A Step-by-Step Guide on IDoc-to-File Using Business Service in the XI Integration Directory

Applies to:

SAP Exchange Infrastructure (XI) 3.0 / Process Integration (PI) 7.0

This document is intended for all XI aspirants who want to read and understand some of the SAP R/3 transactions for creating and posting IDoc to XI. This document explains step-by-step working procedures for the IDoc-to-File scenario in XI including Integration Repository (IR), Integration Directory (ID), and other adapter-specific settings for IDoc sender side. This scenario also explains the Business Service in ID and when/where to use Business Service instead of Business System. Hence, it applies to **Beginners** and **Intermediate** XI developers, especially for those who don't have any ABAP/IDoc knowledge and have limited or no experience using Business Service in ID.

Finally this document might be useful for those interested in preparing themselves for **XI Certification** or attending for some **XI Interviews**.

Summary

This document gives complete step-by-step procedures for implementing and testing an IDoc-to-File scenario from scratch using the Business Service in the Integration Directory (ID). The document also provides some basic idea and working knowledge while posting and testing IDoc processing by creating RFC Destination (SM59), Port (WE21), Logical System (BD54), Partner Profile (WE20), using IDoc Testing Tool (WE19), and Check for IDoc processing status using transactions WE02, WE05, SM58. So, this document can be treated as a ready reference in real life situations when working with IDoc and Exchange Infrastructure in a cross-company environment (to use Business Service).

Author: Satish Jaiswal

Company: HCL Technologies, Kolkata, India

Created on: 19 June 2007

Author Bio

Satish Jaiswal is working as a SAP NetWeaver Consultant for HCL Technologies, Kolkata.

Table of Contents

Introduction

One can find lot of articles and documents on IDoc - XI - FILE scenario in a distributed fashion in SDN but none of these gives a clear IDEA and complete step by step procedures to be followed while implementing such scenario. Also most of the IX scenario implementation & example follows Business System in ID not Business Service. So, it becomes very difficult for any beginner and Intermediate XI aspirants to implement such a scenario from scratch without having any knowledge on IDoc and R/3 system in Cross Company Environment, where all relevant information for creating a Business System in SLD for Sender or Receiver is not is not available. Hence, I have tried to come up with a complete solution and How to guide document to provide all steps required implementing such a scenario from the very beginning. Hope this document will help all XI professionals as a ready reference in real life situation while working with IDoc and XI scenario in cross company environment. Happy Learning…

Basic Concepts Overview (XI Implementation)

As per general thumb rule to implement any XI scenario (like File - XI - File) we need to implement the following:

- In SLD: Create Product and Software Components with/without dependencies, Technical System and Business System for each (Sender and Receiver)

Tip: If we don't have sufficient information to create a Business System in SLD for Sender or Receiver then we can use Business Service in ID as shown in this example (Cross Company Environment).

- In IR: Import Software Components that we have created in SLD, Create Namespaces then under Interface Object we have to create Data type, Message Type, Message Interface for each (Sender and Receiver). After this we have to create Massage Mapping and Interface Mapping for the scenario under Mapping Objects. Finally we have to activate all these objects we have created in IR.
- In ID: Create Configuration Scenario under which we have to import / assign Business Systems / Business Services, Communication Channels for (Sender and Receiver), then we have to create Receiver Determination, Interface Determination, Sender Agreement and Receiver Agreement. Again we have to activate all these objects in ID.

Requirement Study and Scope Identification

We need to implement one scenario where IDoc will be configured and posted from R/3 System to XI Sever. The XI sever need to transform the IDoc Information into target data format (XML in our example). It should store the xml file into the target directory using File Adapter. Assume that we don't have sufficient information to create a Business System in SLD for Sender R/3 System. Hence, I have used Business Service in ID like in any Cross Company Environment.

Identifying Solution

In our IDoc – XI – FILE scenario first we have to identify our requirements and TODO list for implementing the scenario.

In R/3 Sender System (For posting IDoc to XI – Outbound IDoc)

- ✓ Identify Message type and Basic type to be used for posting the IDoc
- ✓ Create RFC Destination (SM59) to Receiver XI system
- ✓ Create PORT for IDoc processing (WE21) and assign the RFC Destination created above to this PORT
- ✓ Create Logical System (LS) for the IDoc (BD54) to be sent to the XI receiver system
- ✓ Create Partner Profile with outbound parameter (WE20) to be send to the XI receiver system

In XI System for Integration (Receive IDoc - Transform - Send File)

In SLD (System Landscape Directory)

- ✓ One Business system as 3rd party option should be available with corresponding Technical System, Product, Software components for Receiver System

In IR (Integration Repository)

- ✓ Import Software components that we have created in SLD
- ✓ Create Namespaces
- ✓ Under Interface Objects Create Data type, Message Type, Message Interface for Receiver

Tip: we don't need any Data Type, Message Type and Message Interface for Sender because we will be sending IDoc from Sender R/3 System so, we will Import the IDoc from Sender system and this will be used for Source Data Type, Message Type and Message Interface.

- ✓ Under Mapping Objects create Massage Mapping and Interface Mapping for the scenario
- ✓ Activate all objects created in IR

In ID (Integration Directory)

- ✓ Create a new Configuration Scenario Object
- ✓ Create New Business Services, and set adapter specific Identifier for the Sender IDoc (Logical System Name, R/3 Sender System ID and Client)

Tip: This is Important for the receiver system to receive the IDoc from the Sender System.

- ✓ Configure Business service for Receiver (Inbound Interface and Communication Channel)
- ✓ Configure Business service for Sender (Outbound Interface)

Tip: Here for Sender System we don't need any Communication Channel because IDoc sits on ABAP Stack.

- ✓ Create Receiver Determination
- ✓ Create Interface Determination
- ✓ Create Receiver Agreement
- ✓ Activate all objects created in ID

Tip: Here for Sender System we don't need any Sender Agreement because IDoc sits on ABAP Stack.

Some Other Configuration (Capturing Metadata Information etc) in XI System

- ✓ Create RFC Destination (SM59) to Sender System
- ✓ Create PORT using transaction (IDX1) to get the Metadata of the IDoc from the Sender System (The Port Name = The Port Name in the source IDoc header – normally in format SAP<System ID>. eg. SAPXID in our example [though this is Optional]
- ✓ To maintain the IDoc Metadata we need to use transaction (IDX2). This is only for XI System as XI needs to construct Idoc-XML from the Idoc. In our example initially we can avoid this step

Pre-Requisites

Knowledge Requirements:

- Basic knowledge of XI and how it works

- Nice to have one file to file scenario implementation experience. Please see the reference section if you want to go through one file to file scenario example

System Requirements:

- One Business System in SLD (System Landscape directory) is required which should be created using Third Party Option
- Access to Transactions SM59, WE21, BD54, WE20, WE19, WE02, WE05, and SM58 is required for **Sender R/3 System.** And SM59, IDX1, IDX2 for XI System.

Tip: If you don't have access right to any of this transaction, you can contact your basis /XI administrator so that he can help you to create / configure objects for you including SLD.

- Developer Access to **Recipient XI System** is required to create this scenario

Implementation - Build Solution

Maintain R/3 Sender System (For posting IDoc to XI – Outbound IDoc)

Step 1: Identify Message type and Basic Type to be used for posting the IDoc

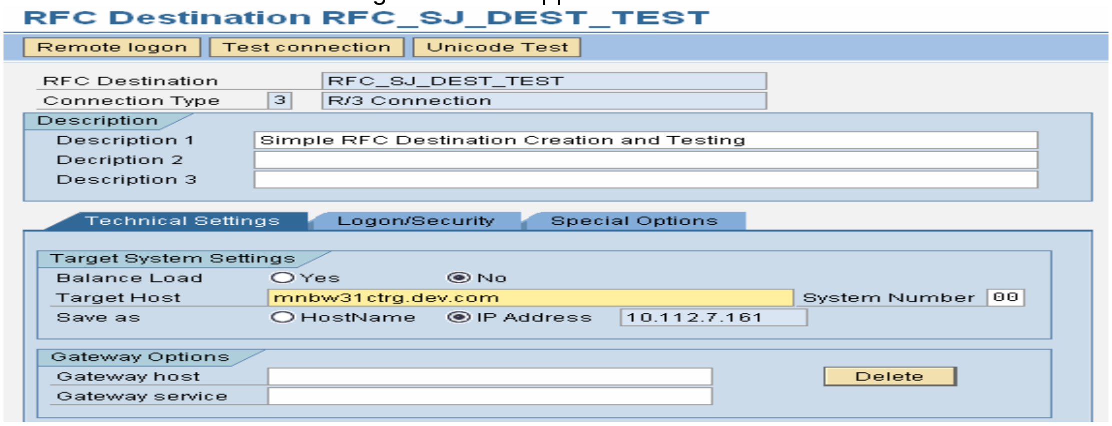
- This scenario is for understanding all step required to implement IDoc to FILE scenario. For simplicity we will use **Message Type** FLCUSTOMER_CREATEFROMDATA and **Basic Type** FLCUSTOMER_CREATEFROMDATA01 (for IDoc). Because this message type provides less mandatory fields and will be easier for mapping.

