Synopsis: This document describes how to set iDOCs in SAP environment.

Number:	B000xxxxx
Updated:	16th March 2020
By:	Alexander Cimbalak, Bentley SK
State:	Draft
Scope:	**BECS CCOM integration**

Step by step guide

for setting of SAP ALE Outbound iDOC communication

1 Document mission 3

2 What is NOT included in this document 3

3 Brief description of SAP ALE 3

4 Step by step guide for setting of standard iDOC 9

4.1 Outbound iDoc Triggering Methods in SAP 9

4.1.1 Generating Outbound iDocs using Change Pointers 9

4.1.2 Creating Outbound iDocs via Output Determination 9

4.1.3 Generate Outbound iDocs using ABAP Programs 10

5 Stetting Outbound iDOC 11

5.1 Required Steps for standard iDOCs 11

5.1.1 Define logical system using tCode “Sale” 11

5.1.2 Setup RFC connection and PORT – tCode “SM59” 12

5.1.3 Test connection with receiver 12

5.1.4 Define IDOC port – tCode “we21” 12

5.1.5 Define partner profile – tCode “we20” 13

5.1.6 Define distribution model – tCode “BD64” 13

5.2 Define custom iDOC 14

5.2.1 Creation of Segment Types – tCode “we31” 14

5.2.2 Creation of iDoc type – tCode “we30” 14

5.2.3 Creation of logical message types – tCode “we81” 15

5.2.4 Linking Message type and IDoc type – tCode “we82” 15
6 References..... 16
7 Glossary..... 16

1 Document mission

This document was created as a documentation of setting of standard SAP ALE outbound iDOC communication. Main mission of this document is focusing only on setting outbound iDOCs related to BECS integration only. Because this process is standard SAP configuration we strictly recommend to take all information about and SAP configuration or customization from official sources of SAP A.G. and understand this guide only as a help in relation with all SAP official information.

2 What is NOT included in this document

This is not documentation of any general questions about SAP, SAP customization or any SAP setting not related to specific topic of setting SAP Outbound iDOCs.

3 Brief description of SAP ALE

Ale Technology is SAP's technology to support distributed yet integrated processes across several SAP systems.

Distributed Process:

A distributed process is one in which part of a business process is carried out on one system and part on another. The two systems would exchange data with each other at appropriate points to stay synchronized.

Need for Distributed Process:

- Business in Different Geographical Locations.
- Non availability of dedicated network.
- Cultural and language differences in Geographical locations.
- Running of Mission-critical Applications (Like Maintenance downtime etc.).
- Separate up gradation of Modules.

Distributed SAP SYSTEM - CHALLENGES

- A system that understands the syntax and semantics of the data. It was important from the very beginning to base the distribution of data on business rules, not on database replication techniques.
- Distributed systems that can maintain their autonomy while being integrated as one logical SAP system. The systems should be able to operate independently and support logical processing of transactions and data.
- Distributed systems that can handle different data models. A local implementation should be able to customize the system to meet its local needs.
- Receiving systems that can handle their own problems and not tie up the sending system.
- Systems that maintain continued operation in spite of network failure. Changes made to either system should be synchronized after the network connection is restored.
- A sound technology and methodology that can be used in all distribution scenarios.

SAP Distributed environment:

ALE allows for efficient and reliable communication between distributed processes across physically separate SAP systems.

ALE is based on application to application integration using messaging architecture. A message defines data that is exchanged between two processes. IDOCs are containers that hold data exchanged between the two systems.

Benefits of ALE:

- **Integration with non-SAP systems:** ALE architecture allows third party applications to integrate with SAP system.
- **Reliable Distribution:** Once message type created and the receiver of the message is determined, ALE delivers the message to the recipient. If there is any network problem it will buffer the message and delivers the message once the network is restored. It also ensures that the message is not delivered twice.
- **Release Upgrade:** Any of the distributed system can be upgraded to the new release of SAP without affecting the functionality. The ALE layer ensures backward compatibility of messages exchanged between systems.

ALE Architecture:
It consists of an Outbound process, an Inbound process, and an Exception - Handling process.
Outbound Process:
ALE Outbound Process in SAP sends data to one or more SAP Systems. It involves four steps.

1. **Identify the need of IDOC:** This step starts upon creating a application document, can relate to a change to a master data object.
2. **Generate the Master IDOC:** The document or master data to be sent is read from the database and formatted into an IDOC format. This IDOC is called as a Master IDOC.
3. **Generate the Communication IDOC:** The ALE Service layer generates a separate IDOC from the Master IDOC for each recipient who is interested in the data. Separate IDOCs are generated because each recipient might demand a different version or a subset of the Master IDOC. These recipient-specific IDOCs are called Communication IDOCs and are stored in the database.
4. **Deliver the Communication IDOC:** The IDOC is delivered to the recipients using an asynchronous communication method. This allows the sending system to continue its processing without having to wait for the destination system to receiver or process the IDOC.

Inbound Process:
The inbound process receives an IDOC and creates a document in the system.

1. **Store the IDOC in the database:** The IDOC is received from the sending system and stored in the database. Then the IDOC goes through a basic integrity check and syntax check.
2. **Invoke the Posting Module:** The control information in the IDOC and configuration tables are read to determine the posting program. The IDOC is then transferred to its posting program.
3. **Create the Document:** The posting program reads the IDOC data and then creates a document in the system. The results are logged in the IDOC.**Over view of IDOCs:**
IDOC is a container that is used to exchange data between any two processes. The document represented in an IDOC is independent of the complex structure SAP uses to store application data. This type of flexibility enables SAP to rearrange its internal structure without affecting the existing interface. IDOC interface represents an IDOC Type or IDOC data. IDOC Type represents IDOC's definition and IDOC Data is an instance of the IDOC Type.
IDOC Types:

IDOC type structure can consist of several segments, and each segment can consist of several data fields. The IDOC structure defines the syntax of the data by specifying a list of permitted segments and arrangement of the segments. Segments define a set of fields and their format.
An IDOC is an instance of an IDOC Type and consists of three types of records.

i. **One Control record:** each IDOC has only one control record. The control record contains all the control information about an IDOC, including the IDOC number, the sender and recipient information, and information such as the message type it represents and IDOC type. The control record structure is same for all IDOCs.

ii. **One or Many Data records:** An IDOC can have multiple data records, as defined by the IDOC structure. Segments translate into data records, which store application data, such as purchase order header information and purchase order detail lines.

iii. **One or Many Status records:** An IDOC can have multiple status records. Status record helps to determine whether an IDOC has any error.

