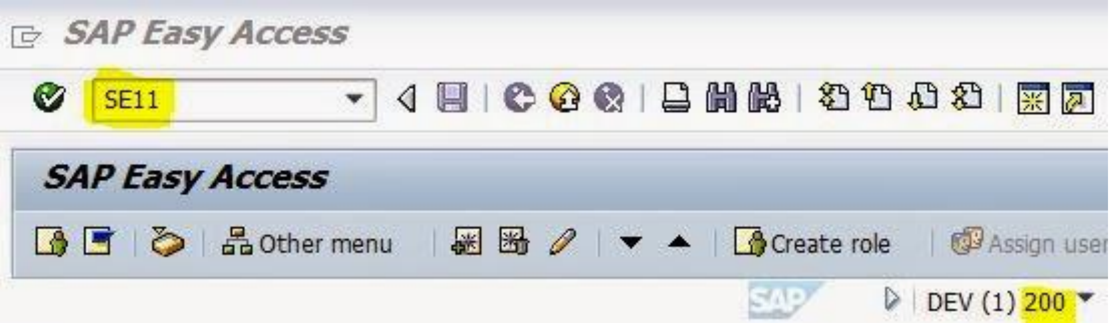
Transferring table records from One Client to Another Client in same SAP System with customer report By IDoc

November 22, 2015 coderobbot Uncategorized Leave a comment

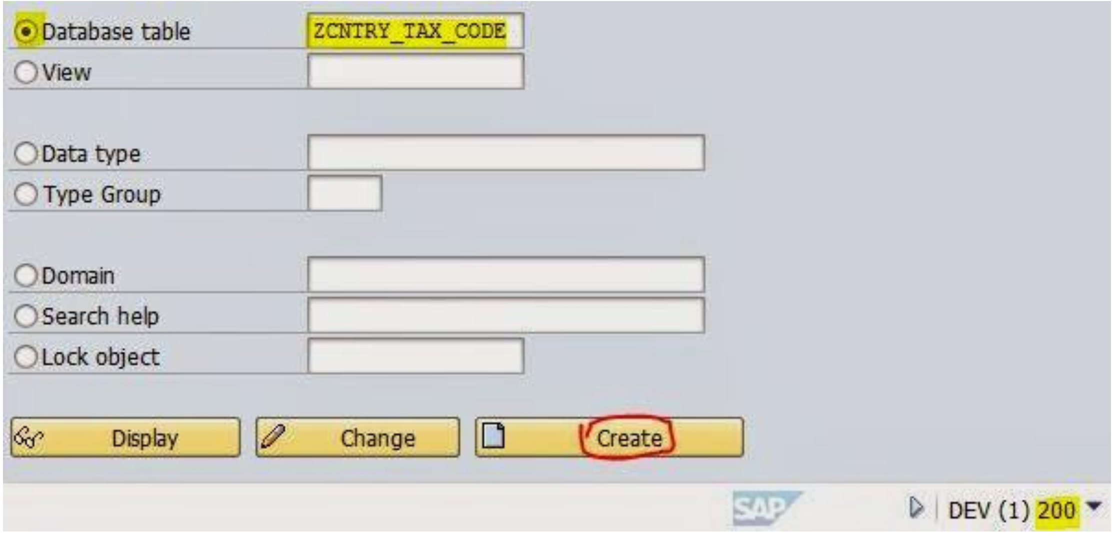
---

**Step1. Go to Tcode-Se11.**



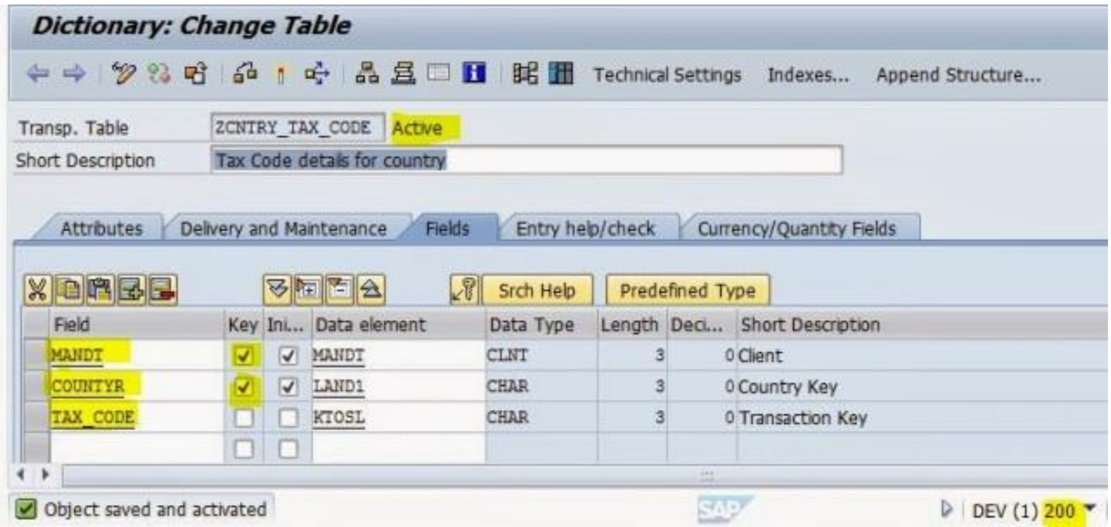
SE11

**Step2. Provide the table name and click on create button.**



● Database table ZCNTRY_TAX_CODE
○ View
○ Data type
○ Type Group
○ Domain
○ Search help
○ Lock object

**Step3. Provide the detail fields and activate the table. Create some records.**



Dictionary: Change Table

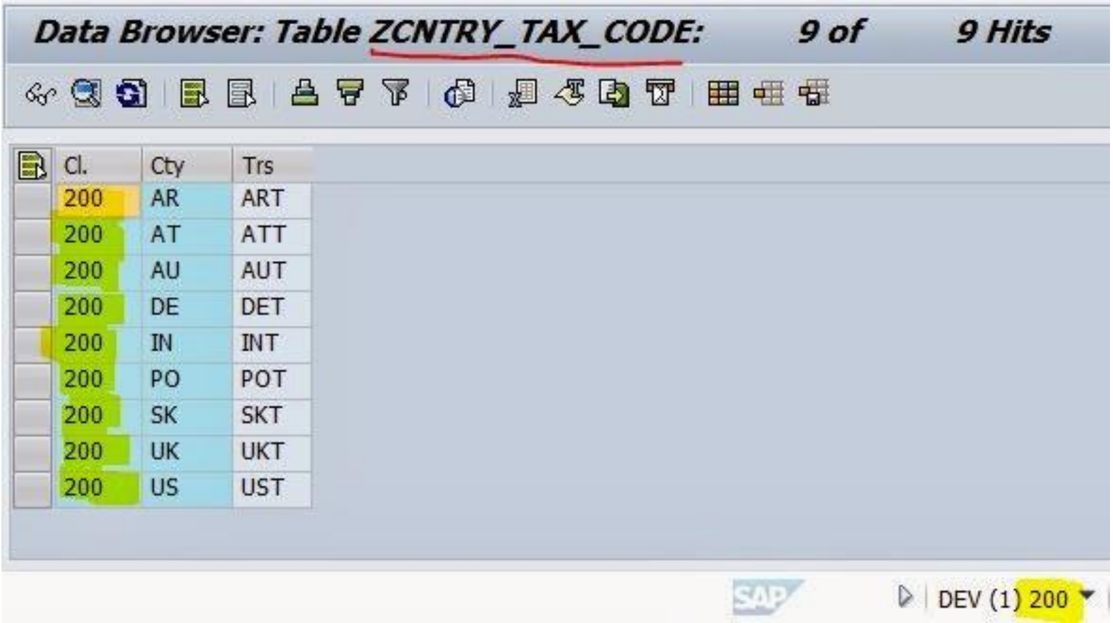
Transp. Table ZCNTRY_TAX_CODE Active

Short Description Tax Code details for country

| Field | Key | Ini... | Data element | Data Type | Length | Deci... | Short Description |
|---|---|---|---|---|---|---|---|
| MANDT | ☑ | ☑ | MANDT | CLNT | 3 | 0 | Client |
| COUNTRY | ☑ | ☑ | LAND1 | CHAR | 3 | 0 | Country Key |
| TAX_CODE | □ | □ | KTOSL | CHAR | 3 | 0 | Transaction Key |
|  | □ | □ |  |  |  |  |  |

Object saved and activated

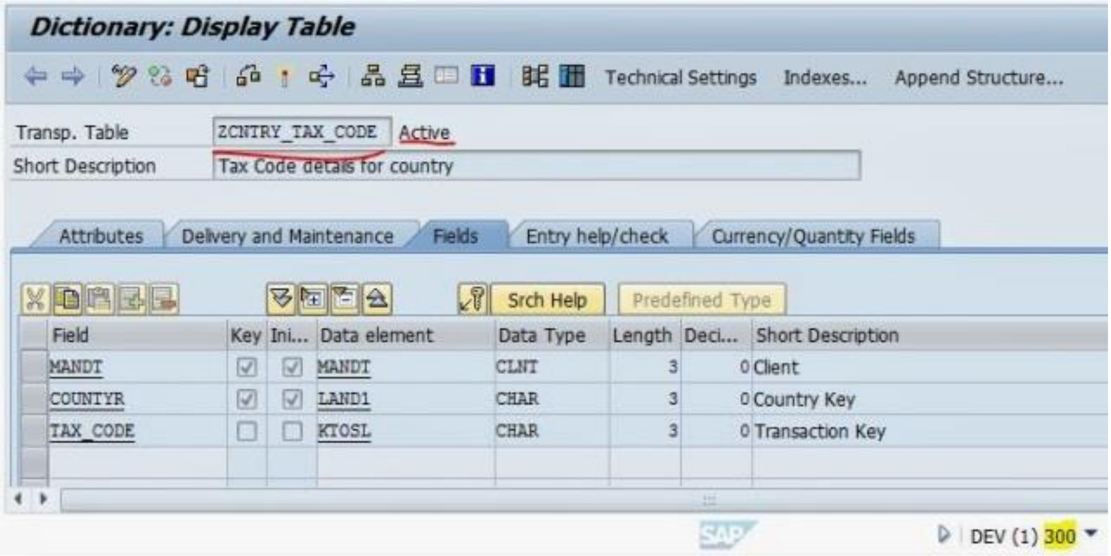
**Step4. The records of the table are displayed below.**



Data Browser: Table ZCNTRY_TAX_CODE: 9 of 9 Hits

| Cl. | Cty | Trs |
|---|---|---|
| 200 | AR | ART |
| 200 | AT | ATT |
| 200 | AU | AUT |
| 200 | DE | DET |
| 200 | IN | INT |
| 200 | PO | POT |
| 200 | SK | SKT |
| 200 | UK | UKT |
| 200 | US | UST |