Step 2: Logon to Sender R3 System

Step 3: Create RFC Destination to Receiver XI System

- Open Transaction SM59
- Click on Create Button
  - Give RFC Destination Name(RFC_SJ_DEST_TEST in example)
  - Connection type select **3**

  Tip: Connection type 3 is required as we are going to send IDoc from R/3 to XI Server which is again R/3 with ABAP Stack.
- Click on Save. The following screen will appear


RFC Destination RFC_SJ_DEST_TEST

RFC Destination RFC_SJ_DEST_TEST
Connection Type 3 R/3 Connection

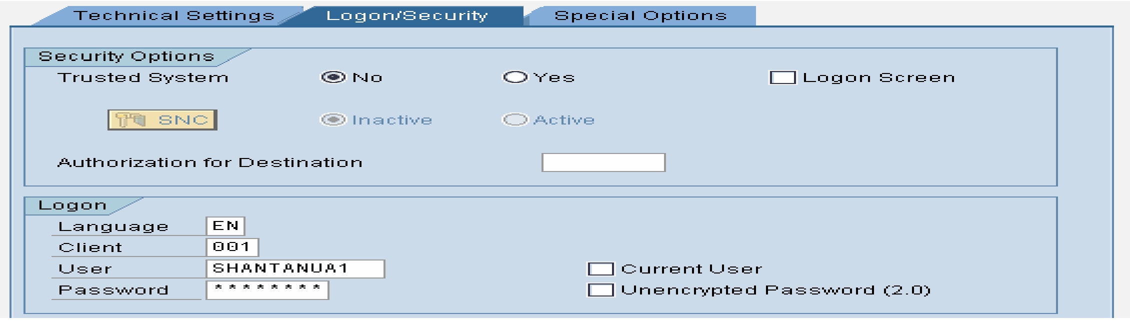
Description
Description 1 Simple RFC Destination Creation and Testing
Decription 2
Description 3

Target System Settings
Balance Load ○ Yes ● No
Target Host mnbw31ctrg.dev.com System Number 00
Save as ○ HostName ● IP Address 10.112.7.161

Gateway Options
Gateway host
Gateway service

- Give Target Host Name and System Number
- Now click on Logon/Security Tab
  - In Security Options Area Leave default values
  - In Logon Area Give Language, Client, User and Password for your Destination System eg. Receiver System in our case XI Server user.

  Tip: You can check the current user check box if the current Sender System User is already available in the receiver system.



Security Options

Trusted System ● No ○ Yes □ Logon Screen

SNC ● Inactive ○ Active

Authorization for Destination

Logon

Language	EN
Client	001
User	SHANTANUA1
Password	********

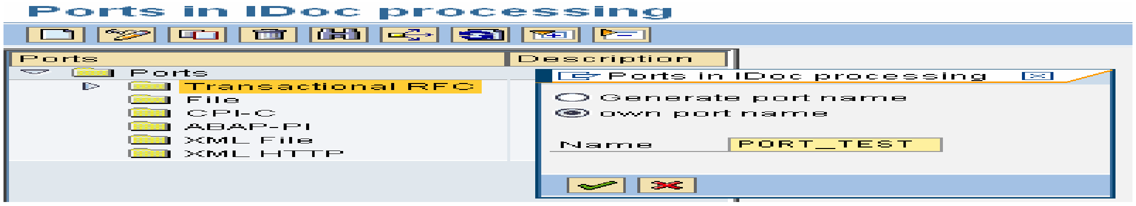
□ Current User

□ Unencrypted Password (2.0)

- Click on Save
- Now the RFC Destination is ready for testing
- Click on Test Connection Button and Remote Logon Button. Both must be successful

Step 4: Create Communication PORT for IDoc Processing

- Open Transaction WE21
- Select the Transactional RFC from Ports (left tree panel)
- Click on Create Button
- The following screen will appear



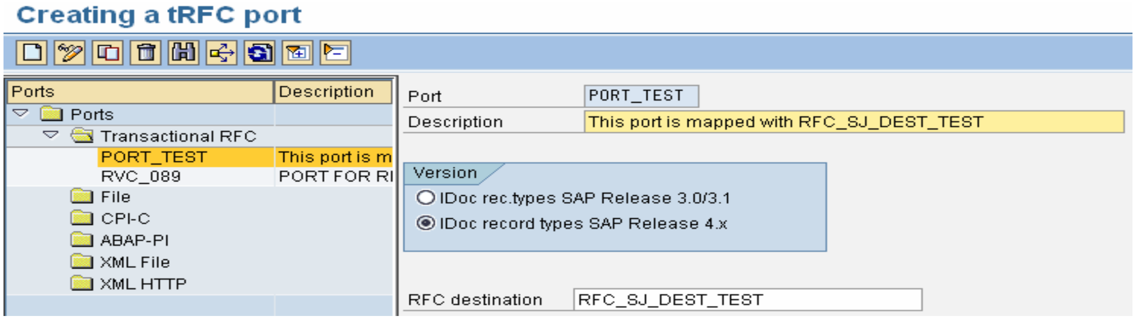
Ports in IDoc processing

○ Generate port name

● own port name

Name PORT_TEST

- Give Port Name here (PORT_TEST in this example) and click on continue. The next screen will appear



Ports	Description
PORT_TEST	This port is m
RVC_089	PORT FOR RI

Port PORT_TEST

Description This port is mapped with RFC_SJ_DEST_TEST

Version

○ IDoc rec.types SAP Release 3.0/3.1

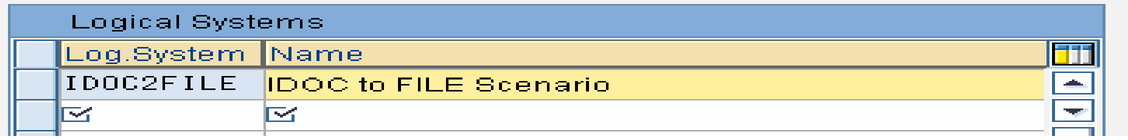
● IDoc record types SAP Release 4.x

RFC destination RFC_SJ_DEST_TEST

- Give the description of the port and select the RFC destination (in this example RFC_SJ_DEST_TEST which we have created in step3). Now click on save.

Step 5: Create Logical System (LS) for IDoc Receiver System

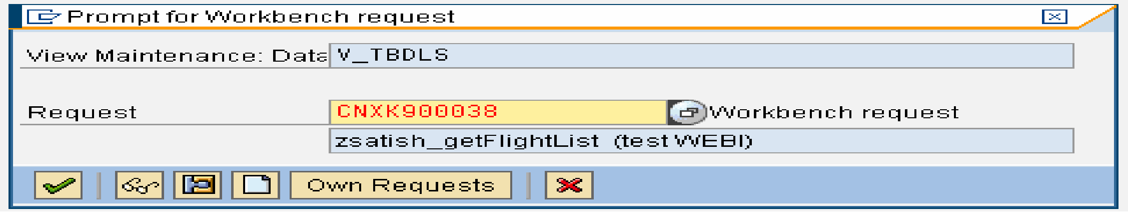
- Open Transaction BD54
- Click on New Entries Button
- Give data for your Logical System and Name (IDoc2FILE in this example)



Logical Systems

Log.System	Name
IDOC2FILE	IDOC to FILE Scenario

- Now click on save. Here one window may appear (if so) just click on continue. Now the Logical System (LS) is ready

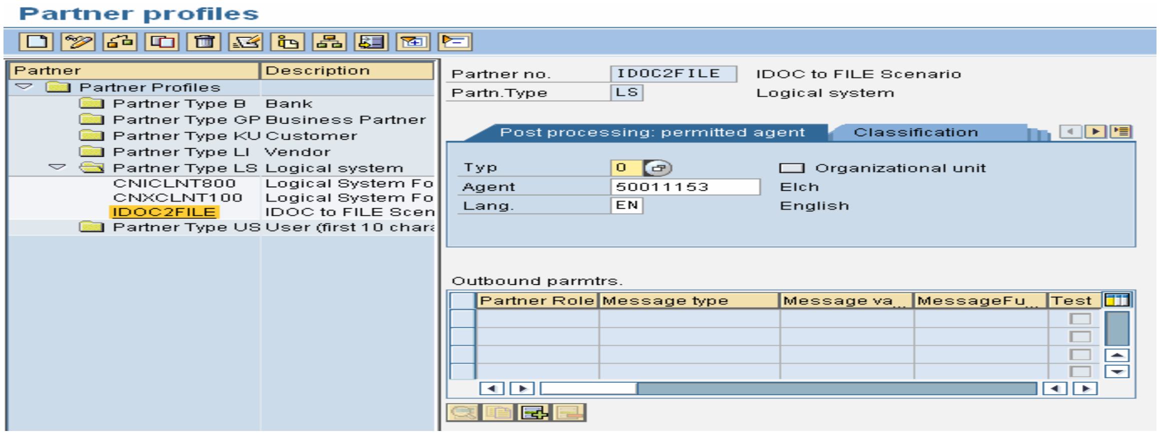


Prompt for Workbench request

View Maintenance: Data	V_TBDLS
Request	CNXK900038 Workbench request
	zsatish_getFlightList (test WEBI)

