

## Conversion of IDOCs to XML format at SAP Level

This article details about the step by step in conversion of IDOCs to XML format for further use in XI or any other application. It is assumed that the reader of this article has some knowledge in ALE, IDOCs and Change Pointers.

**Scenario:** Conversion of the Material IDOC (Message type: MATMAS) to XML format and store the same in the application server of SAP.

### Approach

Change pointers are used for sending IDOCs for master data like Material Master. To work with Change pointers, following two steps have to be performed:

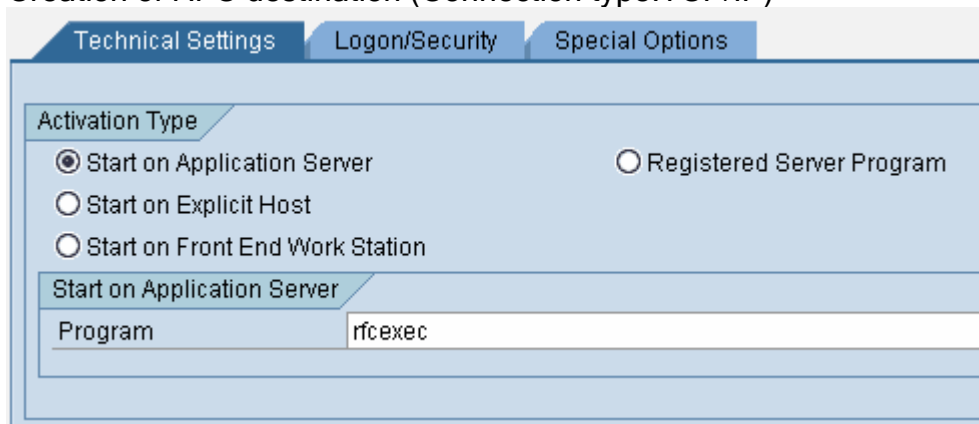
- Turn on change pointer update generally
- Providing the message types to be included for change pointer update.

To do the above configurations:

TCode: SALE → IDOC Interface / Application Link Enabling (SALE) → Modeling and implementing Business Processes → Master Data Distribution → Replication of Master Data

### *ALE Configuration Steps:*

1. Creation of logical system for the sender system.
2. Assignment of logical system to the client.
3. Create a logical system for the recipient
4. Creation of RFC destination (Connection type:TCP/IP)

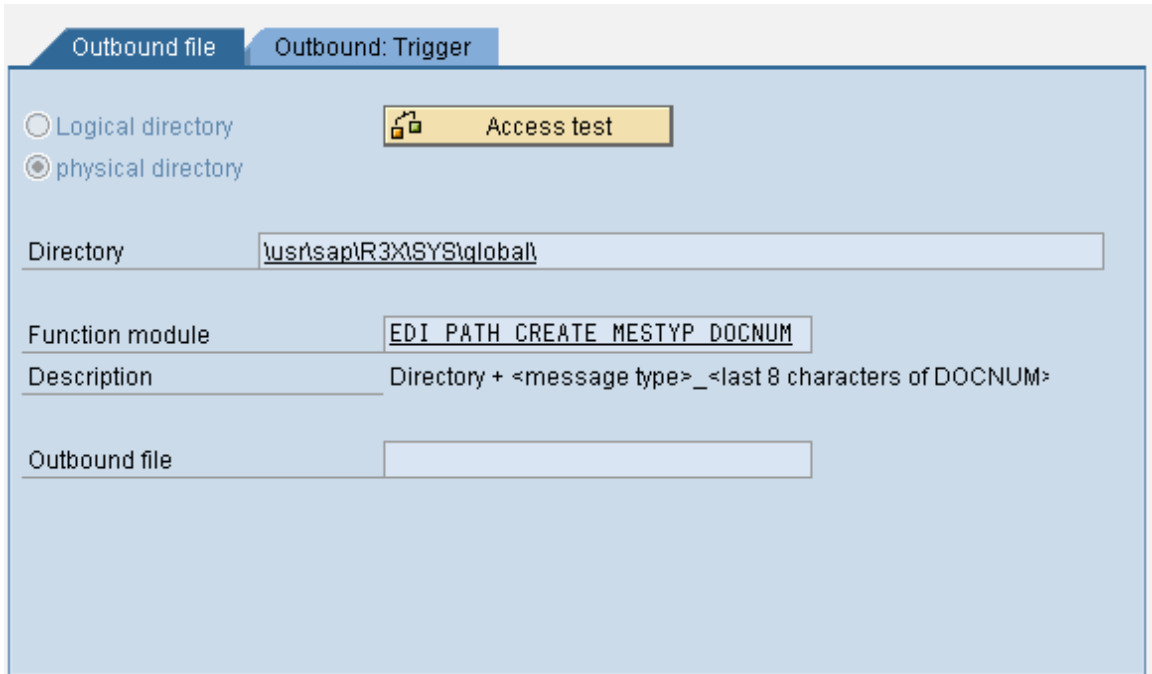


5. Creation of Model View (TCode: BD64).

Customizing Data Synchronization	CONTRLDATA
▼ Demo for IDoc -> XML conversion	IDOCTOXML
▼ Local System Client 100	R3XCLNT100
▼ Dummy logical system for XML placement	LOGSYSXML
▶ MATMAS	Material master
▶ Example of MM contract distribution (filtering at hea	MM-PUR1