**Step5. The same table is available in the client 300.**



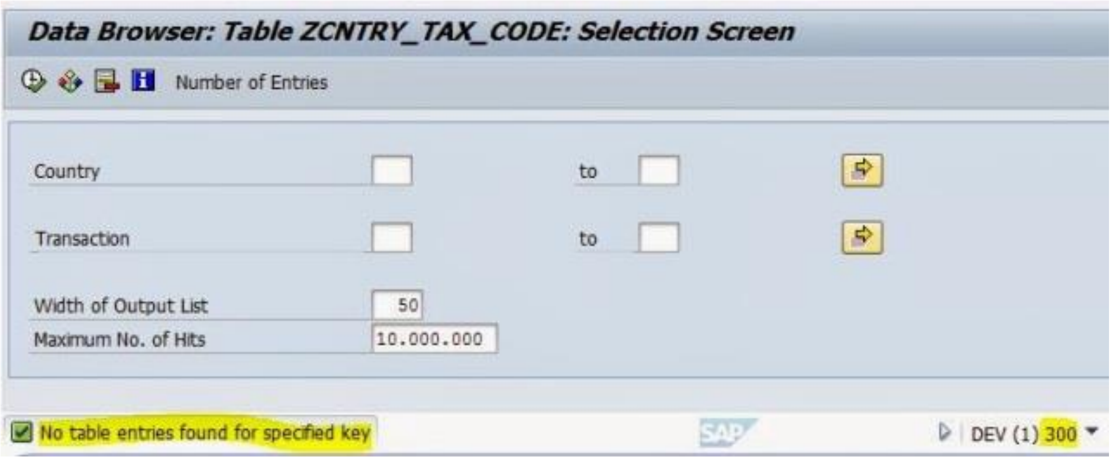
Dictionary: Display Table

Transp. Table: ZCNTRY_TAX_CODE Active

Short Description: Tax Code details for country

| Field | Key | Ini... | Data element | Data Type | Length | Deci... | Short Description |
|---|---|---|---|---|---|---|---|
| MANDT | ☑ | ☑ | MANDT | CLNT | 3 | 0 | Client |
| COUNTYR | ☑ | ☑ | LAND1 | CHAR | 3 | 0 | Country Key |
| TAX_CODE | □ | □ | KTOSL | CHAR | 3 | 0 | Transaction Key |

**Step6. There are no records present in the table for the client 300.**



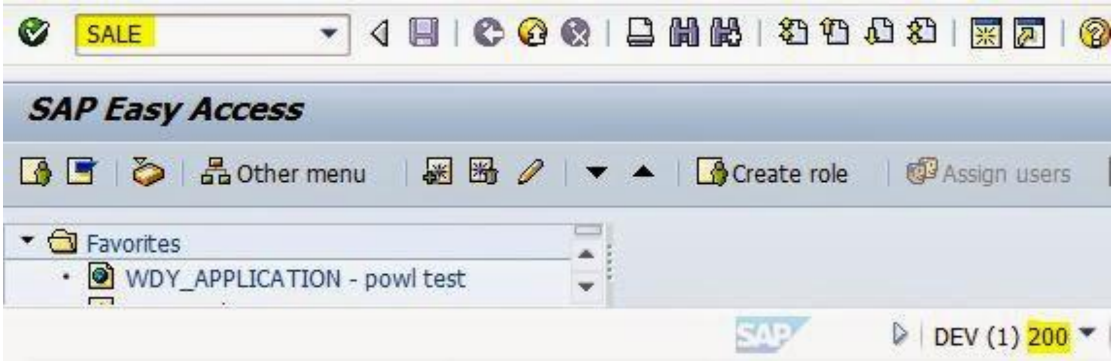
Data Browser: Table ZCNTRY_TAX_CODE: Selection Screen

| | | | |
|---|---|---|---|
| Country | | to | |
| Transaction | | to | |
| Width of Output List | 50 | | |
| Maximum No. of Hits | 10.000.000 | | |

No table entries found for specified key

DEV (1) 300

Step7. Go to Tcode- SALE.

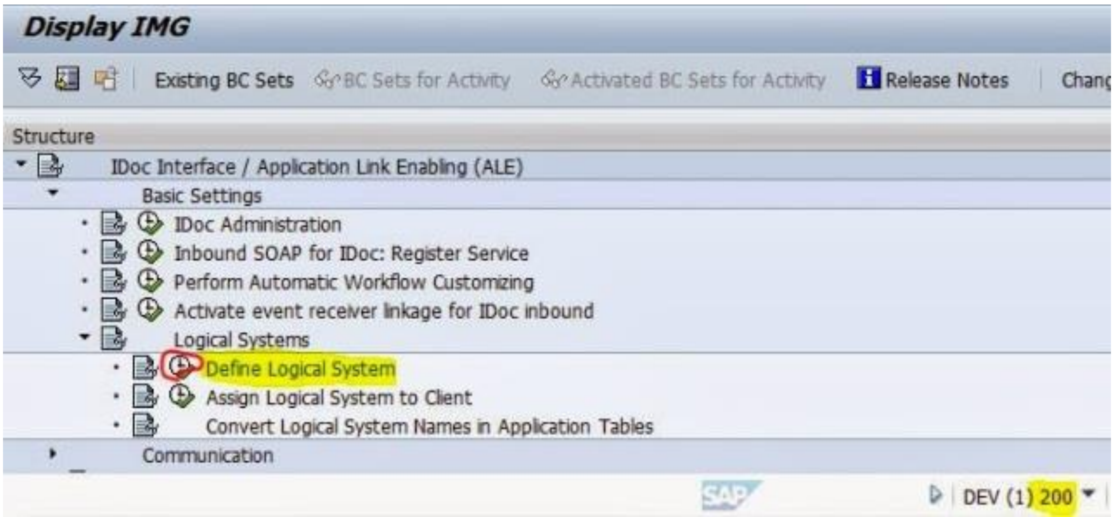


SALE

SAP Easy Access

DEV (1) 200

Step8. Navigate along the highlighted path to create logical systems.

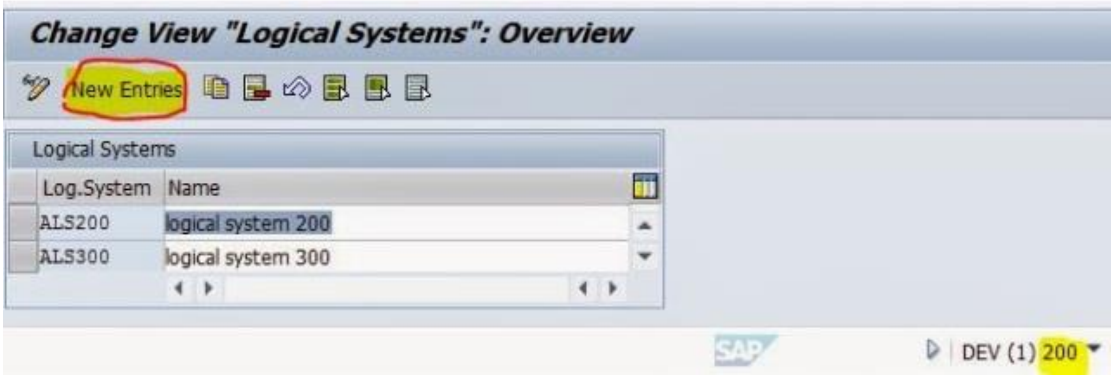


Display IMG

- 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

DEV (1) 200

Step9. Create two logical systems for the client 200 and 300.

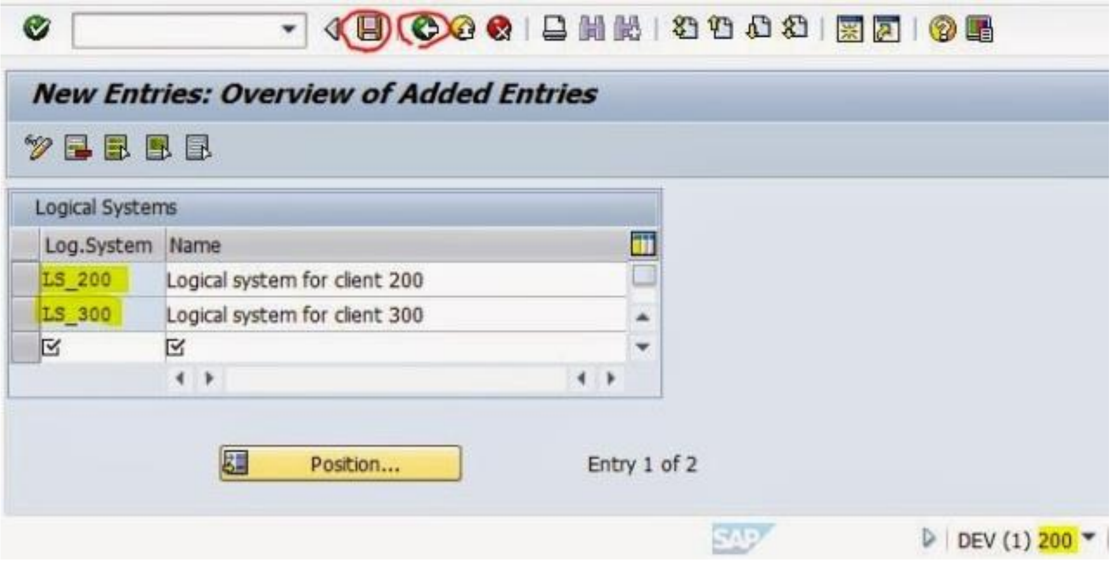


Change View "Logical Systems": Overview

Logical Systems

| Log.System | Name |
|---|---|
| ALS200 | logical system 200 |
| ALS300 | logical system 300 |

**Step10. Save it and click on the BACK button.**

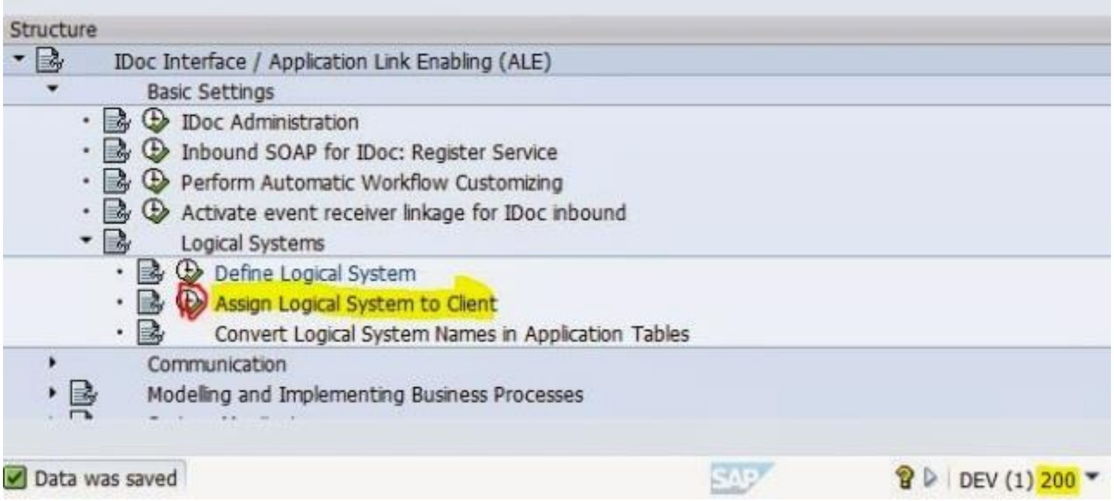


New Entries: Overview of Added Entries

Logical Systems

| Log.System | Name |
|---|---|
| LS_200 | Logical system for client 200 |
| LS_300 | Logical system for client 300 |
| ☑ | ☑ |