Step 6: Create Partner Profile with Outbound Parameter(s)

- Open Transaction WE20
- Select Partner Type LS from Partner (left tree panel)
- Click on Create Button
- Give Logical System Name which we have just created in step 5 as Partner No. (In this example IDoc2FILE), Partner Type should be LS (Just Check), Agent (some valid data) Language (EN). Now click on Save



Partner profiles

Partner	Description
Partner Profiles	
Partner Type B	Bank
Partner Type GP	Business Partner
Partner Type KU	Customer
Partner Type LI	Vendor
Partner Type LS	Logical system
CNICLNT800	Logical System Fo
CNXCLNT100	Logical System Fo
IDOC2FILE	IDOC to FILE Scen
Partner Type US	User (first 10 chara

Partner no.	IDOC2FILE	IDOC to FILE Scenario
Partn.Type	LS	Logical system

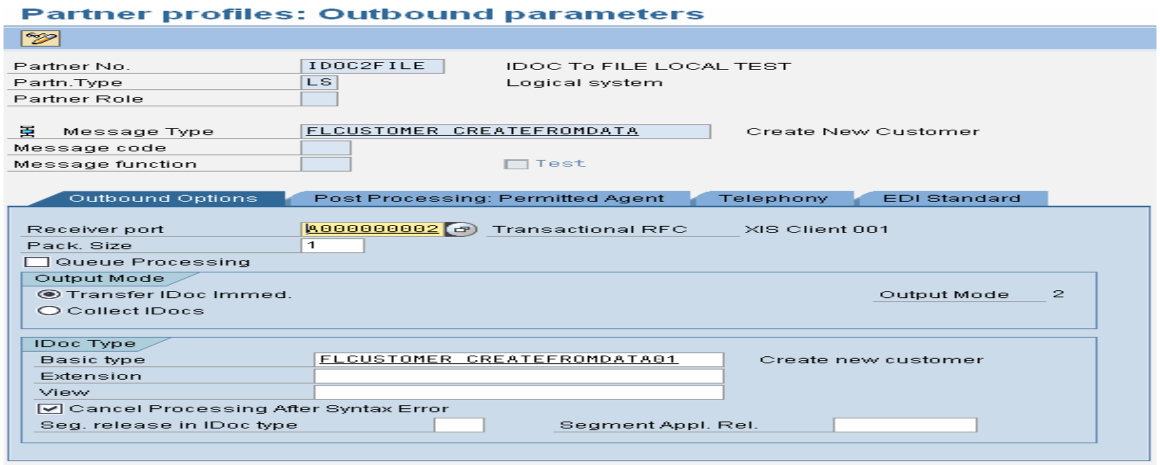
Post processing: permitted agent | Classification

Typ	O	□ Organizational unit
Agent	50011153	Elch
Lang.	EN	English

Outbound parmtrs.

Partner Role	Message type	Message va...	MessageFu...	Test
				□
				□
				□
				□

- Now we have to give some Outbound Parameter, so click on Add to create outbound parameter, the following screen will appear.



Partner profiles: Outbound parameters

Partner No.	IDOC2FILE	IDOC To FILE LOCAL TEST
Partn.Type	LS	Logical system
Partner Role		
Message Type	FLCUSTOMER_CREATEFROMDATA	Create New Customer
Message code		
Message function		□ Test

Outbound Options | Post Processing: Permitted Agent | Telephony | EDI Standard

Receiver port	A000000002	Transactional RFC	XIS Client 001
Pack. Size	1		

□ Queue Processing

Output Mode

● Transfer IDoc Immed. Output Mode 2

○ Collect IDocs

IDoc Type

Basic type	FLCUSTOMER_CREATEFROMDATA01	Create new customer
Extension		
View		

☑ Cancel Processing After Syntax Error

Seg. release in IDoc type Segment Appl. Rel.

- Select the Message Type (In this example FLCUSTOMER_CREATEFROMDATA), then Select the Receiver Port (which we have created in step 4 - PORT_TEST in this example). Please select Transfer IDoc Immediately option in Output mode for Immediate Testing. Select Basic Type: FLCUSTOMER_CREATEFROMDATA01. Save. Press F3 to go back to previous screen, here you will see Message Type FLCUSTOMER_CREATEFROMDATA is added in the outbound parameter.

Maintain XI System for Integration (Receive IDoc - Transform - Send File)

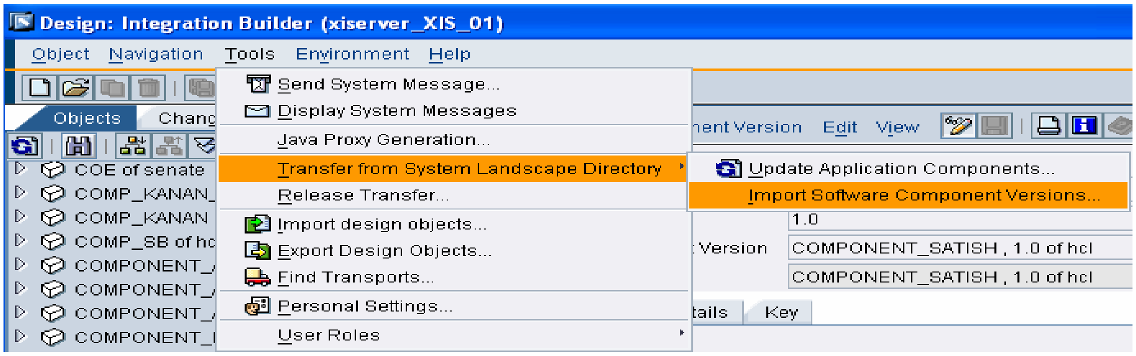
Step 1: Maintain SLD (System Landscape Directory)

One Business system as 3rd party with corresponding Technical System, Product, Software components for Receiver System needs to be created. You can ask your Basis / XI Administrator to create the same for you. You can refer to the article written by Srinivas Vanamala for File to File Scenario and SLD Creation.

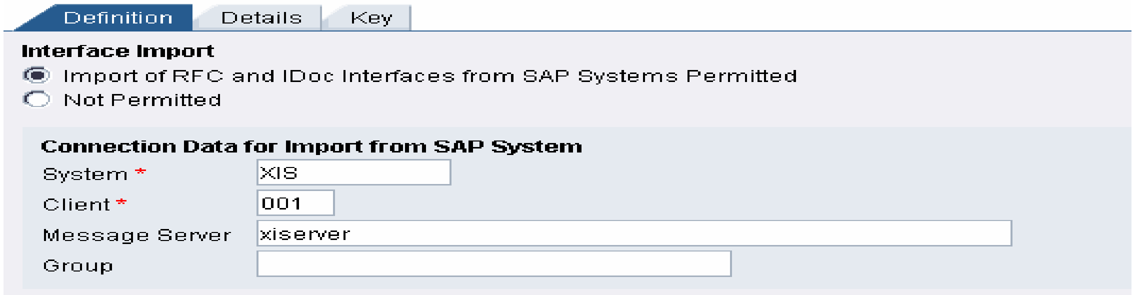
Step 2: Maintain IR (Integration Repository)

✓ Import S/W Component and IDoc structure

- ➢ Open Integration Repository
- ➢ Import you software components information which has been created in SLD to IR.



- ➢ Now select Import of RFC and IDoc interfaces from Sap System permitted. Provide other relevant info like Sender System ID, Client and Message Server (usually the Sender System Host) click on Save.