Message in IDOC Type:

A Message represents a specific type of document transmitted between two partners.

Outbound Process in IDOCs:

Outbound process used the following components to generate an IDOC. A customer model, and IDOC structure, selection programs, filter objects, conversion rules, a port definition, an RFC destination, a partner profile, service programs, and configuration tables.

The Customer Model:

A customer model is used to model a distribution scenario. In a customer model, you identify the systems involved in a distribution scenario and the message exchanged between the systems.

Message control:

Message control is a cross application technology used in pricing, account determination, material determination, and output determination. The output determination technique of Message control triggers the ALE for a business document. Message control separates the logic of generating IDOCs from the application logic.

Change Pointers:

The change pointers technique is based on the change document technique, which tracks changes made to key documents in SAP, such as the material master, customer master and sales order.
Changes made to a document are recorded in the change document header table CDHDR, and additional change pointers are written in the BDCP table for the changes relevant to ALE.

IDOC Structure:

A message is defined for data that is exchanged between two systems. The message type is based on one or more IDOC structures.

Selection Program:

Is typically implemented as function modules, are designed to extract application data and create a master IDOC. A selection program exists for each message type. A selection program's design depends on the triggering mechanism used in the process.

Filter Objects;

Filter Objects remove unwanted data for each recipient of the data basing on the recipients requirement.

Port Definition:

A port is used in an outbound process to define the medium in which documents are transferred to the destination system. ALE used a Transactional RFC port, which transfers data in memory buffers.

RFC Destination:

The RFC destination is a logical name used to define the characteristics of a communication link to a remote system on which a function needs to be executed.

Partner Profile:

A partner profile specifies the components used in an outbound process(logical name of the remote SAP system, IDOC Type, message type, TRFC port), an IDOC's packet size, the mode in which the process sends an IDOC (batch versus immediate), and the person to be notified in case of error.

Service Programs and Configuration Tables:

The outbound process, being asynchronous, is essentially a sequence of several processes that work together. SAP provides service programs and configuration tables to link these programs and provide customizing options for an outbound process.

Process flow for Distributing Transactional Data:

Transactional data is distributed using two techniques: with Message control and without message control.

Process flow for Distributing Master Data:

Master data between SAP systems is distributed using two techniques: Stand alone Programs and Change Pointers.

Triggering the Outbound Process via Stand-Alone Programs:

Stand-Alone programs are started explicitly by a user to transmit data from one SAP system to another. Standard Programs for several master data objects exist in SAP. Ex. The material master data can be transferred using the RBDSEMAT program or transaction BD10.

The stand-alone programs provide a selection screen to specify the objects to be transferred and the receiving system. After the stand-alone program is executed, it calls the IDOC selection program with the specified parameters.

Triggering the Outbound Process via Change Pointers:

The change pointer technique is used to initiate the outbound process automatically when master data is created or changed.

A standard program, RBDMIDOC, is scheduled to run on a periodic basis to evaluate the change pointers for a message type and start the ALE process for distributing the master data to the appropriate destination. The RBDMIDOC program reads the table TBDME to determine the IDOC selection program for a message type.

Processing in the Application Layer:

The customer distribution model is consulted to make sure that a receiver has been defined for the message to be transmitted. If not, processing ends. If at least one receiver exists, the IDOC selection program reads the master data object from the database and creates a master IDOC from it. The master IDOC is stored in memory. The program then calls the ALE service layer by using the function module MASTER_IDOC_DISTRIBUTE, passing the master IDOC and the receiver information.

Processing in the ALE Interface Layer:

Processing in the ALE Layer consists of the following steps:

- **Receiver Determination**: The determination of the receiver is done through Customer Distribution Model.
- **IDOC Filtering**: if an IDOC filter is specified in the distribution model for a receiver, values in the filter are compared against the values in the IDOC data records. If a data record does not meet the filter criteria, it is dropped.
- **Segment Filtering**: For each sender and receiver combination, a set of segments that are not required can be filtered out.
- **Field conversion**: Field values in data records are converted by using the conversion rules specified for the segment.
- **Version change for segments**: Segments are version-controlled. A new version of a segment always contains fields from the preceding version and fields added for the new version. Release in IDOC type field of the partner profile to determine the version of the segment to be generated.

- **Version change for IDOCs**: IDOCs are also version controlled. The version is determined from the Basic Type field of the partner profile.
- **Communication IDOCs generated**: The final IDOC generated for a receiver after all the conversions and filtering operations is the communication IDOC. One master IDOC can have multiple communication IDOCs depending on the number of receivers identified and the filter operations performed. IDOC gets the status record with a status code of 01 (IDOC Created).
- **Syntax check performed**: IDOC goes through a syntax check and data integrity validation. If errors found the IDOC get the status of 26 (error during syntax check of IDOC - Outbound). If no errors found the IDOC gets the status 30 (IDOC ready for dispatch - ALE Service).
- **IDOC dispatched to the communication Layer**: In the ALE process, IDOCs are dispatched using the asynchronous RFC method, which means that the sending system does not await for data to be received or processed on the destination system. After IDOCs have been transferred to the communication layer, they get a status code 01 (Data Passed to Port OK).

Processing in the Communication Layer:
To dispatch an IDOC to a destination system, the system reads the port definition specified in the partner profile to determine the destination system, which is then used to read the RFC destination. The RFC destination contains communication settings to log o to the remote SAP system. The sending system calls the INBOUND_IDOC_PROCESS function module asynchronously on the destination system and passes the IDOC data via the memory buffers.

Inbound Process in IDOCs:
An inbound process used IDOC structure, posting programs, filter objects, conversion rules, a partner profile, service programs, and configuration tables to post an application document from an IDOC.

Posting Program:
Posting programs, which are implemented as function modules, read data from an IDOC and create an application document from it. A posting program exists for each message. Each posting program is assigned a process code. A process code can point to a function module or a work flow. In the standard program process codes always point to a function module.
Ex. The posting program for message type MATMAS is IDOC_INPUT_MATMAS which has a process code MATM.

Workflow:
A workflow represents a sequence of customized steps to be carried out for a process. The workflow management system is used to model the sequence, identify information required to carry out the steps and identify the person responsible for the dialog steps.