**Step11. Click on the highlighted line to assign the logical systems to the clients.**

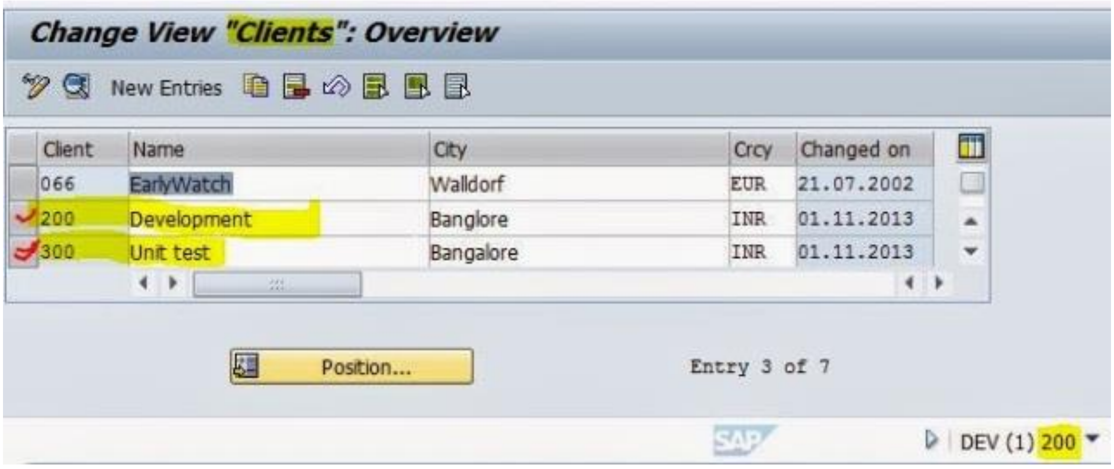


Structure

- 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
  - Modeling and Implementing Business Processes

Data was saved

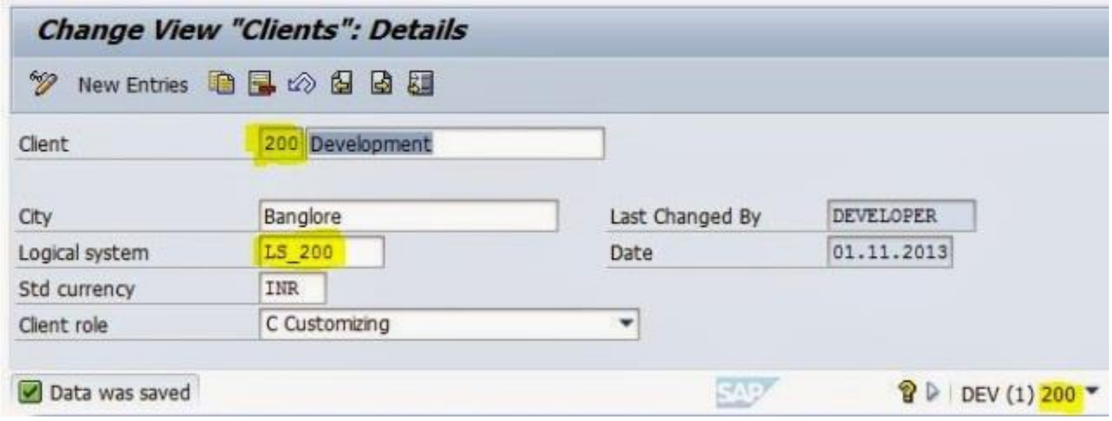
**Step12. Double  Click on the client-200.**


Change View "Clients": Overview

| Client | Name | City | Crcy | Changed on |
|---|---|---|---|---|
| 066 | EarlyWatch | Walldorf | EUR | 21.07.2002 |
| 200 | Development | Banglore | INR | 01.11.2013 |
| 300 | Unit test | Bangalore | INR | 01.11.2013 |

Entry 3 of 7

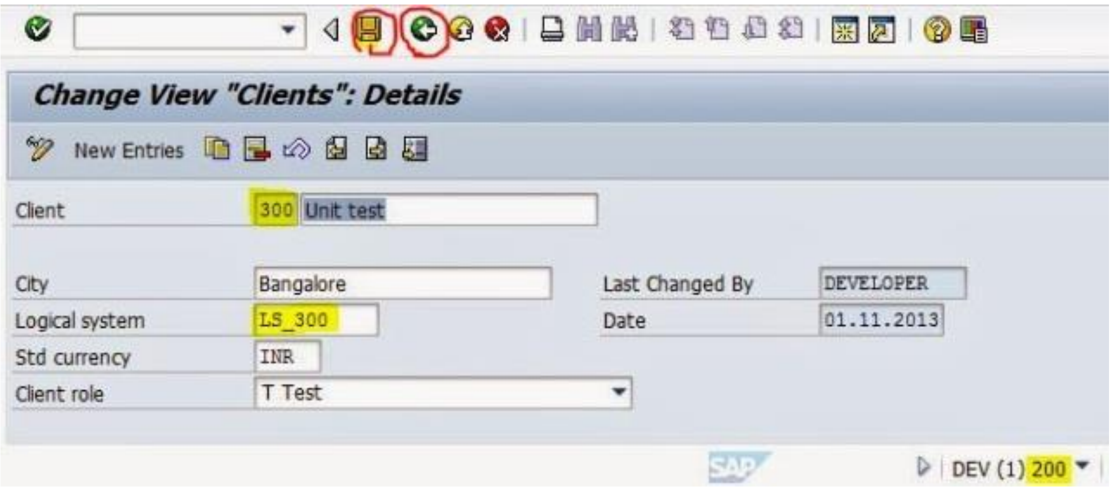
**Step13. Maintain the logical system , click on save and then click on Back button. Then double click on client 300.**


Change View "Clients": Details

| | | | |
|---|---|---|---|
| Client | 200 Development | | |
| City | Banglore | Last Changed By | DEVELOPER |
| Logical system | LS_200 | Date | 01.11.2013 |
| Std currency | INR | | |
| Client role | C Customizing | | |

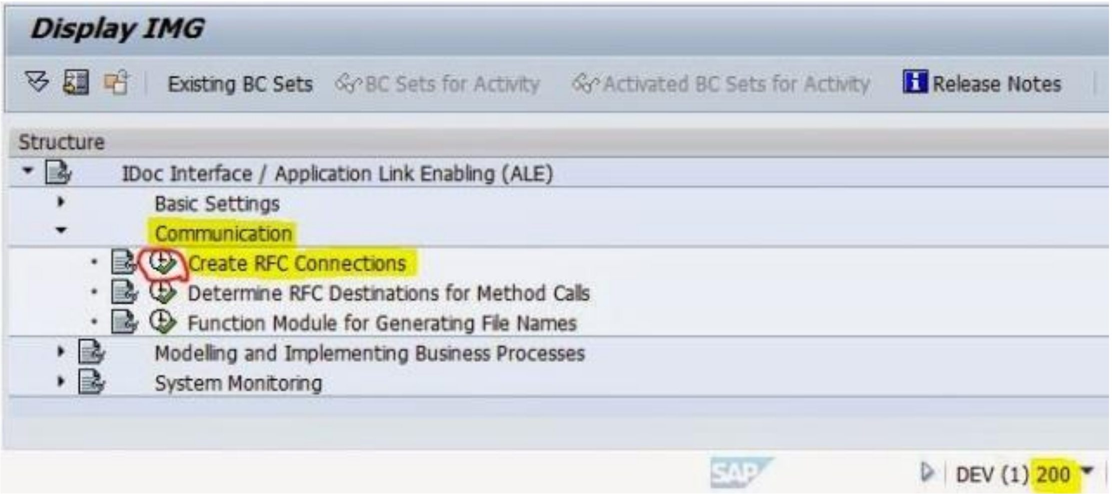
Data was saved

**Step14. Provide the logical system for the client 300, save it and at last click on Back button.**


Change View "Clients": Details

| | | | |
|---|---|---|---|
| Client | 300 Unit test | | |
| City | Bangalore | Last Changed By | DEVELOPER |
| Logical system | LS_300 | Date | 01.11.2013 |
| Std currency | INR | | |
| Client role | T Test | | |

**Step15. Click on the highlighted line to create rfc destination or go to tcode- SM59.**

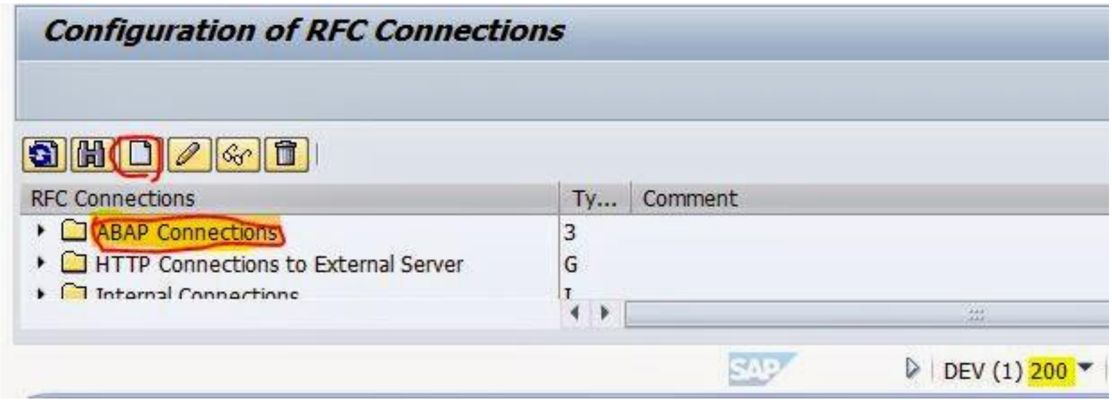


Display IMG

Structure

- IDoc Interface / Application Link Enabling (ALE)
  - Basic Settings
  - Communication
    - Create RFC Connections
    - Determine RFC Destinations for Method Calls
    - Function Module for Generating File Names
  - Modelling and Implementing Business Processes
  - System Monitoring