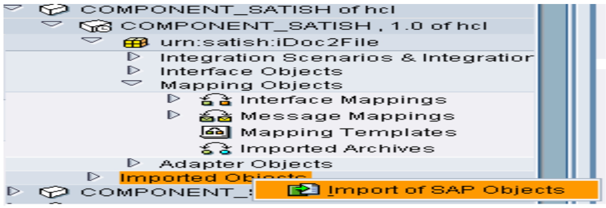
Interface Import

● Import of RFC and IDoc Interfaces from SAP Systems Permitted
○ Not Permitted

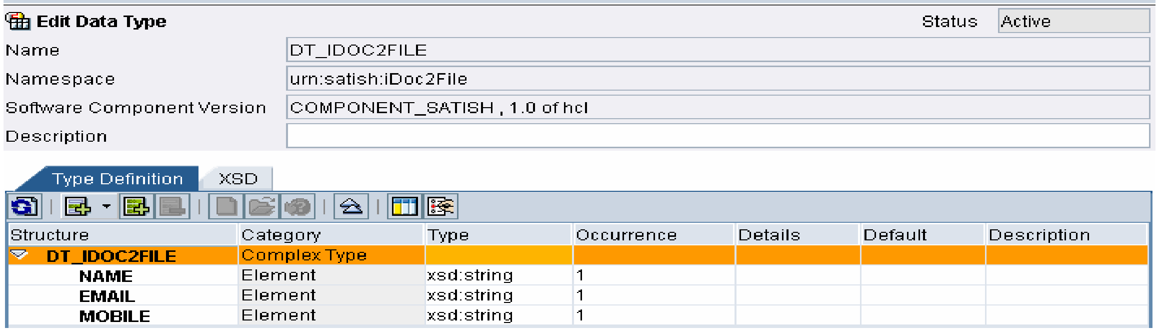
Connection Data for Import from SAP System

System *	XIS
Client *	001
Message Server	xiserver
Group	

- ➢ Now right click on Imported Object under your Software Component and click on Import of SAP Objects. Give the relevant Sender System information and select the IDoc to import the structure and finish.



✓ Create Data Type (DT_IDoc2FILE) for Receiver: (For simplicity in Mapping and Structure, we will use only 3 fields which will be transferred from the source IDoc to the destination file system, you can also use more fields while implementing this scenario)

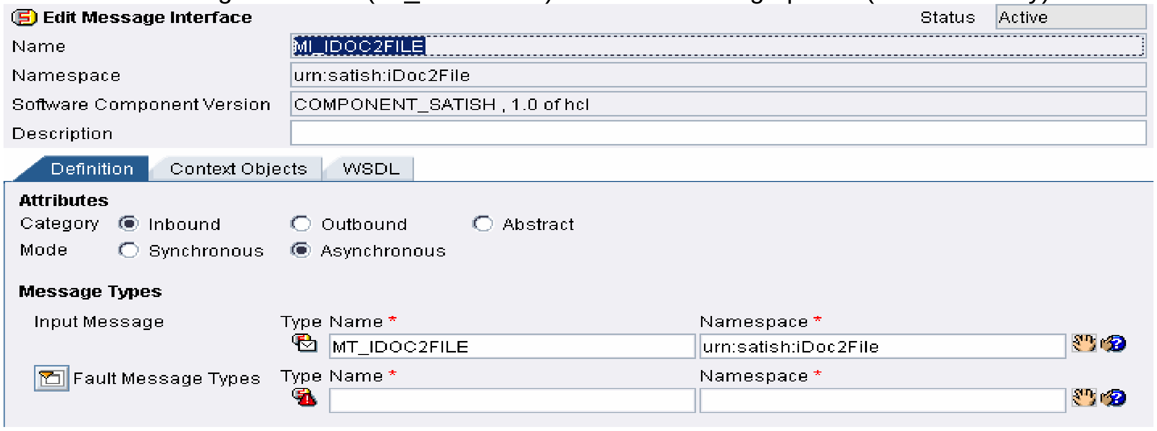


Edit Data Type

Status	Active
Name	DT_IDOC2FILE
Namespace	urn:satish:iDoc2File
Software Component Version	COMPONENT_SATISH , 1.0 of hcl
Description	

Structure	Category	Type	Occurrence	Details	Default	Description
DT_IDOC2FILE	Complex Type					
NAME	Element	xsd:string	1			
EMAIL	Element	xsd:string	1			
MOBILE	Element	xsd:string	1			

- ✓ Likewise create message type(MT_IDoc2FILE) for the above created data type (Receiver Only)
- ✓ Create Massage Interface (MI_IDoc2FILE) with the following options (Receiver Only)



Edit Message Interface

Status	Active
Name	MI_IDOC2FILE
Namespace	urn:satish:iDoc2File
Software Component Version	COMPONENT_SATISH , 1.0 of hcl
Description	

Attributes

Category ● Inbound ○ Outbound ○ Abstract

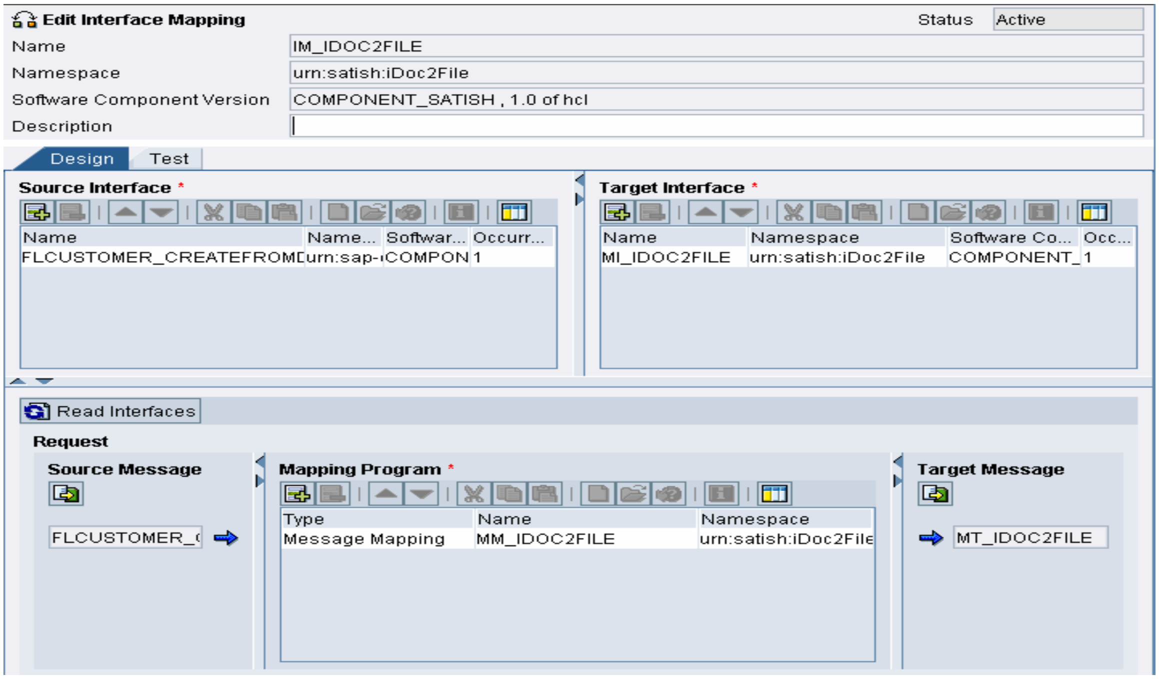
Mode ○ Synchronous ● Asynchronous

Message Types

	Type Name *	Namespace *
Input Message	MT_IDOC2FILE	urn:satish:iDoc2File
Fault Message Types		

- ✓ Create Message Mapping(Transformation between Sender and Receiver) as given below:
  - Select IDoc (FLCUSTOMER_CREATEFROMDATA01) as source message type and MT_IDoc2FILE as Target Message type and do mappings as follows and save:
    I. Customer Name -> Name
    II. Phone ->Mobile
    III. Email -> Email.

- ✓ Create Interface Mapping (IM_IDoc2FILE) as below:



Edit Interface Mapping — Status: Active

Name: IM_IDOC2FILE

Namespace: urn:satish:iDoc2File

Software Component Version: COMPONENT_SATISH , 1.0 of hcl

Description:

Name	Name...	Softwar...	Occurr...
FLCUSTOMER_CREATEFROMI	urn:sap-i	COMPON	1

Name	Namespace	Software Co...	Occ...
MI_IDOC2FILE	urn:satish:iDoc2File	COMPONENT_	1

Source Message: FLCUSTOMER_(

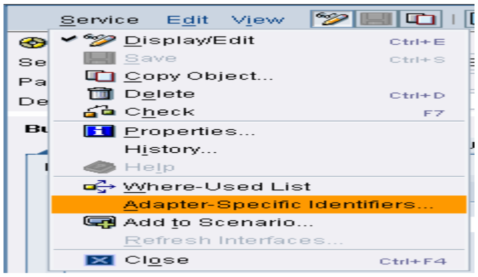
Type	Name	Namespace
Message Mapping	MM_IDOC2FILE	urn:satish:iDoc2File