Partner Profile;
A partner profile specifies the components used in an inbound process (partner number, message type, and process code), the mode in which IDOCs are processed (batch versus immediate), and the person to be notified in case of errors.

Process flow for the Inbound process via a Function Module:
In this process, IDOCs are received from another system and passed to the posting function module directly.

1. **Processing in the communication Layer:**

   The IDOC_INBOUND_ASYCHRONOUS program, triggered as a result of an RFC from the sending system, acts as the entry point for all inbound ALE processes. The IDOC to be processed is passed as an input parameter. Control is transferred to the ALE/EDI layer.

1. **Processing in the ALE/EDI Interface Layer:**

- **Basic integrity check**: A basic integrity check is performed on the control record.
- **Segment Filtering and conversion**: Filtering out unwanted segments and carry out any required conversion of field values.
- **Creation of Application IDOC**: The application IDOC is created and stored in the database and a syntax check is performed. If there are errors it gets status code of 60 (Error during Syntax check of IDOC - Inbound). At this point a tangible IDOC, which can be monitored via one of the monitoring transactions, is created and the IDOC gets status code 50 (IDOC Added).
- **IDOC Marked ready for Dispatch**: IDOC gets the status code 64 (IDOC ready to be passed to application).
- **IDOC is passed to the posting program**: The partner profile table is read. If the value of the Processing field is set to Process Immediately, the IDOC is passed to the posting program immediately using the program RBDAPP01.

1. **Processing in the Posting Module:**

The process code in the partner profile points to a posting module for the specific message in the IDOC. The posting program implemented as a function module either calls a standard SAP transaction by using the Call Transaction command for posting the document or invokes a direct input function module. The results of execution are passed back via the function module's output parameters. If the posting is successful IDOC gets the status code 53 (Application Document Posted) or it gets status code 51 (Error: Application Document Not Posted).

4 Step by step guide for setting of standard iDOC

In this chapter, we will look at an end-to-end EDI transmission that utilizes outbound iDocs and Output Determination. Not only will we examine Outbound iDoc and Output Determination configuration steps in detail, but also discuss different outbound iDoc generating methods in SAP, as well as how to troubleshoot and test an outbound iDoc scenario.

Moreover, we will look at some common error messages of iDoc and Output control.

4.1 Outbound iDoc Triggering Methods in SAP

There are several methods to generate outbound iDocs in SAP. Each of them serves a different purpose.

The main methods of generating outbound iDocs in SAP are,

1. Using ALE **Change Pointer** mechanism
2. Via **Output Determination**
3. Using **custom ABAP Program**

4.1.1 Generating Outbound iDocs using Change Pointers

To transfer **master data** between different systems we can use the ALE/iDoc framework and, particularly, its **Change Pointers** functionality. In most organizational system landscapes, SAP is the central system that manages master data such as Customer master, Vendor master, Materials master, etc. Using iDocs issued by Change Pointers you can transfer master data from the central SAP system to other systems in the landscape and other integrated systems.

You can configure Change Pointers to **flag master data changes** (Create/Update/Delete) in SAP. Using these flags, outbound iDocs generated in SAP can be directed to different receiving systems. The flagged master data changes are stored in Change Pointers table BDCP2 in SAP S4 HANA.

With the help of program **RBDMIDOC**, entries stored in the BDCP2 table can be processed to generate outbound iDocs.

4.1.2 Creating Outbound iDocs via Output Determination

Similar to how we use Change Pointers for master data distribution, the **Output determination** technique can be used to **generate iDocs for transactional data**. To send transactional data such as sales orders, Purchase Orders, Delivery, Shipment Confirmations, etc., to partners and their systems, output types can be configured in SAP.

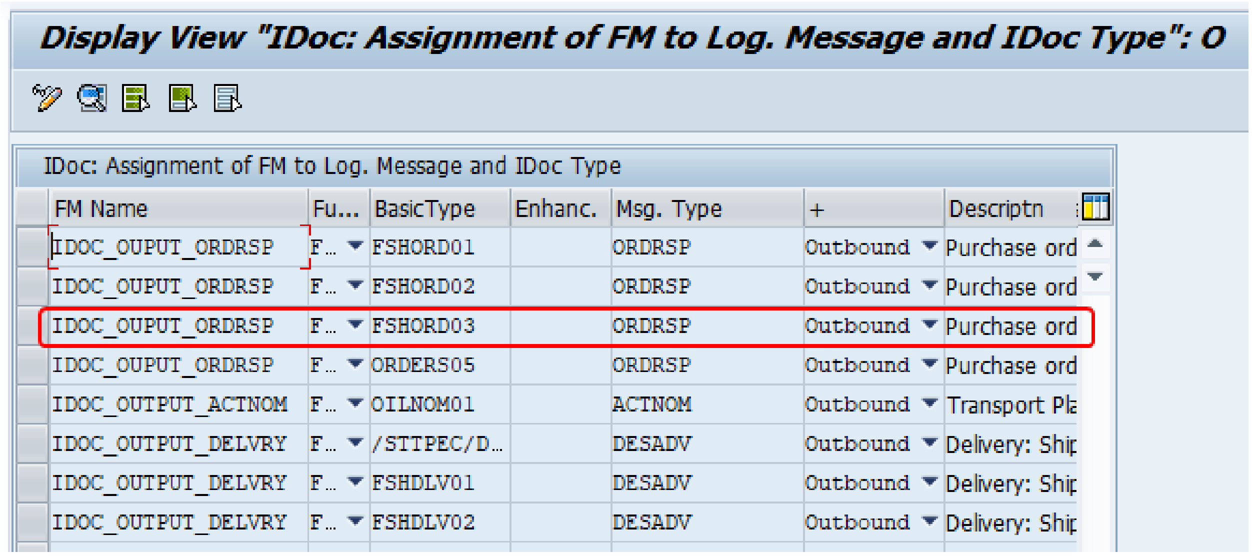
In certain business processors every time a Purchase Order is created in SAP, the PO should be sent to the vendor of the Purchase Order. Output determination can be configured to trigger an EDI Output Type to vendor when a Purchase Order is created in SAP. Outbound iDocs generated from these EDI Output Types can be transferred to the vendor or external system.

In this document, we will look at how to configure the **Output Determination** in **Condition Record technique** and how to trigger **Outbound iDocs** from the generated outputs.

