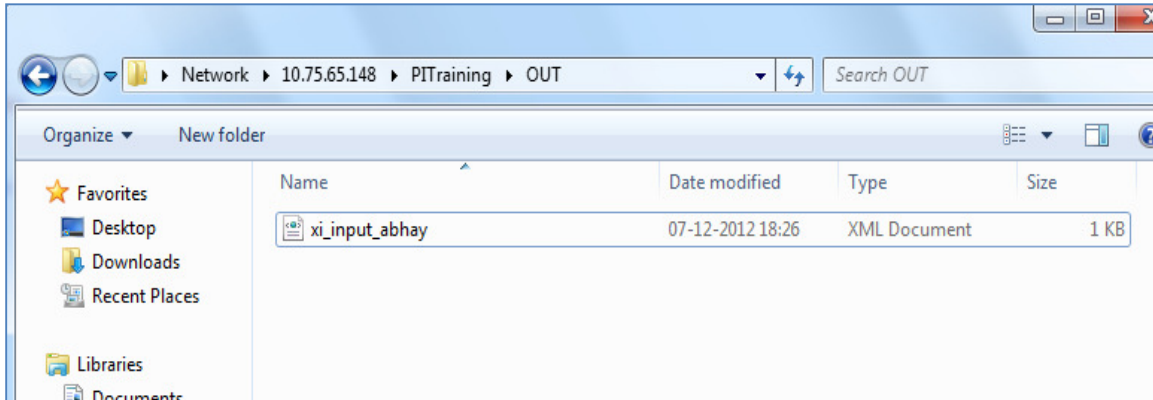
Testing:

Place the file in the source directory.

Use the following path in the run command

\\ 10.75.65.148\PITraining\OUT\

And paste your file in it as shown below



Once the file is placed. It is picked up by PI and processed and placed in the mentioned target directory. So we can check in our target directory \\ 10.75.65.148\PITraining\IN\ Xi_output_abhay xml file would be created.