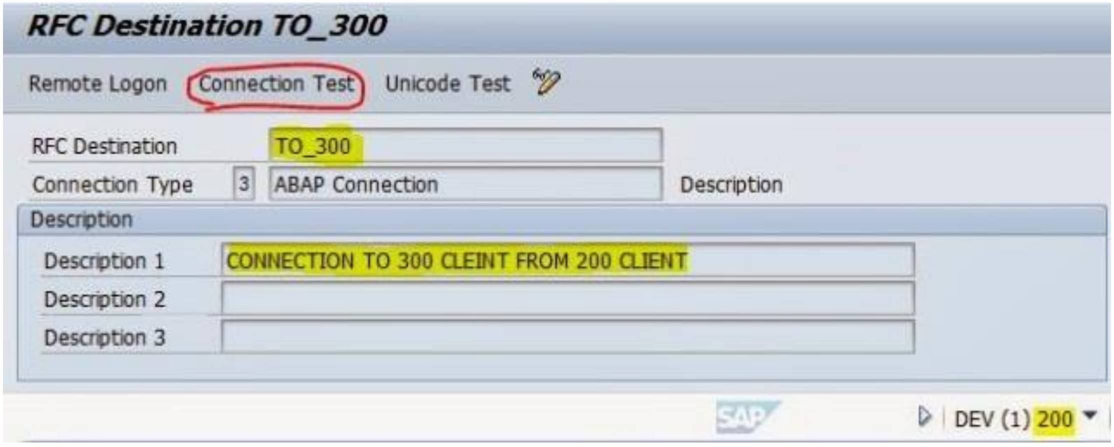
**Step16. Select the ABAP Connection type and click on Create Button.**



Configuration of RFC Connections

| RFC Connections | Ty... | Comment |
|---|---|---|
| ABAP Connections | 3 | |
| HTTP Connections to External Server | G | |
| Internal Connections | I | |

**Step17. Provide the details and save it. Click on the connection the button.**



RFC Destination TO_300

RFC Destination: TO_300

Connection Type: 3 ABAP Connection

Description

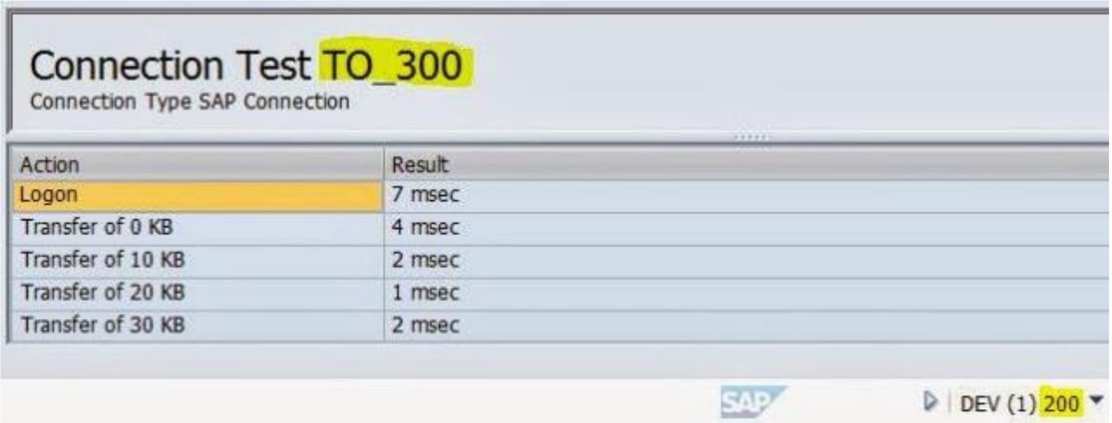
Description

Description 1: CONNECTION TO 300 CLEINT FROM 200 CLIENT

Description 2:

Description 3:

**Step18. The connection works fine.**



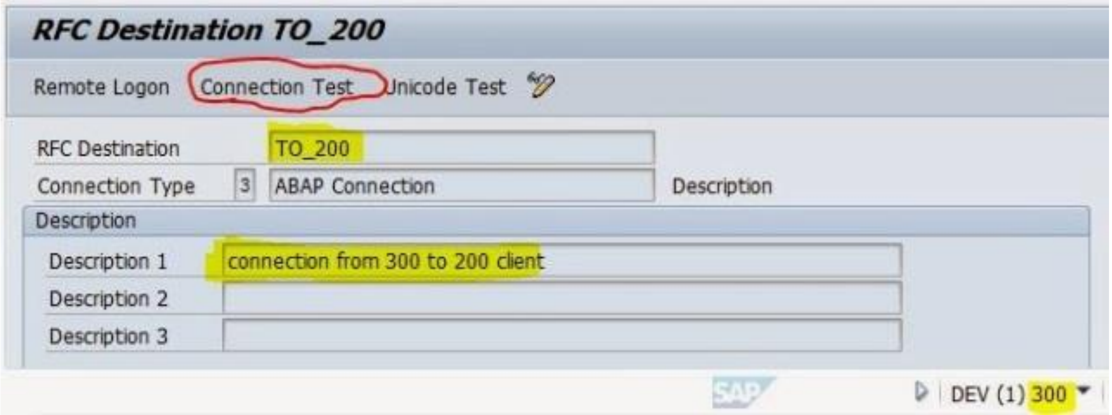
Connection Test TO_300

Connection Type SAP Connection

| Action | Result |
|---|---|
| Logon | 7 msec |
| Transfer of 0 KB | 4 msec |
| Transfer of 10 KB | 2 msec |
| Transfer of 20 KB | 1 msec |
| Transfer of 30 KB | 2 msec |

DEV (1) 200

**Step19. Now logon to the 300 client and create a rfc destination. Provide the details, save it and at last click on Connection Test button.**



RFC Destination TO_200

RFC Destination: TO_200

Connection Type: 3 ABAP Connection

Description

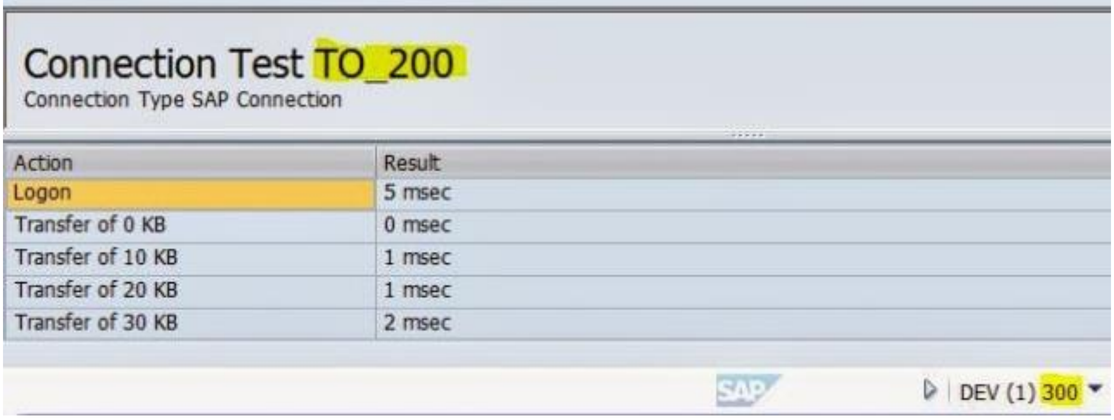
Description 1: connection from 300 to 200 client

Description 2:

Description 3:

DEV (1) 300

**Step20. The RFC connection works fine.**



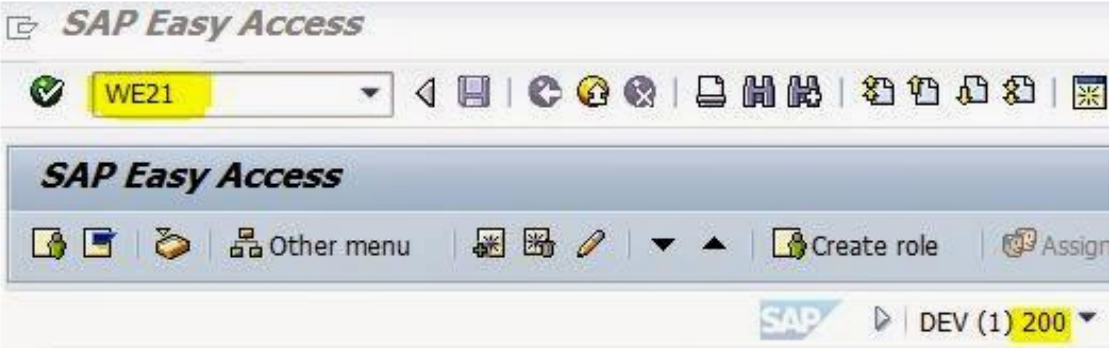
Connection Test TO_200

Connection Type SAP Connection

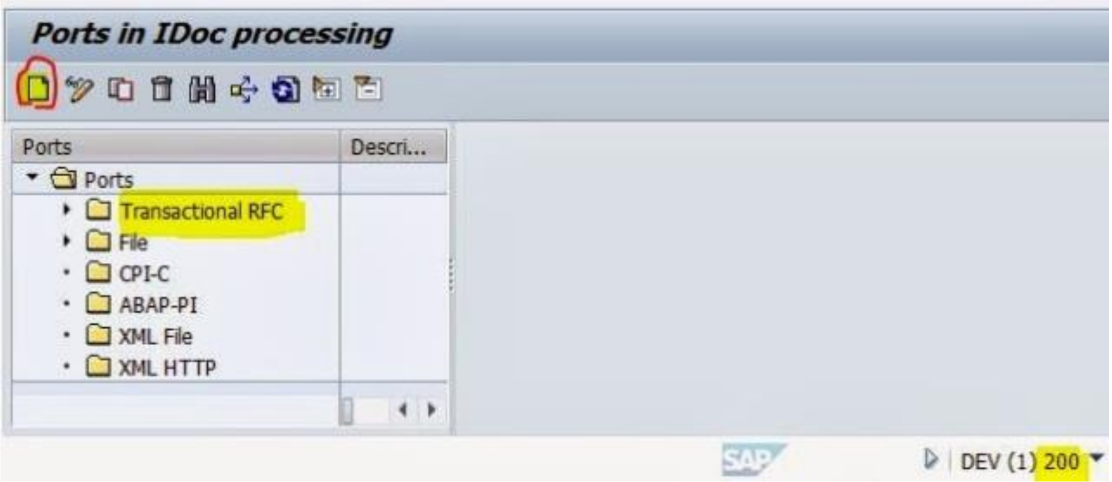
| Action | Result |
|---|---|
| Logon | 5 msec |
| Transfer of 0 KB | 0 msec |
| Transfer of 10 KB | 1 msec |
| Transfer of 20 KB | 1 msec |
| Transfer of 30 KB | 2 msec |