4.1.3 Generate Outbound iDocs using ABAP Programs

Although **not as common** as Change Pointers or Output Determination methods, **custom ABAP programs** are sometimes used to create iDocs.

To generate Outbound iDocs from ABAP programs, first, you need to find the Outbound iDoc that creates Function Module of the iDoc Message type and iDoc Basic type. Go to transaction we57 to find the FM you can use in an ABAP program to generate outbound iDocs .



Display View "IDoc: Assignment of FM to Log. Message and IDoc Type": O

IDoc: Assignment of FM to Log. Message and IDoc Type

FM Name	Fu...	BasicType	Enhanc.	Msg. Type	+	Descriptn
IDOC_OUPUT_ORDRSP	F...	FSHORD01		ORDRSP	Outbound	Purchase ord
IDOC_OUPUT_ORDRSP	F...	FSHORD02		ORDRSP	Outbound	Purchase ord
IDOC_OUPUT_ORDRSP	F...	FSHORD03		ORDRSP	Outbound	Purchase ord
IDOC_OUPUT_ORDRSP	F...	ORDERS05		ORDRSP	Outbound	Purchase ord
IDOC_OUTPUT_ACTNOM	F...	OILNOM01		ACTNOM	Outbound	Transport Pl
IDOC_OUTPUT_DELVRY	F...	/STTPEC/D...		DESADV	Outbound	Delivery: Ship
IDOC_OUTPUT_DELVRY	F...	FSHDLV01		DESADV	Outbound	Delivery: Ship
IDOC_OUTPUT_DELVRY	F...	FSHDLV02		DESADV	Outbound	Delivery: Ship

For example use the Function Module ‘IDOC_OUPUT_ORDRSP’ in your custom ABAP program to generate ORDRSP iDocs.

If the outbound iDoc is a master data iDoc, use the FM ‘MASTER_IDOC_DISTRIBUTE’ in the ABAP program to create iDocs.

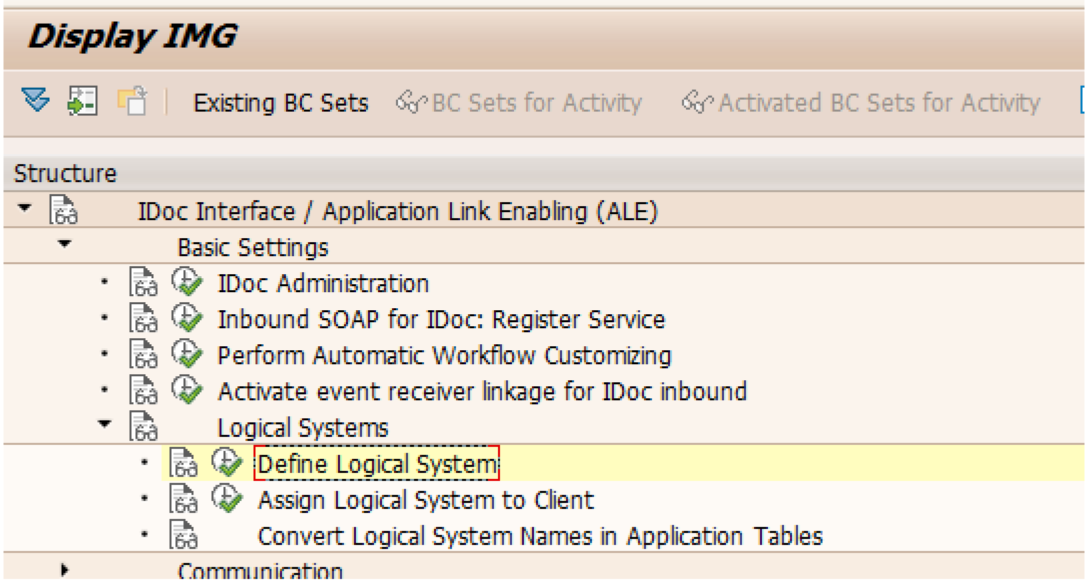
5 Stetting Outbound iDOC

5.1 Required Steps for standard iDOCs

This process can be used for setting all standard iDOC required for standard BECS integration.:

Standard SAP Message Types	Basic Type used	SAP Object related
FUNC_LOC_CHANGE	FUNC_LOC_CHANGE02	Functional Location
EQUIPMENT_CREATE	EQUIPMENT_CREATE02	Equipment
INOTIF	INOTIF01	PM Notification
IORDER	IORDER01 (extension ZORDER01)	PM Work Order

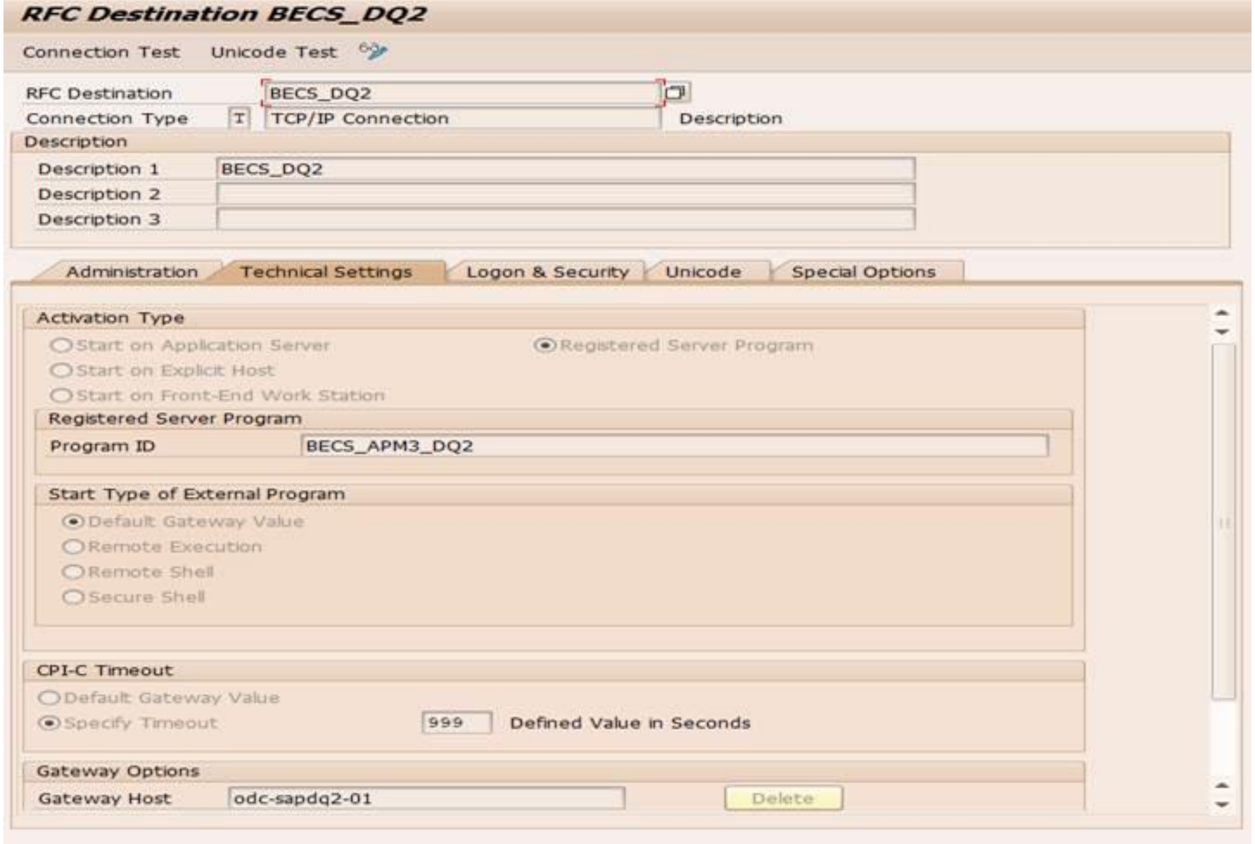
5.1.1 Define logical system using tCode "Sale"