Target Message: MT_IDOC2FILE

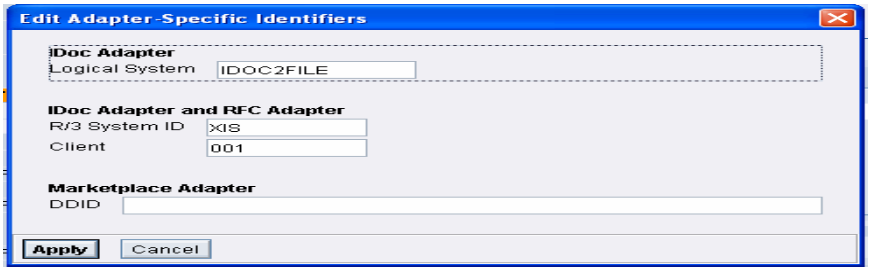
- ✓ Now GOTO Change List and activate

Step 3: Maintain ID (Integration Directory)

- ✓ Open Integration Directory
- ✓ Create One new Configuration Scenario(CS_IDoc2FILE in this example)
- ✓ Now select the newly created Configuration Scenario - > Service Without Party-> Business service-> Right Click and create New. Give one Business Service Name(BS_SJ_IDoc2FILE in example)
- ✓ Now click On Menu Service-> Adapter Specific Identifiers



- ✓ Please Select the Logical System (LS) that we have created in the Sender System(R/3) for the IDoc (IDoc2FILE in this example). Also give the Sender R3 System ID and Client number and click on Apply. Please note this step is very important so check for data consistency twice. Here Logical System (LS) and the LS of Sender System(R/3) for the IDoc should be Identical



Edit Adapter-Specific Identifiers

IDoc Adapter

Logical System: IDOC2FILE

IDoc Adapter and RFC Adapter

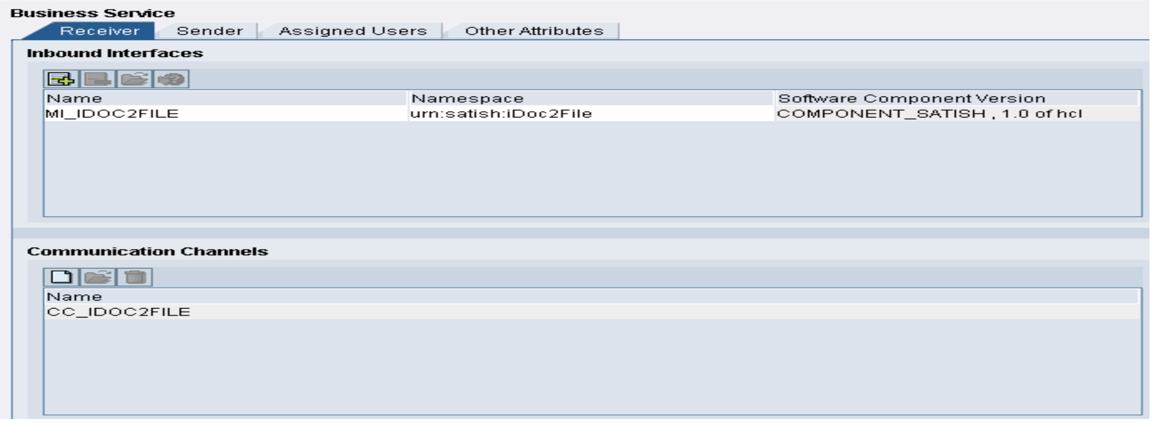
R/3 System ID: XIS

Client: 001

Marketplace Adapter

DDID:

- ✓ Now in Business Service window select the following (Inbound Interface and Communication Channel-File Adapter) for Receiver. Put all valid entry as shown below



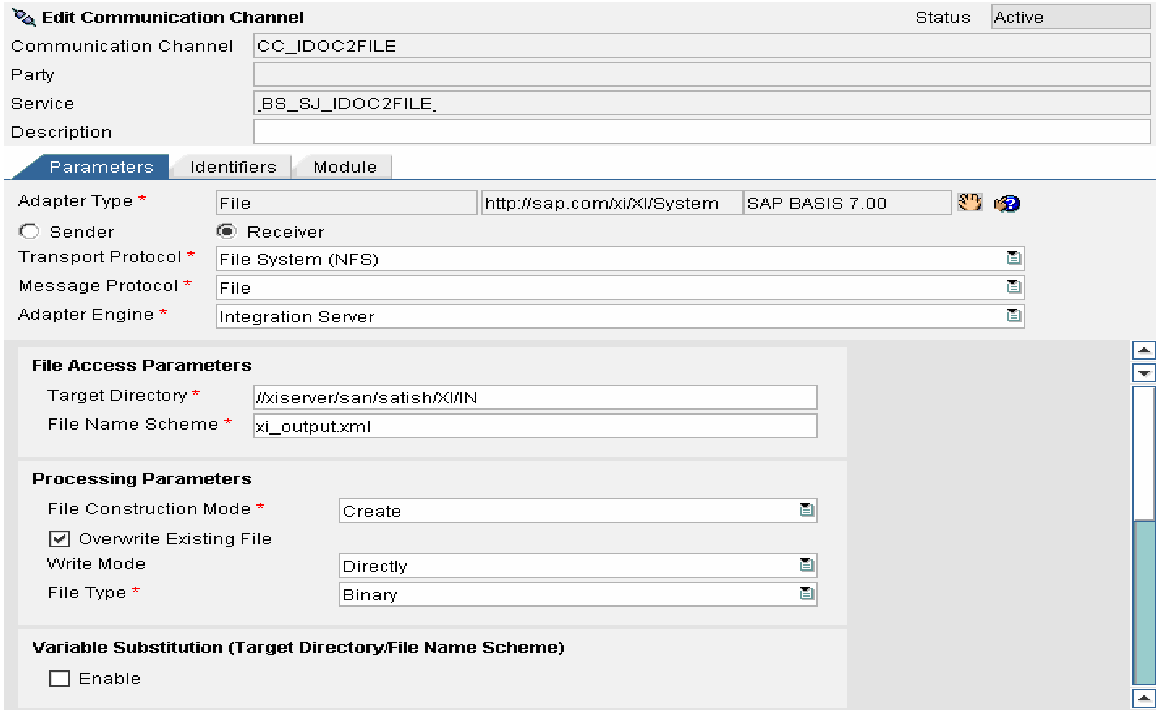
Inbound Interfaces

Name	Namespace	Software Component Version
MI_IDOC2FILE	urn:satish:iDoc2File	COMPONENT_SATISH , 1.0 of hcl

Communication Channels

Name
CC_IDOC2FILE

- ✓ Please select Communication channel details as given below. Please remember to change the Target Directory and File Name Scheme as per your available directory and required file name.



Status	Active
Communication Channel	CC_IDOC2FILE
Party	
Service	BS_SJ_IDOC2FILE
Description	

Adapter Type * File | http://sap.com/xi/XI/System | SAP BASIS 7.00

○ Sender ● Receiver

Transport Protocol *	File System (NFS)
Message Protocol *	File
Adapter Engine *	Integration Server

File Access Parameters

Target Directory *	//xiserver/san/satish/XI/IN
File Name Scheme *	xi_output.xml

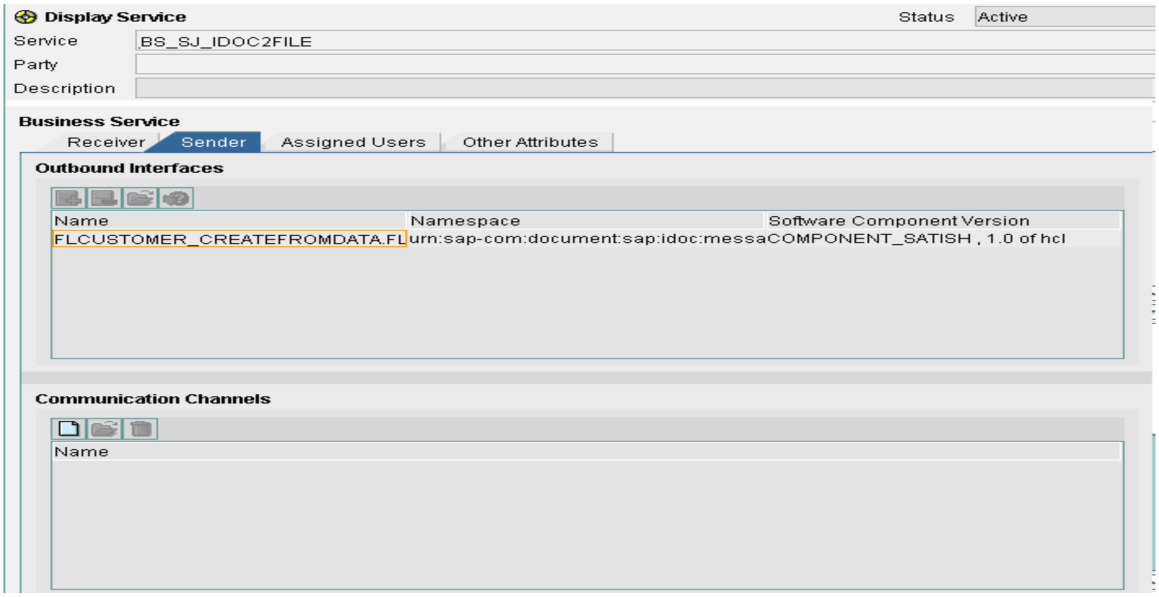
Processing Parameters

File Construction Mode *	Create
☑ Overwrite Existing File	
Write Mode	Directly
File Type *	Binary

Variable Substitution (Target Directory/File Name Scheme)

☐ Enable

- ✓ Goto Sender Tab and select the following. Please note we don't need any Sender Channel as IDoc sits on ABAP Stack and for Sender side IDoc we don't need any Sender Communication channel and Sender Agreement.