DEV (1) 300

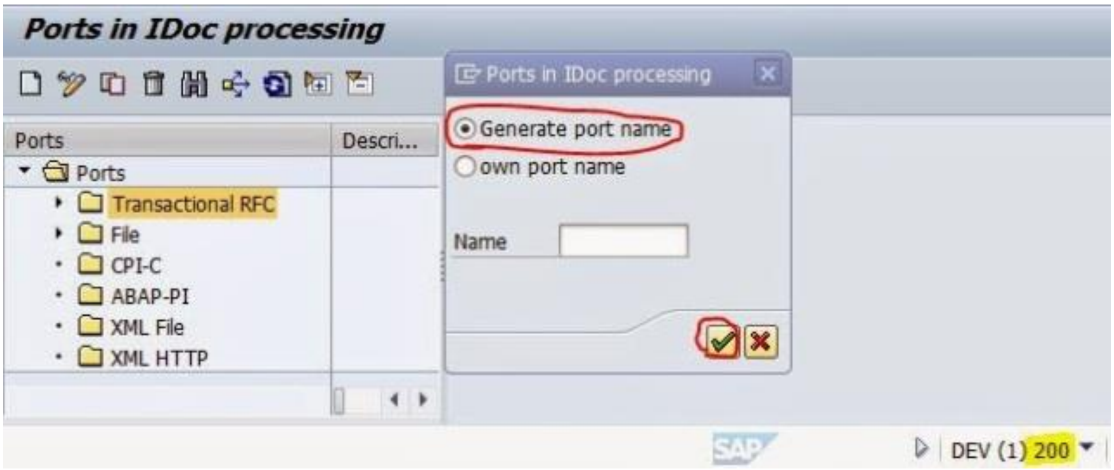
**Step21. Now go to Tcode- WE21 to create port.**



**Step22. Select the transactional RFC line and click on the create button.**



**Step23. Select the first line and click on continue button.**

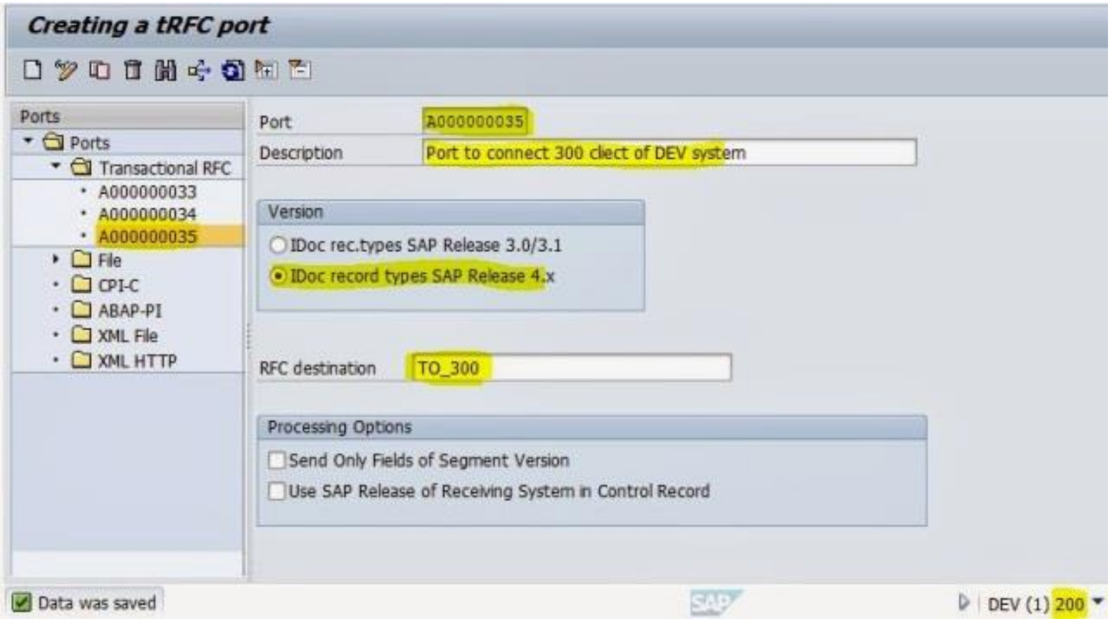


● Generate port name

○ own port name

Name

**Step24. The system automatically creates a port number, provide the description and the RFC destination name and save it.**



Creating a tRFC port

Port: A000000035

Description: Port to connect 300 clect of DEV system

Version

○ IDoc rec.types SAP Release 3.0/3.1

● IDoc record types SAP Release 4.x

RFC destination: TO_300

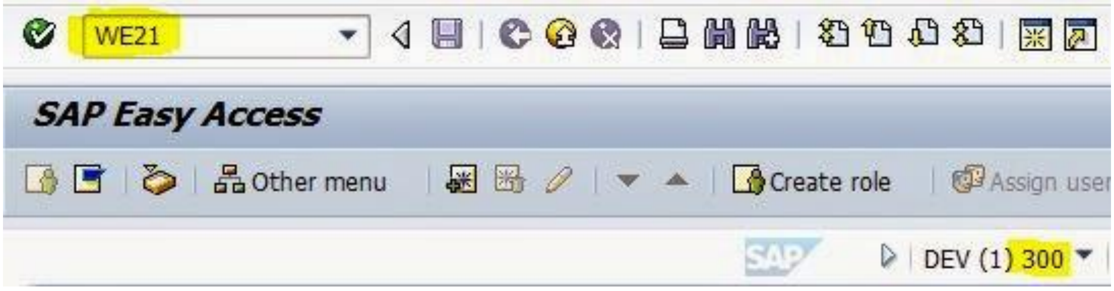
Processing Options

□ Send Only Fields of Segment Version

□ Use SAP Release of Receiving System in Control Record

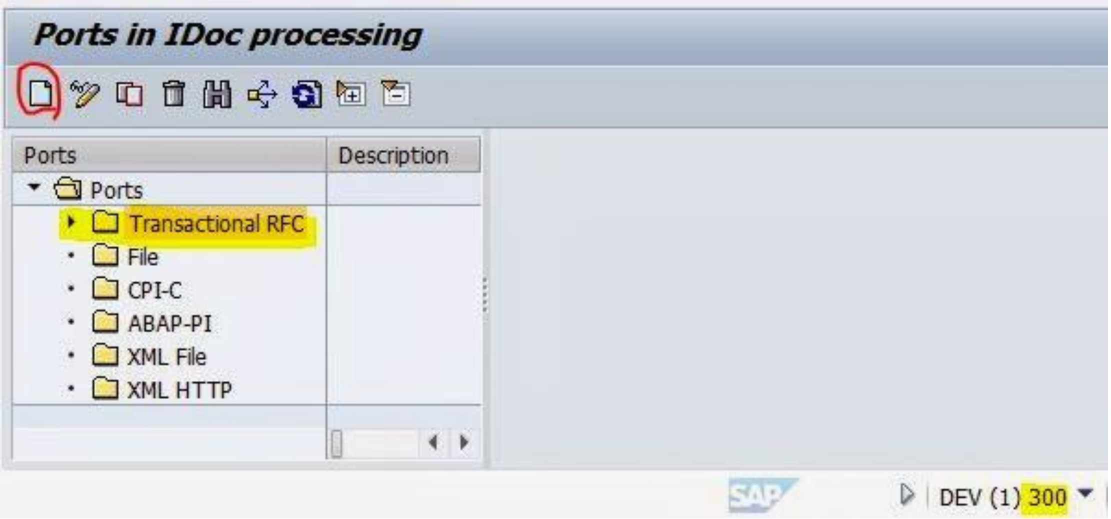
Data was saved

**Step25. Now go to client 300. Go to Tcode- WE21.**

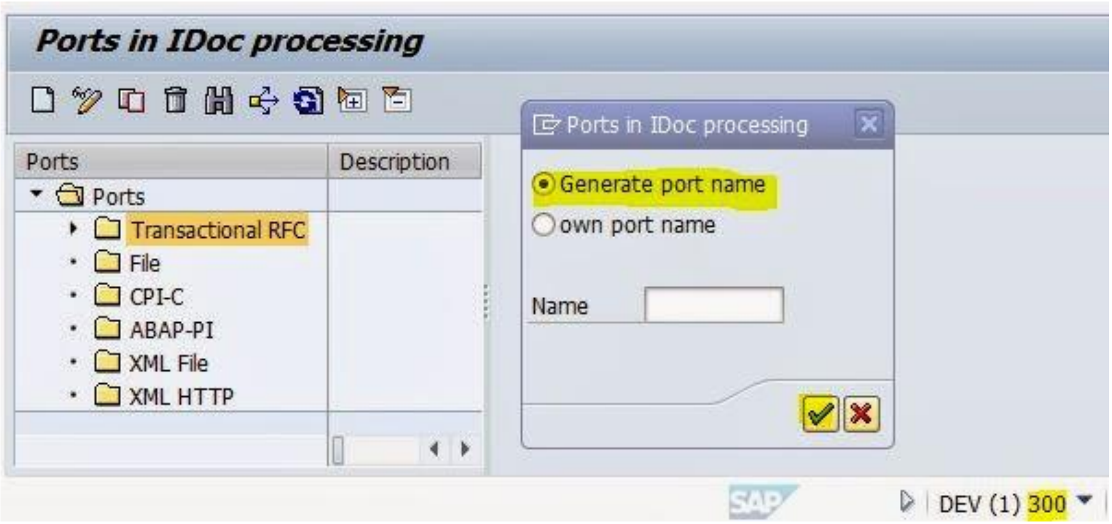


WE21

**Step26. Select the transactional RFC line and click on the create button.**



**Step27. Select the first line and click on continue button.**



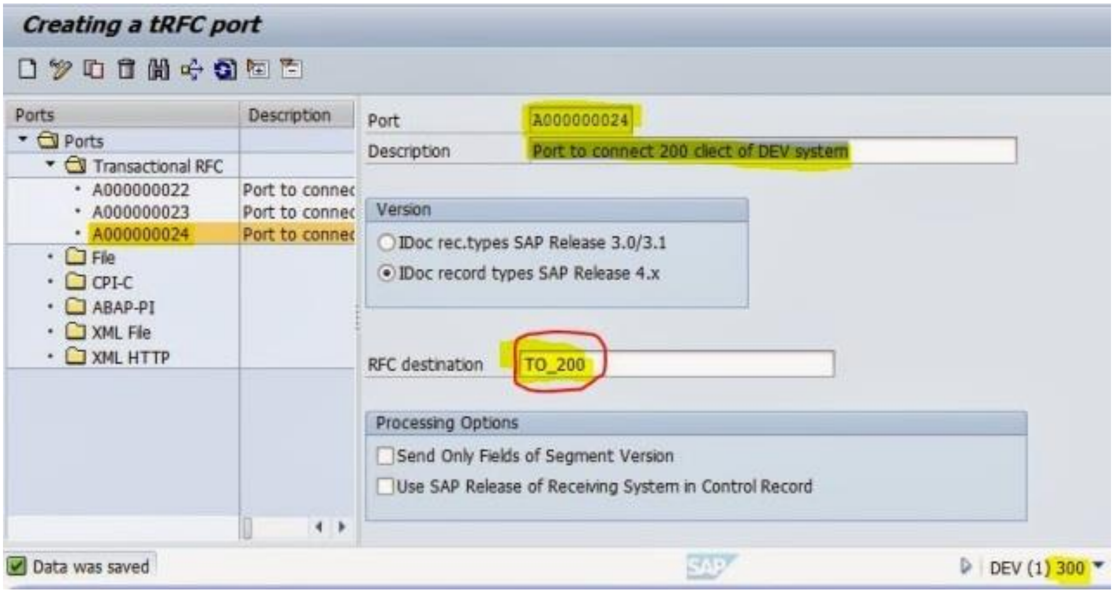
Ports in IDoc processing

● Generate port name

○ own port name

Name

**Step28. The system automatically creates a port number, provide the description and the RFC destination name and save it.**



| Ports | Description |
|---|---|
| A000000022 | Port to connec |
| A000000023 | Port to connec |
| A000000024 | Port to connec |