Display IMG

Existing BC Sets
BC Sets for Activity
Activated BC Sets for Activity
Structure
IDoc Interface / Application Link Enabling (ALE)
Basic Settings
IDoc Administration
Inbound SOAP for IDoc: Register Service
Perform Automatic Workflow Customizing
Activate event receiver linkage for IDoc inbound
Logical Systems
Define Logical System
Assign Logical System to Client
Convert Logical System Names in Application Tables
Communication

5.1.2 Setup RFC connection and PORT – tCode “SM59”



RFC Destination BECS_DQ2

Connection Test Unicode Test

RFC Destination: BECS_DQ2

Connection Type: T TCP/IP Connection Description

Description

Description 1: BECS_DQ2

Description 2:

Description 3:

Administration | Technical Settings | Logon & Security | Unicode | Special Options

Activation Type

○ Start on Application Server ● Registered Server Program

○ Start on Explicit Host

○ Start on Front-End Work Station

Registered Server Program

Program ID: BECS_APM3_DQ2

Start Type of External Program

● Default Gateway Value

○ Remote Execution

○ Remote Shell

○ Secure Shell

CPI-C Timeout

○ Default Gateway Value

● Specify Timeout 999 Defined Value in Seconds

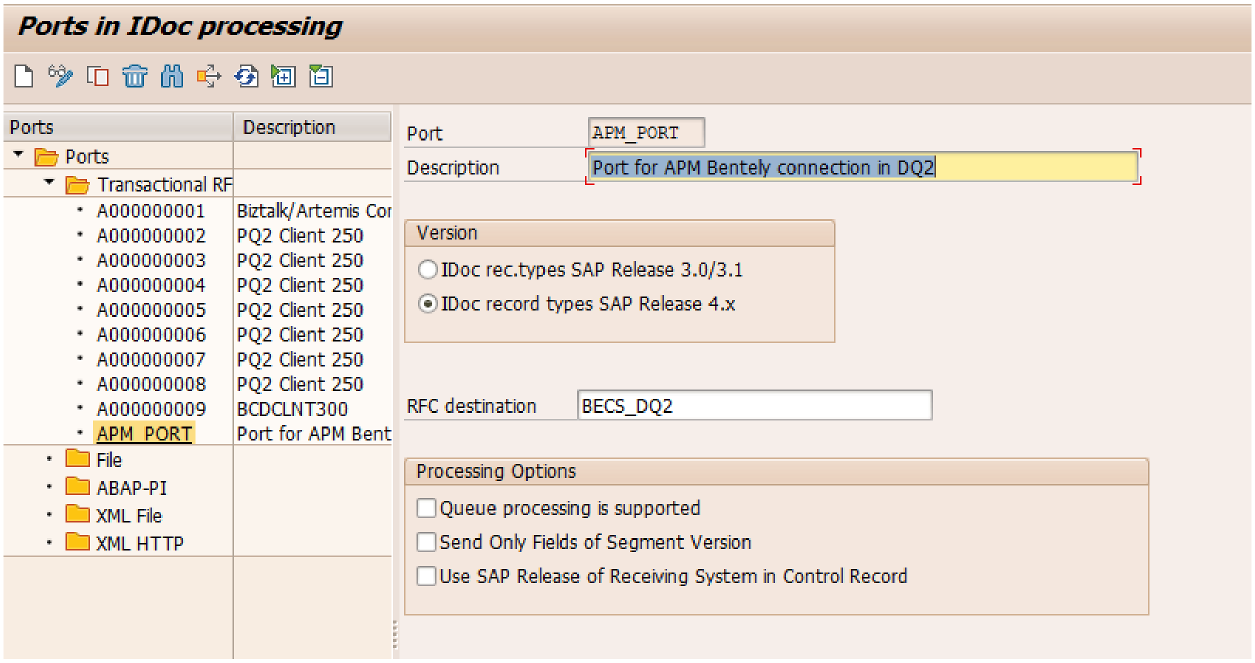
Gateway Options

Gateway Host: odc-sapdq2-01 Delete

5.1.3 Test connection with receiver

Check the connection success by pressing “Connection Test” in previous screen

5.1.4 Define IDOC port – tCode “we21”



Ports in IDoc processing

Ports	Description
Ports	
Transactional RF	
A000000001	Biztalk/Artemis Cor
A000000002	PQ2 Client 250
A000000003	PQ2 Client 250
A000000004	PQ2 Client 250
A000000005	PQ2 Client 250
A000000006	PQ2 Client 250
A000000007	PQ2 Client 250
A000000008	PQ2 Client 250
A000000009	BCDCLNT300
APM_PORT	Port for APM Bent
File	
ABAP-PI	
XML File	
XML HTTP	