6. Save the Model View and Generate Partner Profiles.
7. There might be a problem with the automatic Port creation. Creation of the port has to be done manually.
8. Create an XML Port from the transaction WE21 (Port type: XML File).

## Conversion of IDOCs to XML format at SAP Level

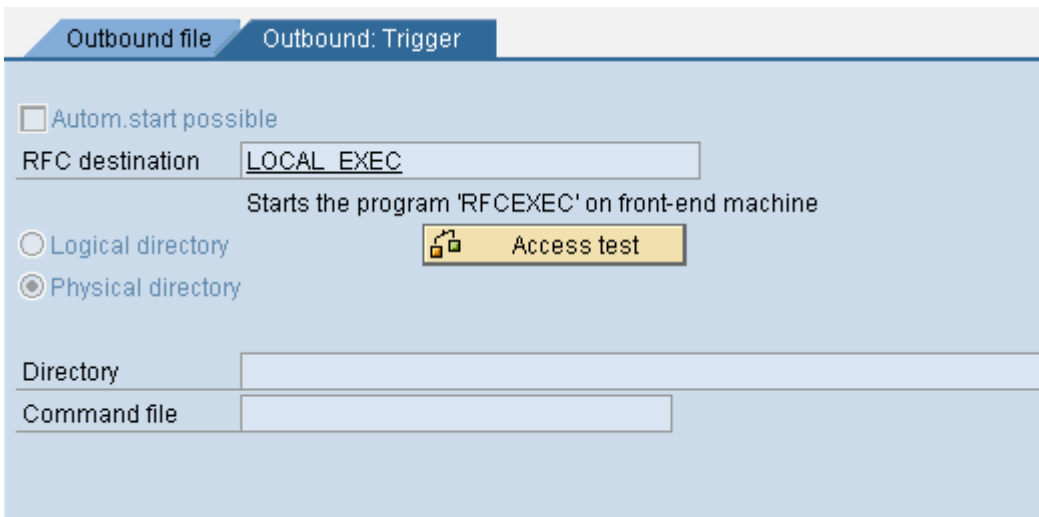


The screenshot shows the 'Outbound: Trigger' tab in the SAP configuration interface. It features several input fields and a button:

- Logical directory
- physical directory
- Access test button (with a lock icon)
- Directory:
- Function module:
- Description:
- Outbound file:

Here directory is the path on the application server. The Function Module is used for file naming conventions. Any of the SAP provided function modules could be used for this (Use F4 help to check on this) or create any custom function module for any other naming conventions.

In the outbound trigger tab, mention the RFC destination created earlier.



The screenshot shows the 'Outbound: Trigger' tab in the SAP configuration interface. It features several input fields and a button:

- Autom.start possible
- RFC destination:
- Starts the program 'RFCEXEC' on front-end machine
- Logical directory
- Physical directory
- Access test button (with a lock icon)
- Directory:
- Command file:

9. Make an entry in the partner profile generated earlier for message type MATMAS.

## Conversion of IDOCs to XML format at SAP Level

Partner No.	LOGSYSXML	Dummy logical system for XML placem
Partn.Type	LS	Logical system
Partner Role		
Message Type	MATMAS	
Message code		
Message function		<input type="checkbox"/> Test
<b>Outbound Options</b>   <b>Message Control</b>   <b>Post Processing: Permitted Agent</b>   <b>Tele...</b>		
Receiver port	XML_PORT	
<b>Output Mode</b>		
<input type="radio"/> Transfer IDoc Immed.	<input type="radio"/> Start subsystem	Output Mode
<input checked="" type="radio"/> Collect IDocs	<input checked="" type="radio"/> Do not start subsystem	
<b>IDoc Type</b>		
Basic type	MATMAS05	
Extension		
View		
<input checked="" type="checkbox"/> Cancel Processing After Syntax Error		
Seg. release in IDoc type		Segment Appl. Rel.

10. A background job need to be scheduled, for a periodic run (interval as required) for the program RBDMIDOC with the message type MATMAS.
11. Depending on the settings in the partner profiles, it may be necessary to send IDocs directly by executing the program RSEOUT00 (if the setting is to "Collect IDocs")

Test the above scenario by creating a material using MM01. An XML file would have been created in the directory specified in the XML port. The file could be downloaded onto the front-end system using the transaction CG3Y.