Port A000000024

Description Port to connect 200 cliect of DEV system

Version

○ IDoc rec.types SAP Release 3.0/3.1

● IDoc record types SAP Release 4.x

RFC destination TO_200

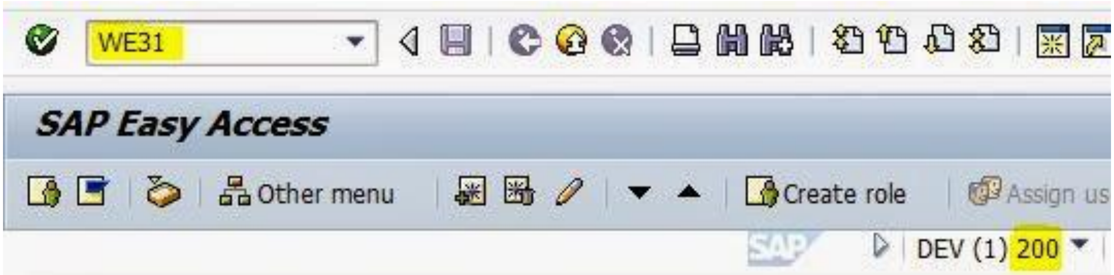
Processing Options

□ Send Only Fields of Segment Version

□ Use SAP Release of Receiving System in Control Record

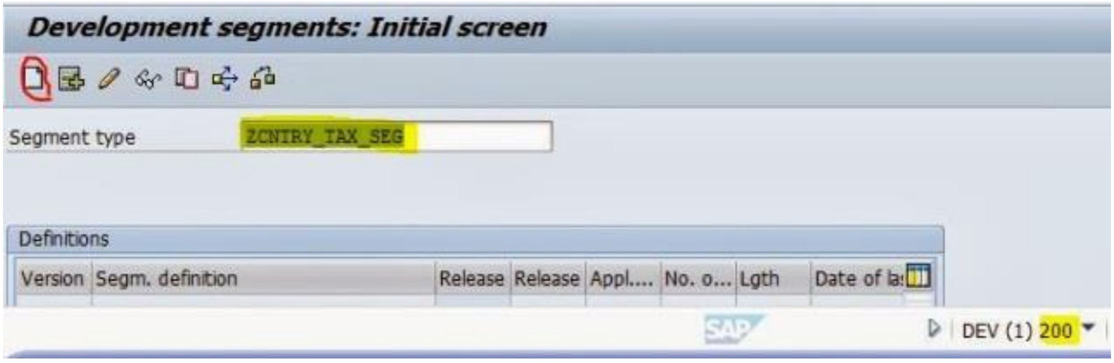
☑ Data was saved

**Step29. Now in client-200 , go to Tcode- WE31.**



WE31

**Step30. Provide the segment name and click on the create button.**



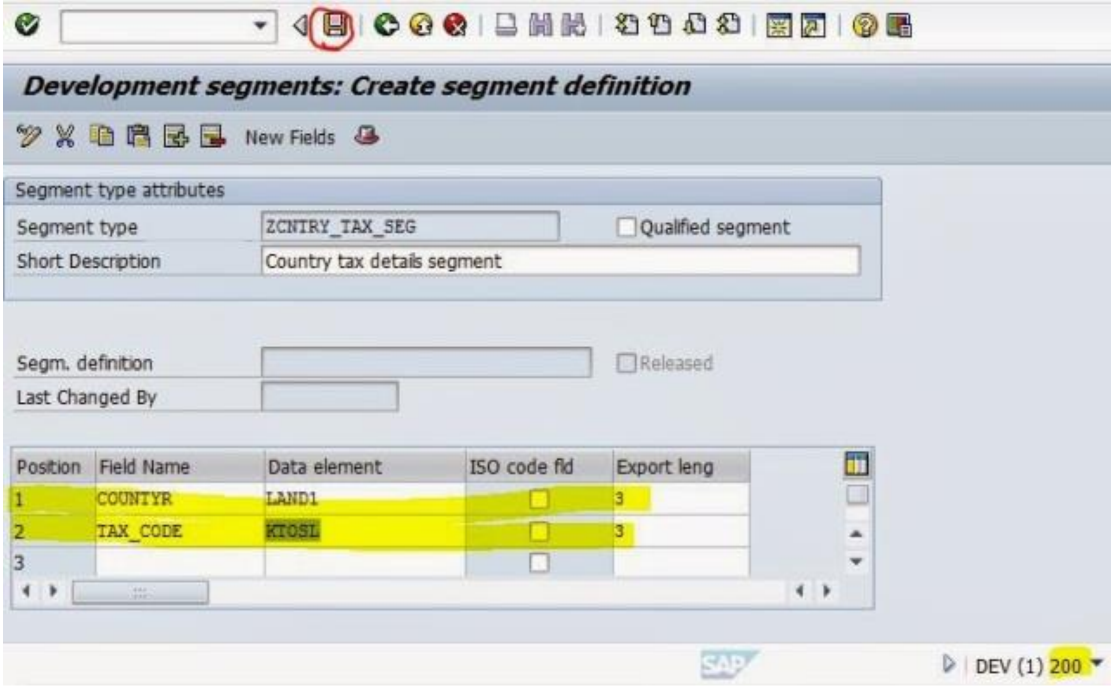
**Development segments: Initial screen**

Segment type: ZCNTRY_TAX_SEG

Definitions

| Version | Segm. definition | Release | Release | Appl.... | No. o... | Lgth | Date of la: |
|---|---|---|---|---|---|---|---|

**Step31. Provide the short text, the fields of the table with data element and click on the save button.**



**Development segments: Create segment definition**

Segment type attributes

Segment type: ZCNTRY_TAX_SEG □ Qualified segment

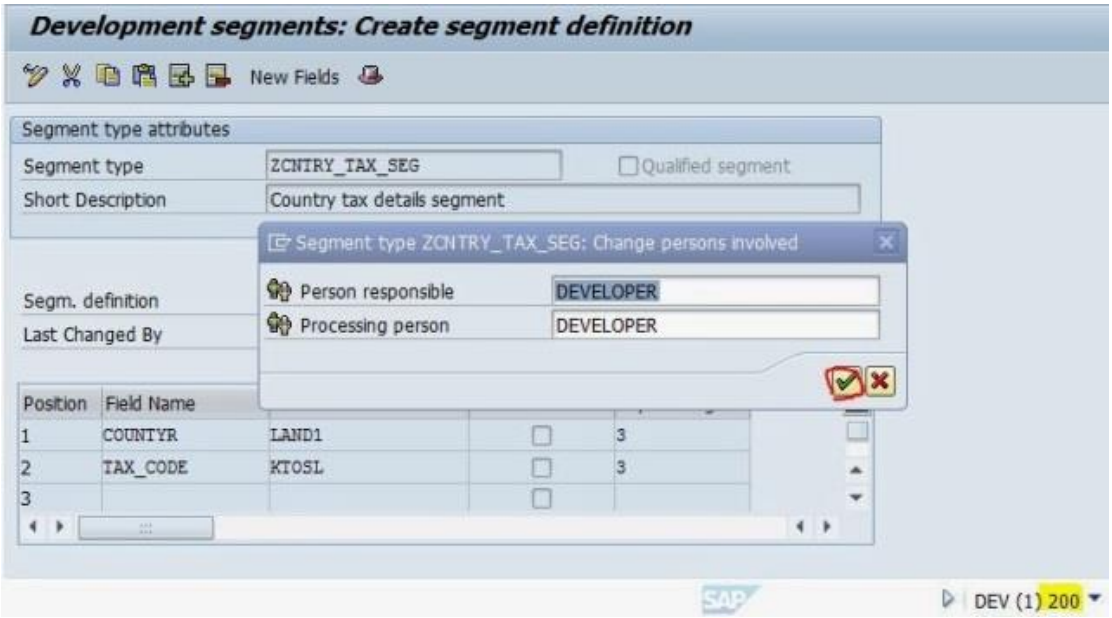
Short Description: Country tax details segment

Segm. definition: □ Released

Last Changed By:

| Position | Field Name | Data element | ISO code fld | Export leng |
|---|---|---|---|---|
| 1 | COUNTYR | LAND1 | □ | 3 |
| 2 | TAX_CODE | KTOSL | □ | 3 |
| 3 | | | □ | |

**Step32. Click the continue button.**



**Development segments: Create segment definition**

Segment type attributes

Segment type: ZCNTRY_TAX_SEG □ Qualified segment

Short Description: Country tax details segment

Segm. definition:

Last Changed By:

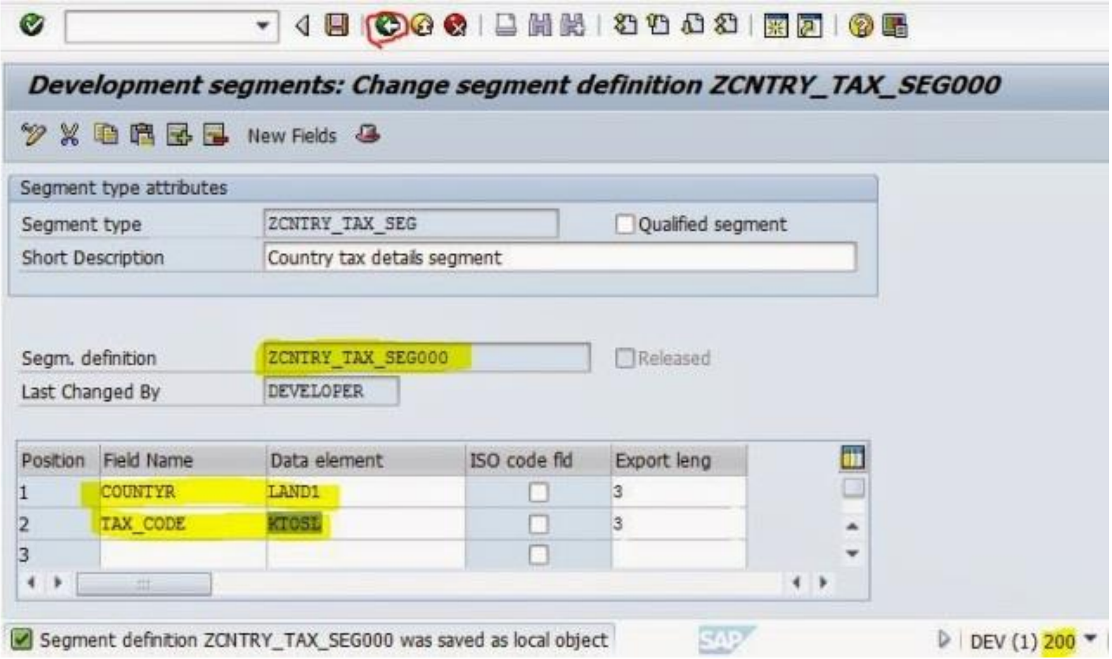
Segment type ZCNTRY_TAX_SEG: Change persons involved