Port: APM_PORT

Description: Port for APM Bentely connection in DQ2

Version

○ IDoc rec.types SAP Release 3.0/3.1

● IDoc record types SAP Release 4.x

RFC destination: BECS_DQ2

Processing Options

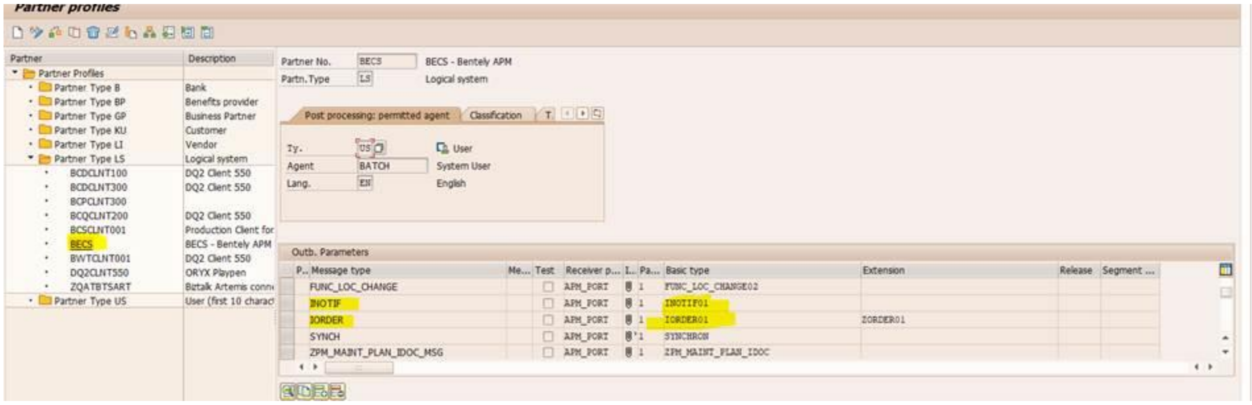
□ Queue processing is supported

□ Send Only Fields of Segment Version

□ Use SAP Release of Receiving System in Control Record

5.1.5 Define partner profile – tCode “we20”

This step can be done automatically from distribution model – see screen below.



Partner	Description
Partner Profiles	
Partner Type B	Bank
Partner Type BP	Benefits provider
Partner Type GP	Business Partner
Partner Type KU	Customer
Partner Type LI	Vendor
Partner Type LS	Logical system
BCDCLNT100	DQ2 Client 550
BCDCLNT300	DQ2 Client 550
BCPCLNT300	
BCQCLNT200	DQ2 Client 550
BCSCLNT001	Production Client for
BECS	BECS - Bentely APM
BWTCLNT001	DQ2 Client 550
DQ2CLNT550	ORYX Playpen
ZQATBTSART	Biztalk Artemis conn
Partner Type US	User (first 10 charac

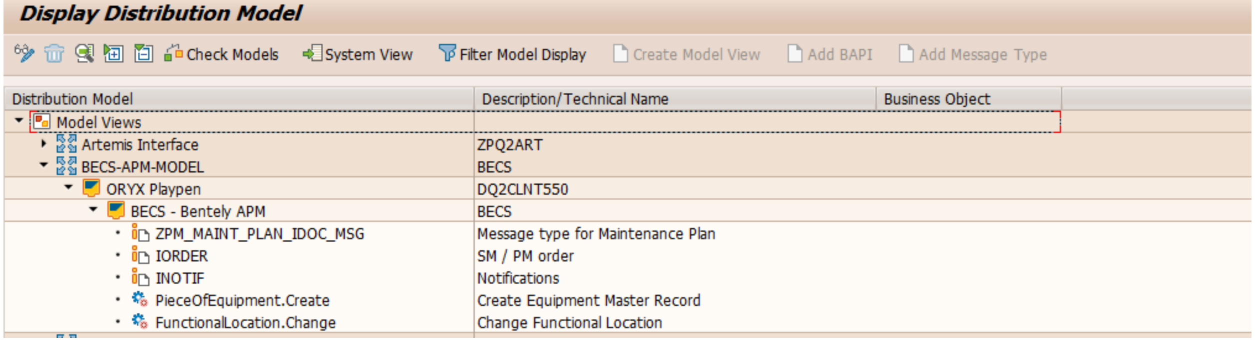
Partner No. BECS BECS - Bentely APM
Partn.Type LS Logical system

Ty. US User
Agent BATCH System User
Lang. EN English

Outb. Parameters

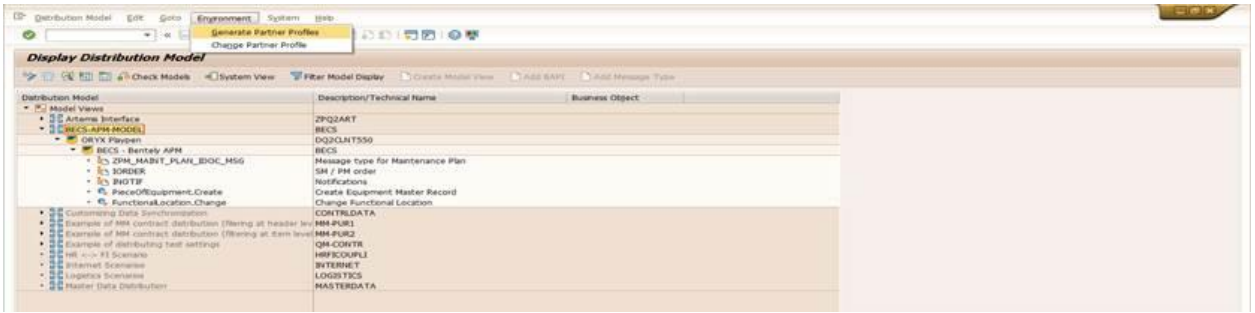
P..	Message type	Me...	Test	Receiver p...	I...	Pa...	Basic type	Extension	Release	Segment ...
	FUNC_LOC_CHANGE		☐	APM_PORT		1	FUNC_LOC_CHANGE02			
	INOTIF		☐	APM_PORT		1	INOTIF01			
	IORDER		☐	APM_PORT		1	IORDER01	ZORDER01		
	SYNCH		☐	APM_PORT		1	SYNCHRON			
	ZPM_MAINT_PLAN_IDOC_MSG		☐	APM_PORT		1	ZPM_MAINT_PLAN_IDOC			