Display Service — Status: Active

Service	BS_SJ_IDOC2FILE
Party	
Description	

Business Service

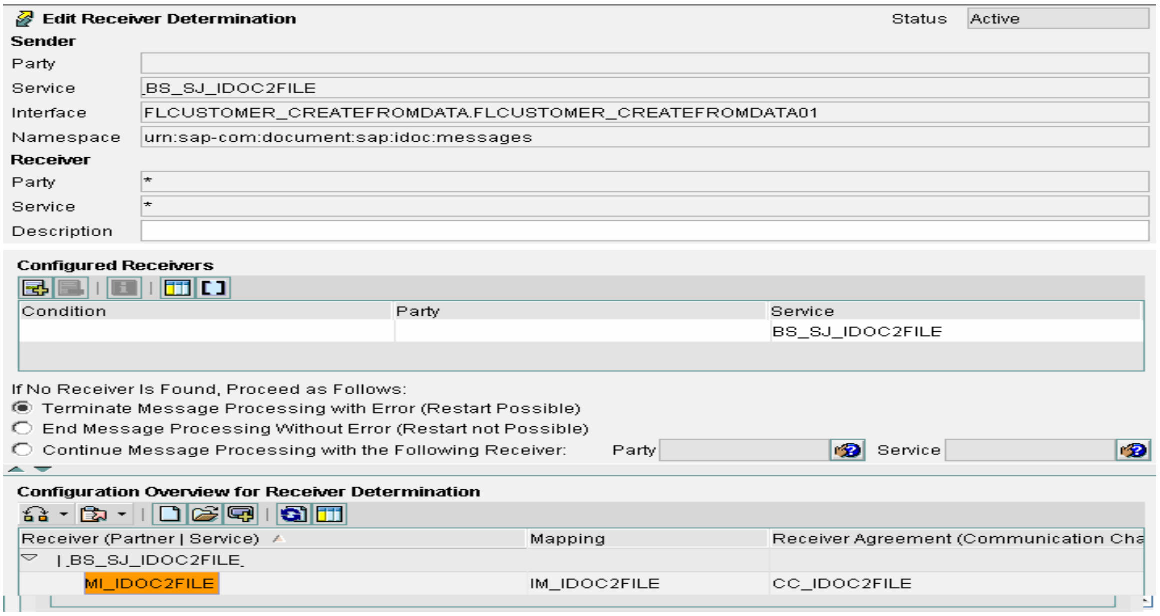
Outbound Interfaces

Name	Namespace	Software Component Version
FLCUSTOMER_CREATEFROMDATA.FL	urn:sap-com:document:sap:idoc:messa	COMPONENT_SATISH , 1.0 of hcl

Communication Channels

Name

✓ Now create Receiver Determination as shown below:



Edit Receiver Determination — Status: Active

Sender

Party	
Service	BS_SJ_IDOC2FILE
Interface	FLCUSTOMER_CREATEFROMDATA.FLCUSTOMER_CREATEFROMDATA01
Namespace	urn:sap-com:document:sap:idoc:messages

Receiver

Party	*
Service	*
Description	

Configured Receivers

Condition	Party	Service
		BS_SJ_IDOC2FILE

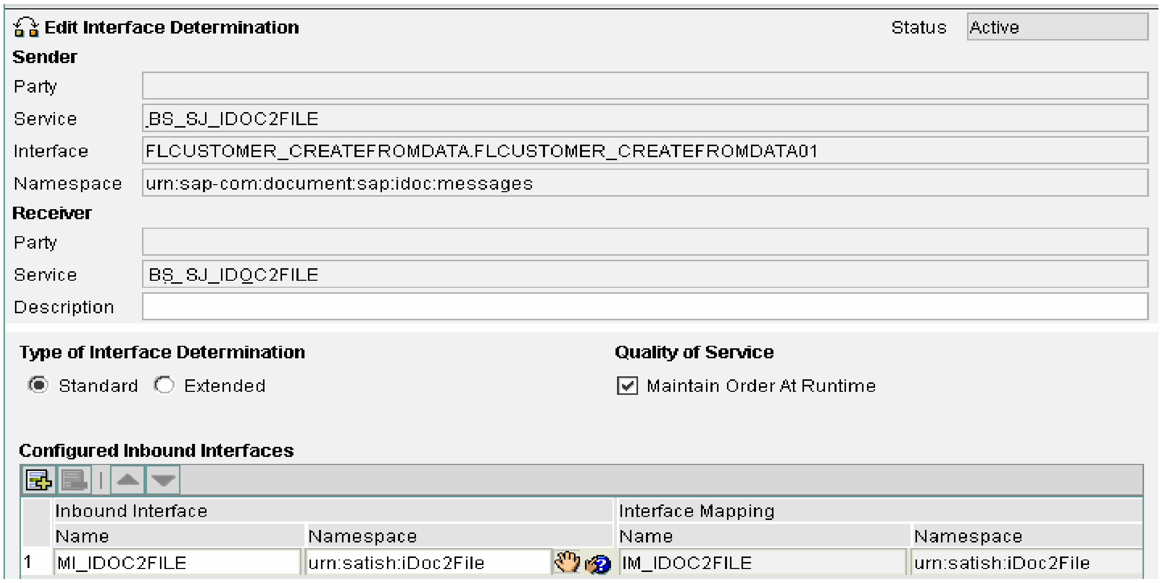
If No Receiver Is Found, Proceed as Follows:

● Terminate Message Processing with Error (Restart Possible)
○ End Message Processing Without Error (Restart not Possible)
○ Continue Message Processing with the Following Receiver: Party Service

Configuration Overview for Receiver Determination

Receiver (Partner | Service)	Mapping	Receiver Agreement (Communication Cha
| BS_SJ_IDOC2FILE		
MI_IDOC2FILE	IM_IDOC2FILE	CC_IDOC2FILE

✓ Now create Interface Determination as shown below:



Edit Interface Determination — Status: Active

Sender

Party	
Service	BS_SJ_IDOC2FILE
Interface	FLCUSTOMER_CREATEFROMDATA.FLCUSTOMER_CREATEFROMDATA01
Namespace	urn:sap-com:document:sap:idoc:messages

Receiver

Party	
Service	BS_SJ_IDOC2FILE
Description	

Type of Interface Determination

● Standard ○ Extended

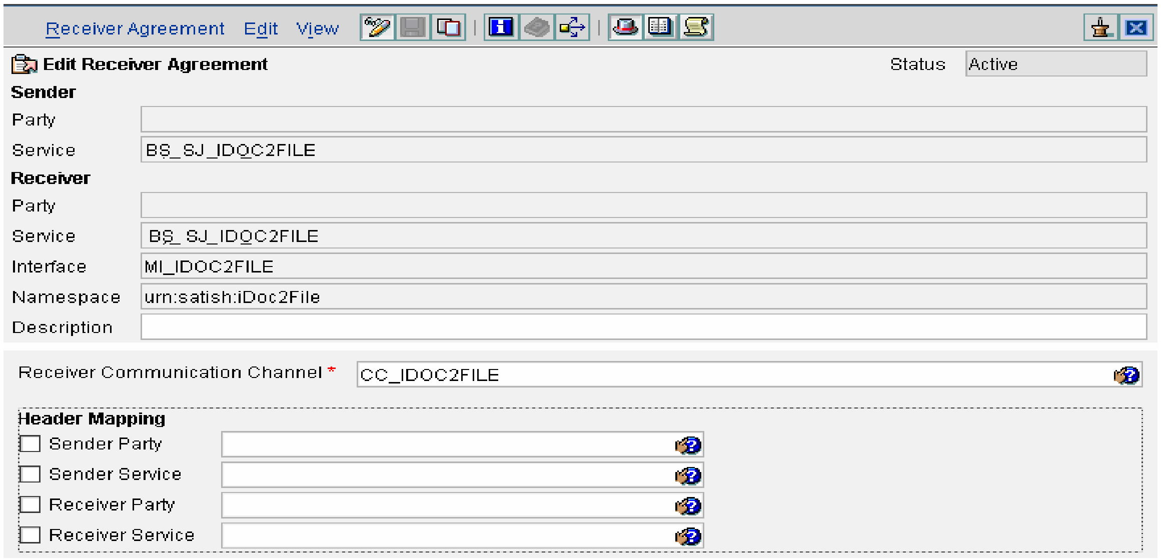
Quality of Service

☑ Maintain Order At Runtime

Configured Inbound Interfaces

	Inbound Interface		Interface Mapping	
	Name	Namespace	Name	Namespace
1	MI_IDOC2FILE	urn:satish:iDoc2File	IM_IDOC2FILE	urn:satish:iDoc2File

- ✓ Now we have to create the Receiver Agreement as shown in figure below. Please select the Business Service (BS_SJ_IDoc2FILE) that we have already created.