Person responsible: DEVELOPER

Processing person: DEVELOPER

| Position | Field Name | | | |
|---|---|---|---|---|
| 1 | COUNTYR | LAND1 | □ | 3 |
| 2 | TAX_CODE | KTOSL | □ | 3 |
| 3 | | | □ | |

**Step33. Save it and click on Back button.**



Development segments: Change segment definition ZCNTRY_TAX_SEG000

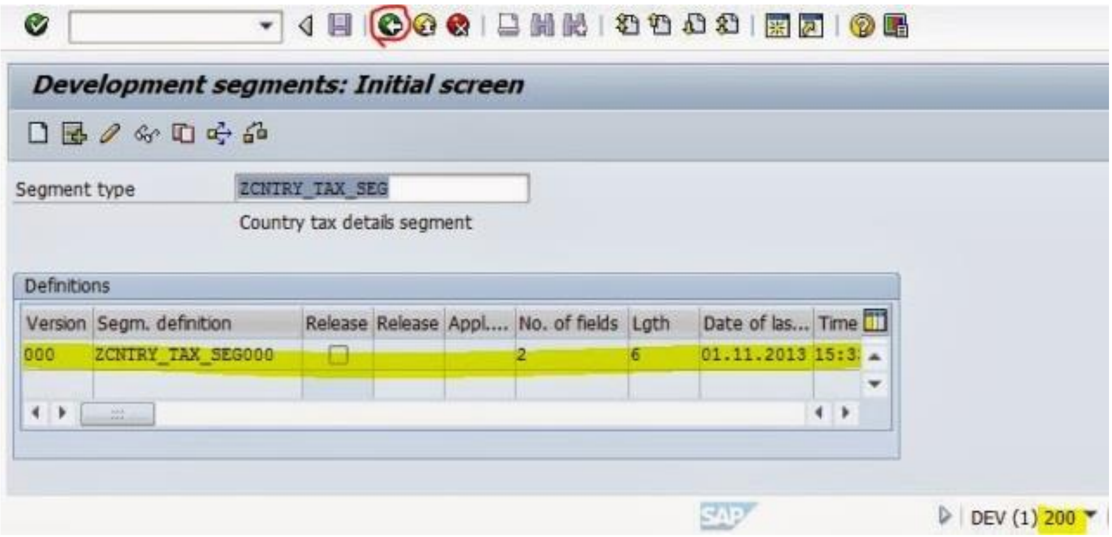
Segment type attributes

| | | |
|---|---|---|
| Segment type | ZCNTRY_TAX_SEG | □ Qualified segment |
| Short Description | Country tax details segment | |
| Segm. definition | ZCNTRY_TAX_SEG000 | □ Released |
| Last Changed By | DEVELOPER | |

| Position | Field Name | Data element | ISO code fld | Export leng |
|---|---|---|---|---|
| 1 | COUNTYR | LAND1 | □ | 3 |
| 2 | TAX_CODE | KTOSL | □ | 3 |
| 3 | | | □ | |

Segment definition ZCNTRY_TAX_SEG000 was saved as local object

**Step34. Again click on Back button.**



Development segments: Initial screen

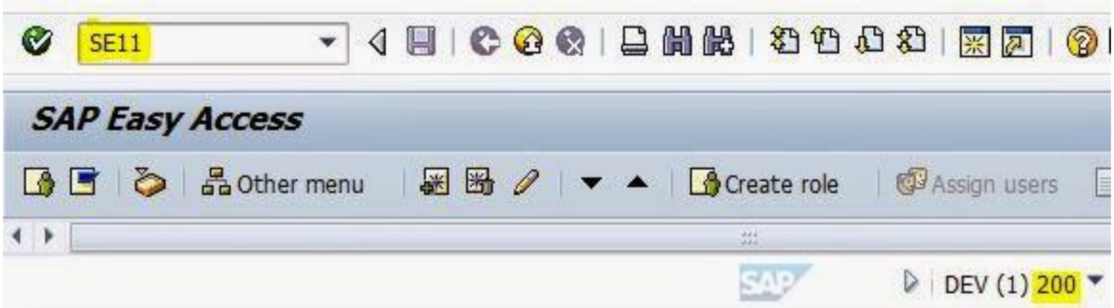
Segment type ZCNTRY_TAX_SEG

Country tax details segment

Definitions

| Version | Segm. definition | Release | Release | Appl.... | No. of fields | Lgth | Date of las... | Time |
|---|---|---|---|---|---|---|---|---|
| 000 | ZCNTRY_TAX_SEG000 | □ | | | 2 | 6 | 01.11.2013 | 15:3: |

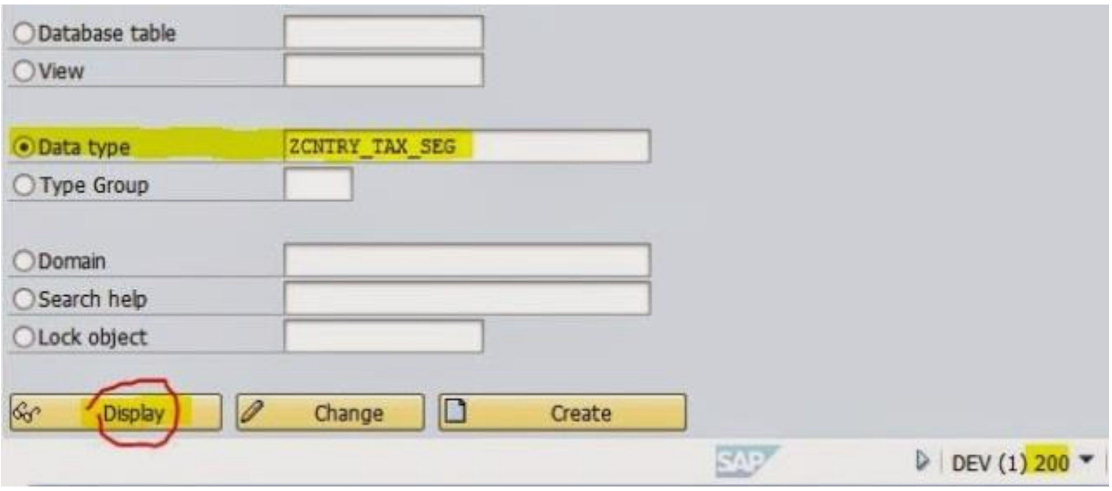
**Step35. Now go to Tcode- SE11.**



SE11

SAP Easy Access

**Step36.  Provide the segment name and click on display button.**



○ Database table

○ View

● Data type ZCNTRY_TAX_SEG

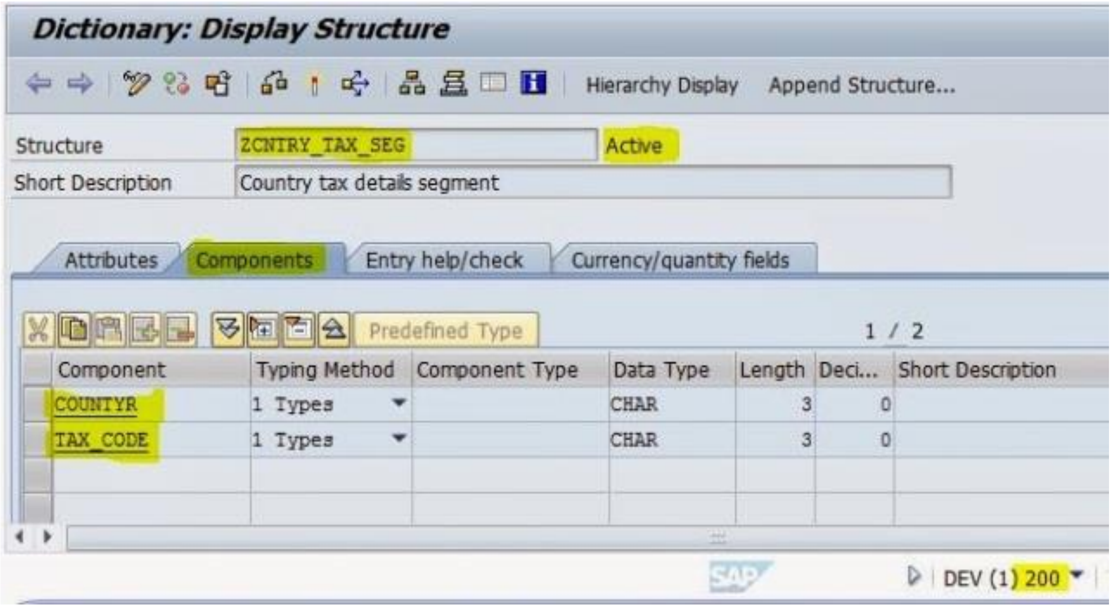
○ Type Group

○ Domain

○ Search help

○ Lock object

**Step37. The system automatically creates a structure with the name of the segment.**



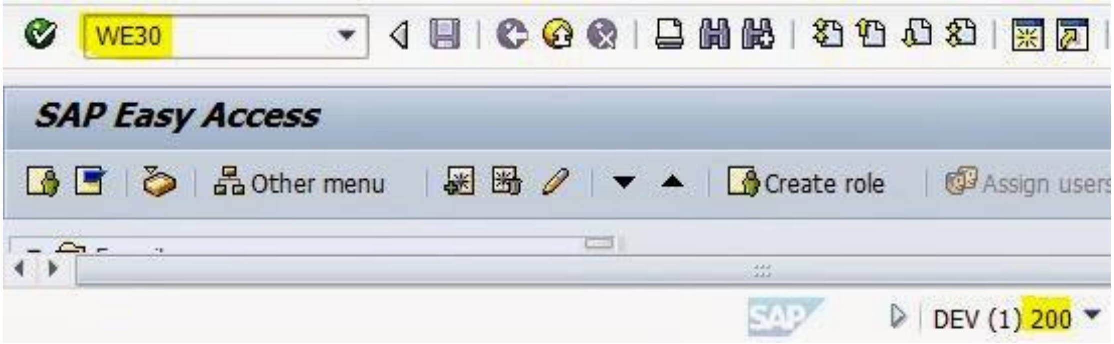
Dictionary: Display Structure

Structure ZCNTRY_TAX_SEG Active

Short Description Country tax details segment

| Component | Typing Method | Component Type | Data Type | Length | Deci... | Short Description |
|---|---|---|---|---|---|---|
| COUNTYR | 1 Types | | CHAR | 3 | 0 | |
| TAX_CODE | 1 Types | | CHAR | 3 | 0 | |