5.1.6 Define distribution model – tCode “BD64”



Distribution Model	Description/Technical Name	Business Object
Model Views		
Artemis Interface	ZPQ2ART	
BECS-APM-MODEL	BECS	
ORYX Playpen	DQ2CLNT550	
BECS - Bentely APM	BECS	
ZPM_MAINT_PLAN_IDOC_MSG	Message type for Maintenance Plan	
IORDER	SM / PM order	
INOTIF	Notifications	
PieceOfEquipment.Create	Create Equipment Master Record	
FunctionalLocation.Change	Change Functional Location	

Generate partner profile (if not done manually in 5.1.5)



Distribution Model	Description/Technical Name	Business Object
Model Views		
Artemis Interface	ZPQ2ART	
BECS-APM-MODEL	BECS	
ORYX Playpen	DQ2CLNT550	
BECS - Bentely APM	BECS	
ZPM_MAINT_PLAN_IDOC_MSG	Message type for Maintenance Plan	
IORDER	SM / PM order	
INOTIF	Notifications	
PieceOfEquipment.Create	Create Equipment Master Record	
FunctionalLocation.Change	Change Functional Location	
Customizing Data Synchronization	CONTRLDATA	
Example of MM contract distribution (filtering at header lev	MM-PUR1	
Example of MM contract distribution (filtering at item level	MM-PUR2	
Example of distributing test settings	QM-CONTR	
HR <--> FI Scenario	HRFICOUPLI	
Internet Scenarios	INTERNET	
Logistics Scenarios	LOGISTICS	
Master Data Distribution	MASTERDATA	

Refer below blog for details

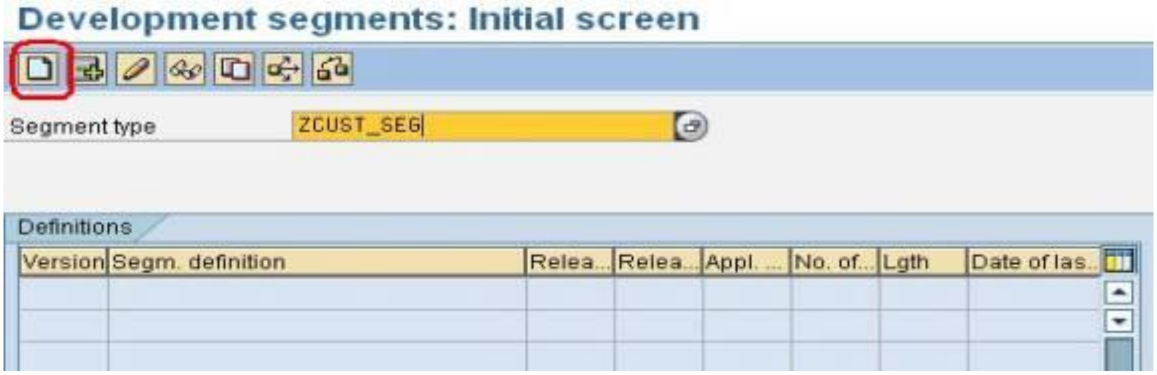
https://sapintegrationhub.com/abap/ale-idoc/outbound-idoc-configuration-output-determination-techno-functional-sap/

5.2 Define custom iDOC

In respect of covering custom iDOC (for example for Maintenance Plan) please use following process:

5.2.1 Creation of Segment Types – tCode “we31”

Run T-code ‘WE31’ to create segment type, which has fields to contain the data and are added to the segment type in the same transaction. The data stored contained into the segment mesh is finally stored in EDISEGMENT table.



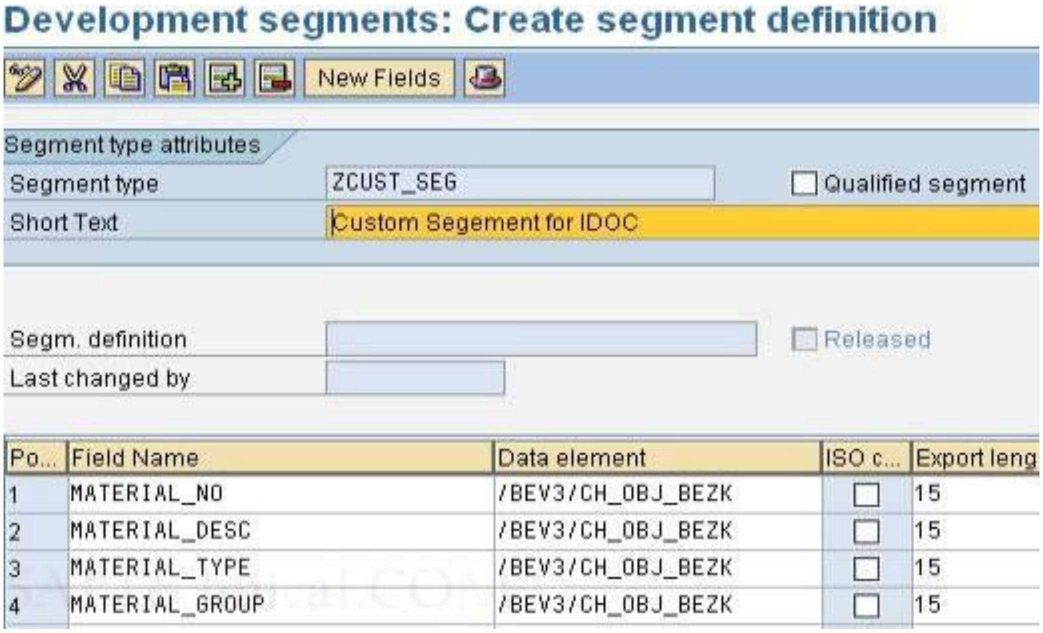
Development segments: Initial screen

Segment type: ZCUST_SEG

Definitions

Version	Segm. definition	Relea...	Relea...	Appl. ...	No. of...	Lgth	Date of las...

Add your custom fields as per business scenario.