Edit Receiver Agreement — Status: Active

Sender

Party	
Service	BS_SJ_IDOC2FILE

Receiver

Party	
Service	BS_SJ_IDOC2FILE
Interface	MI_IDOC2FILE
Namespace	urn:satish:iDoc2File
Description	

Receiver Communication Channel * CC_IDOC2FILE

Header Mapping

□ Sender Party
□ Sender Service
□ Receiver Party
□ Receiver Service

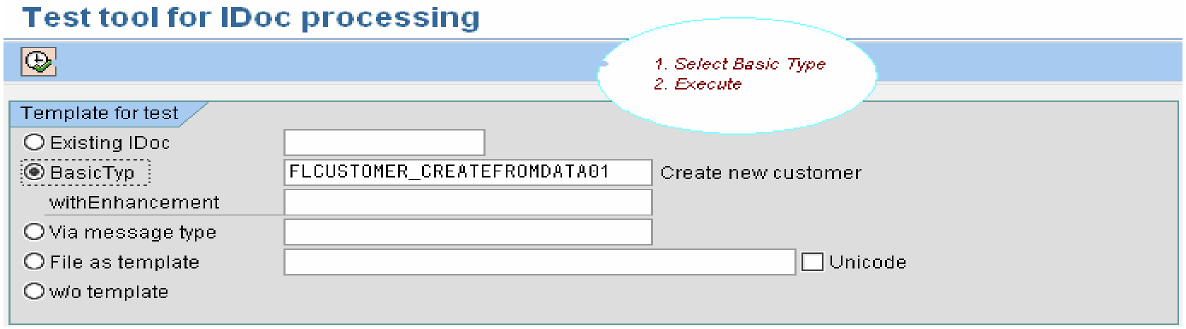
- ✓ All the objects created should be activated without any error.

Step 3: Some Other Configuration in XI System

- ✓ Create RFC Destination (SM59) to Sender System (R/3). This can also be used for acknowledgement. This is similar Step as mentioned in Step 3 (Create RFC Destination to Receiver XI System). We have to specify this RFC Destination while maintaining the Metadata Information of the Sender System in XI using transaction IDX1.
- ✓ Create PORT using transaction (IDX1) to get the Metadata of the IDoc from the Sender System (The Port Name = The Port Name in the source IDoc header – normally in format SAP<System ID>.   eg. SAPXID in our example [though this is Optional]. This is Similar to transaction WE21. This should be in sync with data maintained in R/3 else the IDoc is will not transmitted to the recipient system successfully
- ✓ To maintain the IDoc Metadata we need to use transaction (IDX2). This is only for XI System as XI needs to construct Idoc-XML from the Idoc. In our example initially we can avoid this step.

IDoc Posting and Testing the Scenario

- ✓ Log in to sender system, go to transaction WE19 (test tool to generate IDoc), generate IDoc of the required message type.



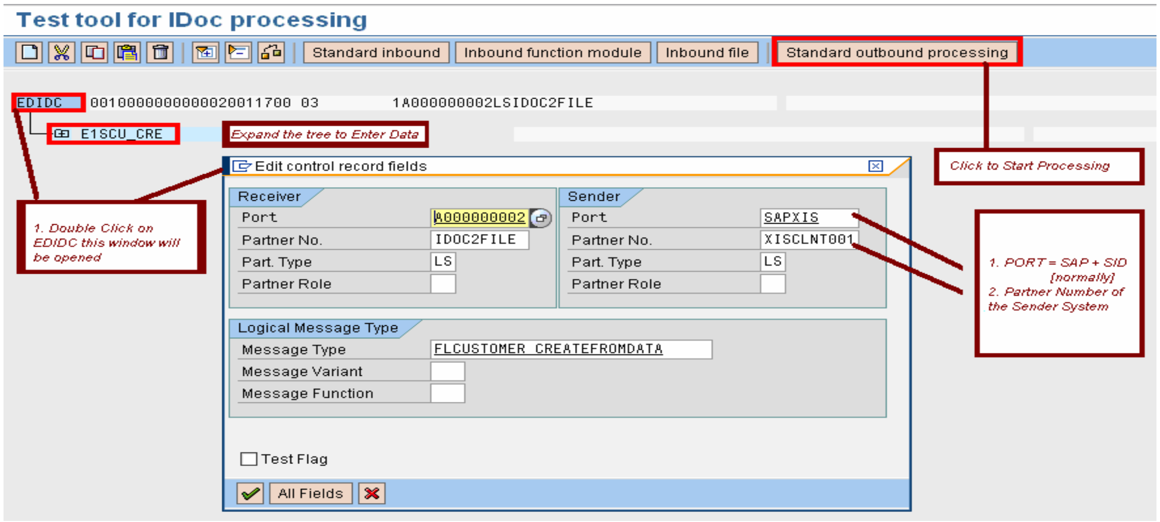
Test tool for IDoc processing

1. Select Basic Type
2. Execute

Template for test

○ Existing IDoc		
● BasicTyp	FLCUSTOMER_CREATEFROMDATA01	Create new customer
withEnhancement		
○ Via message type		
○ File as template		□ Unicode
○ w/o template		

- ✓ In the following screen follow the steps as shown in figure below: Give Receiver and Sender System Info(Port, Partner No, Port Type and Message Type) and click on Continue



Test tool for IDoc processing

EDIDC 00100000000000020011700 03 1A000000002LSIDOC2FILE

E1SCU_CRE

Expand the tree to Enter Data

Click to Start Processing

1. Double Click on EDIDC this window will be opened

1. PORT = SAP + SID [normally]
2. Partner Number of the Sender System

Edit control record fields

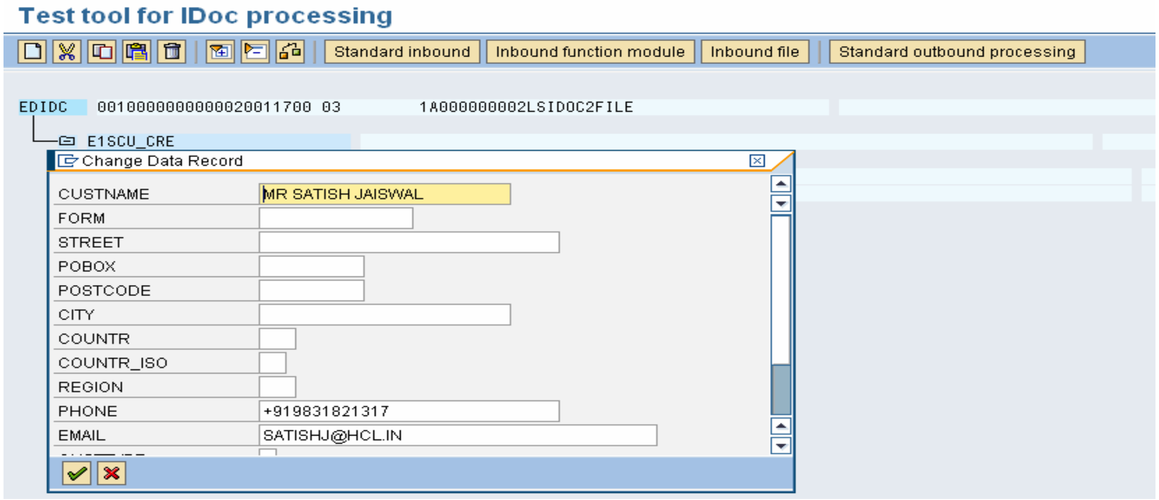
Receiver		Sender	
Port	A000000002	Port	SAPXIS
Partner No.	IDOC2FILE	Partner No.	XISCLNT001
Part. Type	LS	Part. Type	LS
Partner Role		Partner Role	

Logical Message Type

Message Type	FLCUSTOMER_CREATEFROMDATA
Message Variant	
Message Function	

□ Test Flag

- ✓ Now put some entry into the IDoc before you send



Test tool for IDoc processing

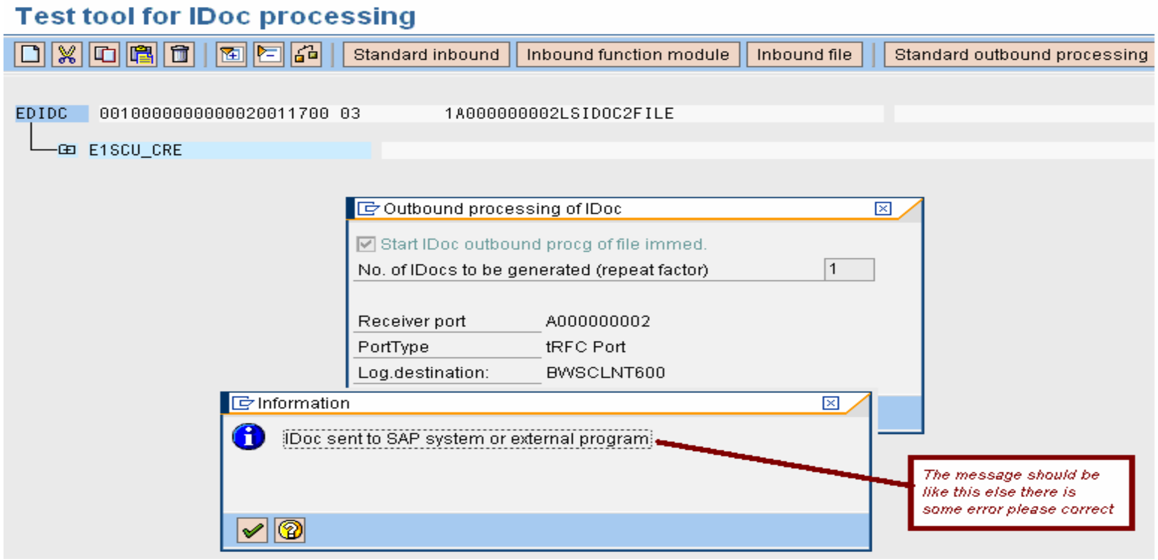
EDIDC 00100000000000020011700 03 1A000000002LSIDOC2FILE

E1SCU_CRE

Change Data Record

CUSTNAME	MR SATISH JAISWAL
FORM	
STREET	
POBOX	
POSTCODE	
CITY	
COUNTR	
COUNTR_ISO	
REGION	
PHONE	+919831821317
EMAIL	SATISHJ@HCL.IN

- ✓ Now click on Standard outbound processing button the following screen will appear: Click on continue again continue. In the last window Idoc sent to Sap system or external program information should appear else you may have to check Receiver and Sender System Info(Port, Partner No, Port Type and Message Type) for valid entry.



EDIDC 0010000000000020011700 03 1A000000002LSIDOC2FILE

E1SCU_CRE

Outbound processing of IDoc

☑ Start IDoc outbound procg of file immed.

No. of IDocs to be generated (repeat factor) 1

Receiver port A000000002

PortType tRFC Port

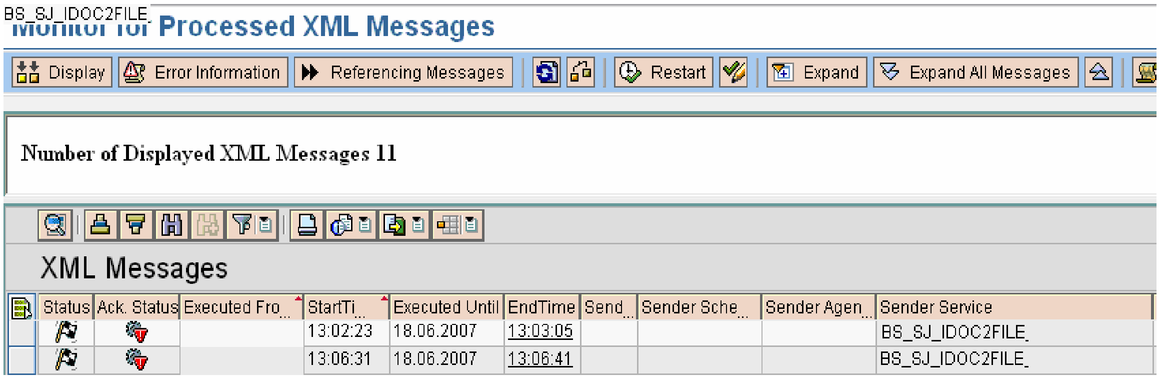
Log.destination: BWSCLNT600

Information

IDoc sent to SAP system or external program

The message should be like this else there is some error please correct

- ✓ Execute Transaction WE05; enter correct date/time frame and logical message type. Hopefully the Idoc Status should be 3 (successful), if not check the status code and correct the error and reprocess the message. You can use transaction code SM58 with enter correct date/time frame and execute. Here if it is successful then nothing should be available in the list else it will show the relevant Status / Message. WE02 transaction can be used to get the error info.
- ✓ Log in to Xi system and execute transaction SXMB_MONI and check for the XML messages. Use your source and destination system to filter messages; it should have flag as in picture below.

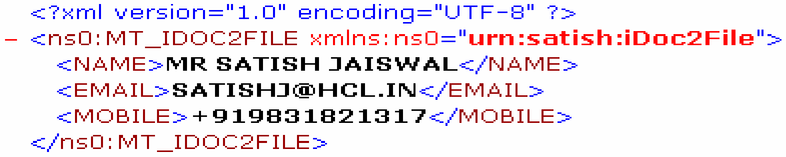


Number of Displayed XML Messages 11

XML Messages

Status	Ack. Status	Executed Fro...	StartTi...	Executed Until	EndTime	Send...	Sender Sche...	Sender Agen...	Sender Service
⚑	●		13:02:23	18.06.2007	13:03:05				BS_SJ_IDOC2FILE_
⚑	●		13:06:31	18.06.2007	13:06:41				BS_SJ_IDOC2FILE_

- ✓ Check the destination directory, it should have file with the name specified. Check the file; it should have data for all segments like below:



```
  <?xml version="1.0" encoding="UTF-8" ?>
- <ns0:MT_IDOC2FILE xmlns:ns0="urn:satish:iDoc2File">
    <NAME>MR SATISH JAISWAL</NAME>
    <EMAIL>SATISHJ@HCL.IN</EMAIL>
    <MOBILE>+919831821317</MOBILE>
  </ns0:MT_IDOC2FILE>
```

Some Error Tracking Tips:

Check whether IDoc was created:

1. Execute Transaction WE05; enter correct date/time parameters etc. If IDoc found in failed status, open status records folder and double click on the failed status records to get error message.

Check Whether IDoc is received by the Receiver:

2. Execute Transaction WE05; enter correct date/time parameters etc and verify if the IDoc was created or not and if IDoc is in a successful status or not. If the Idoc is in Status-30(Yellow), you can push it again through Transaction BD87.
3. If the IDoc is located and status is successful but receiver has not received the IDoc, Execute transaction **SM58** to determine if IDoc is stuck in the ALE/RFC layer. If entries exist in SM58, look for Message Type
4. If message type exist and Run the Program "RSARFCEX". If after the RSARFCEX program has ran and IDocs are not in SM58, issue should be solved. Validate in receiving system via WE05 that the IDoc was received or not.
5. If IDoc is still stuck in SM58, check RFC connection. Go to SE37, enter function module RFC_PING. We need to have access to SM59 to test the RFC Connection.
6. After an issue has been resolved and/or you need to process an IDoc, besides processing IDocs via programs, you can also process IDocs via Transaction BD87.

Understanding Outcomes (Tips)

- ✓ **Tip 1:** One XI Message contains exactly one IDoc as payload
- ✓ **Tip 2:** The XI sender& receiver from main header are converted into sender & receiver of Idoc control record
- ✓ **Tip 3:** If we don’t have sufficient information to create a Business System in SLD for Sender or Receiver then we can use Business Service in ID as shown in this example (Cross Company Environment)
- ✓ **Tip 4:** we don’t need to create any Data Type, Message Type and Message Interface for Sender while IDoc is in Sender Side, because we will be sending IDoc from Sender R/3 System so, we will Import the IDoc from Sender system and this will be used for Source Data Type, Message Type and Message Interface
- ✓ **Tip 5:** In Business Service (ID) set adapter specific Identifier for the Sender IDoc (Logical System Name, R/3 Sender System ID and Client). This is Important for the receiver system to receive the IDoc from the Sender System and should be Identical with the Sender System
- ✓ **Tip 6:** For Sender System we don’t need any Sender Agreement and Communication Channel because IDoc sits on ABAP Stack

Related Content

- Info on Business Services (SAP Help Site)
- File To File scenario with SLD creation by Srinivas Vanamala
- Information on ALE / IDoc

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.