Development segments: Create segment definition

Segment type attributes

Segment type: ZCUST_SEG □ Qualified segment

Short Text: Custom Segement for IDOC

Segm. definition: □ Released

Last changed by:

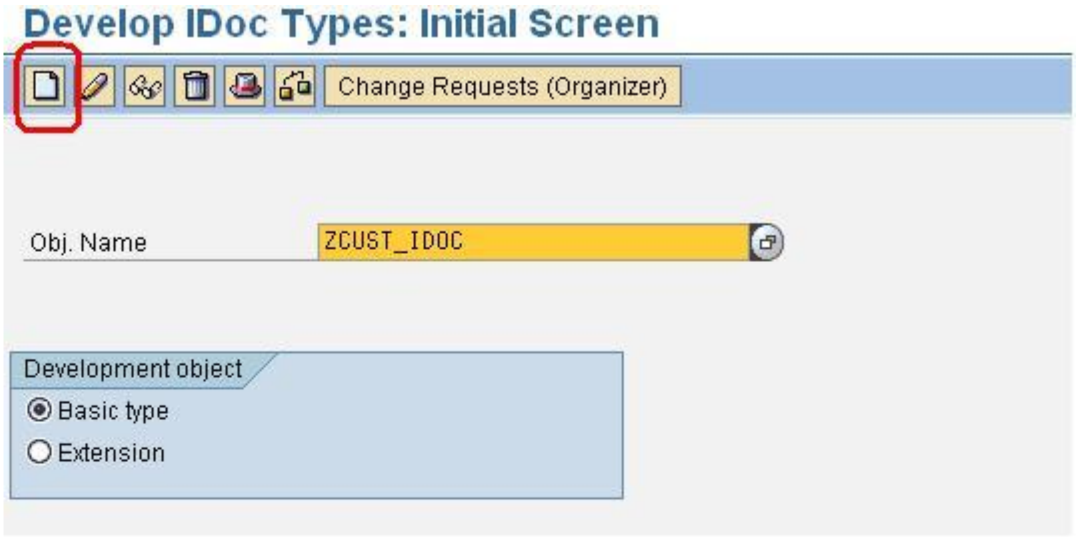
Po...	Field Name	Data element	ISO c...	Export leng
1	MATERIAL_NO	/BEV3/CH_OBJ_BEZK	□	15
2	MATERIAL_DESC	/BEV3/CH_OBJ_BEZK	□	15
3	MATERIAL_TYPE	/BEV3/CH_OBJ_BEZK	□	15
4	MATERIAL_GROUP	/BEV3/CH_OBJ_BEZK	□	15

To make it available to other transactions, release the segment created.

Go to **Edit -> Set Release**

5.2.2 Creation of iDoc type – tCode “we30”

Run T-code ‘WE30’ to create custom IDoc type. Enter the name of custom IDoc you want to create and click on red box for creation.



Develop IDoc Types: Initial Screen

Obj. Name: ZCUST_IDOC

Development object

● Basic type

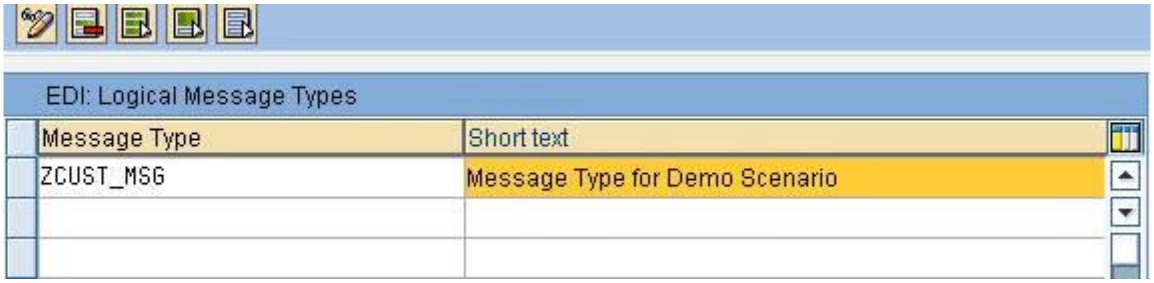
○ Extension

Now, it takes you to following screen. Here you can add description for your IDoc type. Also you can specify name of existing IDoc for Copy or Successor mode.

Now, you can maintain attributes of the custom IDoc, which consists of attaching segment type created above to the IDoc type. Also specifying the details of frequency of these segments getting filled i.e. Maximum and Minimum number. Fill the necessary details and release this IDoc type as well.

5.2.3 Creation of logical message types – tCode "we81"

Run T-code 'WE81' to create the logical message types. Go to Edit mode and click New Entry to go to following screen.



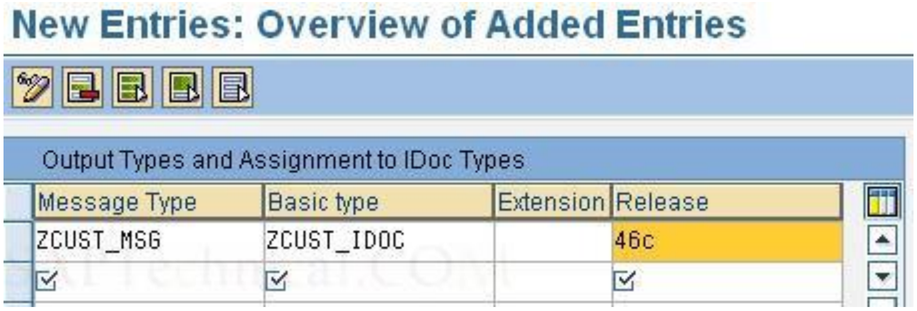
EDI: Logical Message Types

Message Type	Short text
ZCUST_MSG	Message Type for Demo Scenario

Save the entered data.

5.2.4 Linking Message type and IDoc type – tCode "we82"

Run T-Code 'WE82'. Now we have to link these created IDoc types and Message types. Enter the message type name, Basic IDoc type (ZCUST_IDC) and release type to be linked. In the data transfer through ALE, these message types represent the IDOC structure.



New Entries: Overview of Added Entries

Output Types and Assignment to IDoc Types

Message Type	Basic type	Extension	Release
ZCUST_MSG	ZCUST_IDOC		46c
☑	☑		☑

Extension field above will be used as per the Extended IDoc type scenario i.e. in case of addition of few more fields into the existing IDoc type.

Hence, now our Creation of Custom IDoc is ready to use in ALE scenarios.

6 References

Number	Revision	Title	Status

7 Glossary

Term	Description

End of document