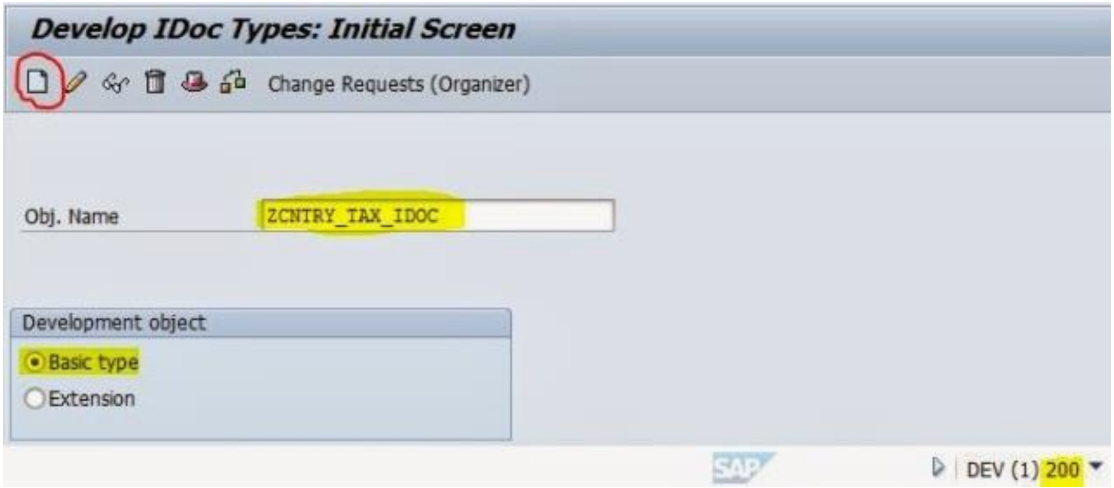
**Step38. Now go to Tcode- WE30 to asisgn the segment the the IDOC type.**



WE30

SAP Easy Access

**Step39. Provide the IDOC type name and click on Create button.**



Develop IDoc Types: Initial Screen

Change Requests (Organizer)

Obj. Name: ZCNTRY_TAX_IDOC

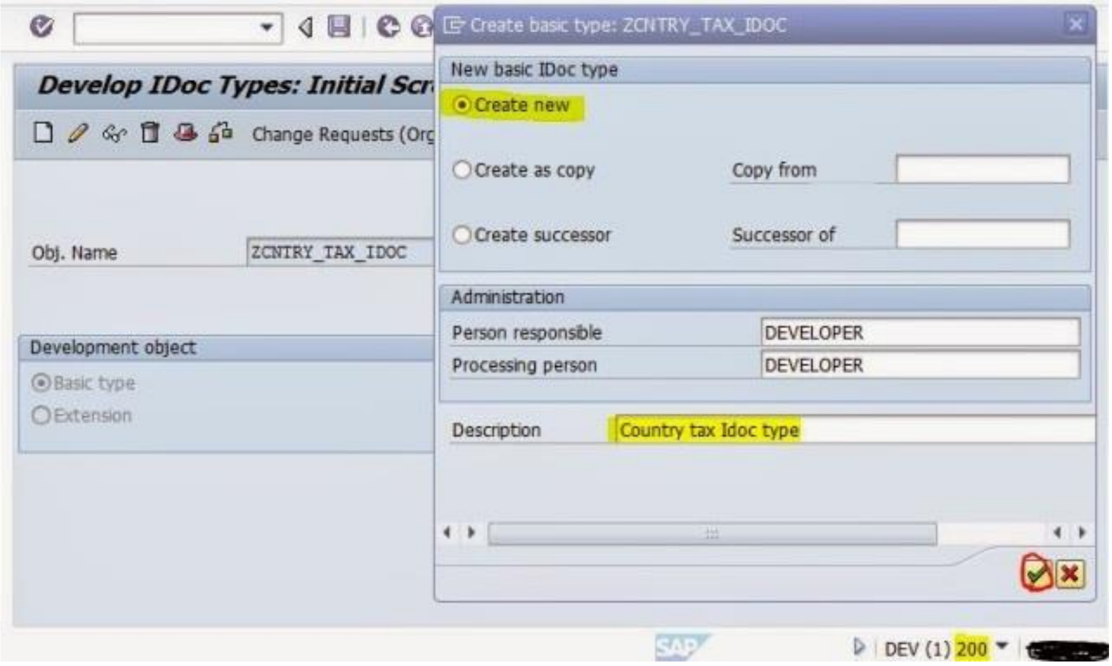
Development object

● Basic type

○ Extension

DEV (1) 200

**Step40. Select the create New radio button and provide the description and click on continue button.**



Develop IDoc Types: Initial Scr

Change Requests (Org

Obj. Name: ZCNTRY_TAX_IDOC

Development object

● Basic type

○ Extension

Create basic type: ZCNTRY_TAX_IDOC

New basic IDoc type

● Create new

○ Create as copy — Copy from

○ Create successor — Successor of

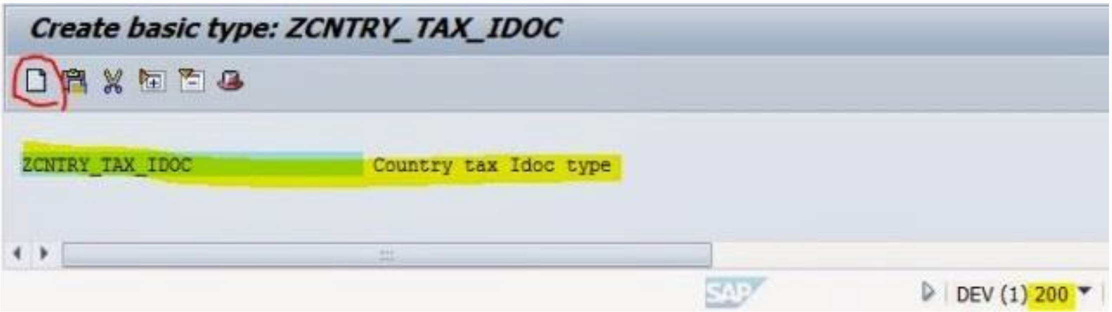
Administration

| | |
|---|---|
| Person responsible | DEVELOPER |
| Processing person | DEVELOPER |

Description: Country tax Idoc type

DEV (1) 200 [redacted]

**Step41. Select the Idoc type and click on the Create button.**

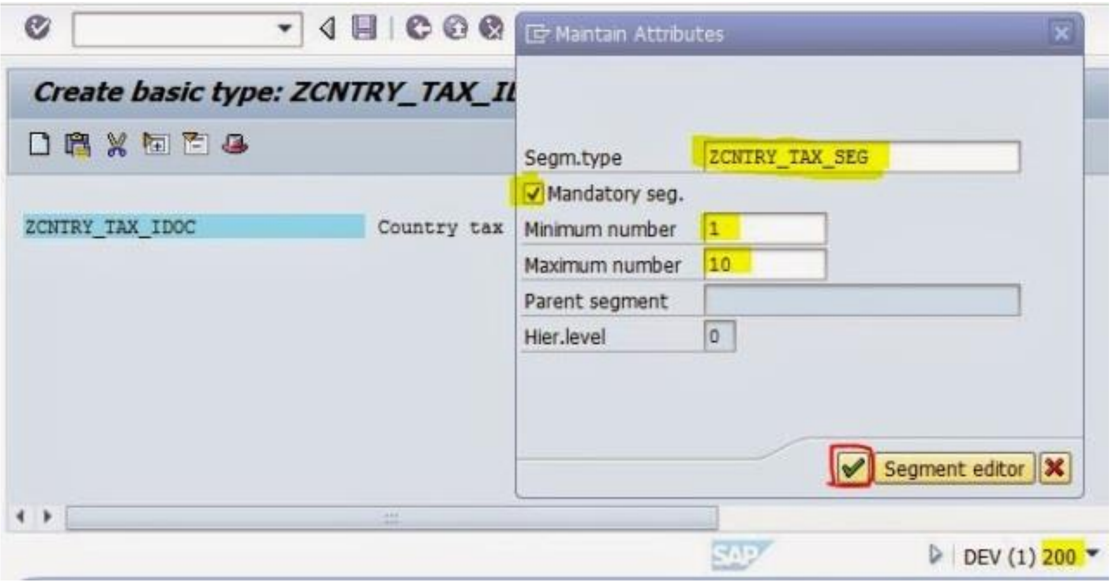


Create basic type: ZCNTRY_TAX_IDOC

ZCNTRY_TAX_IDOC Country tax Idoc type

DEV (1) 200

**Step42. Provide the segment name created above. Select it as Mandatory segment and at last click on the Continue button.**



Create basic type: ZCNTRY_TAX_IL

ZCNTRY_TAX_IDOC Country tax

Maintain Attributes

Segm.type: ZCNTRY_TAX_SEG

☑ Mandatory seg.

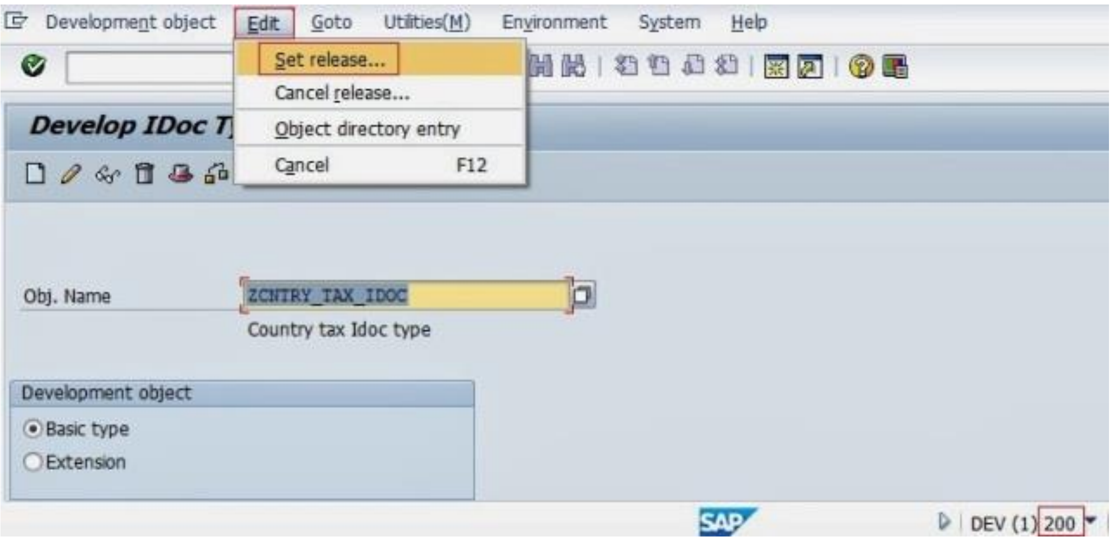
Minimum number: 1

Maximum number: 10

Parent segment:

Hier.level: 0

**Step43. Now click on the Edit menu and Click Set Release .**



Develop IDoc T

Obj. Name: ZCNTRY_TAX_IDOC

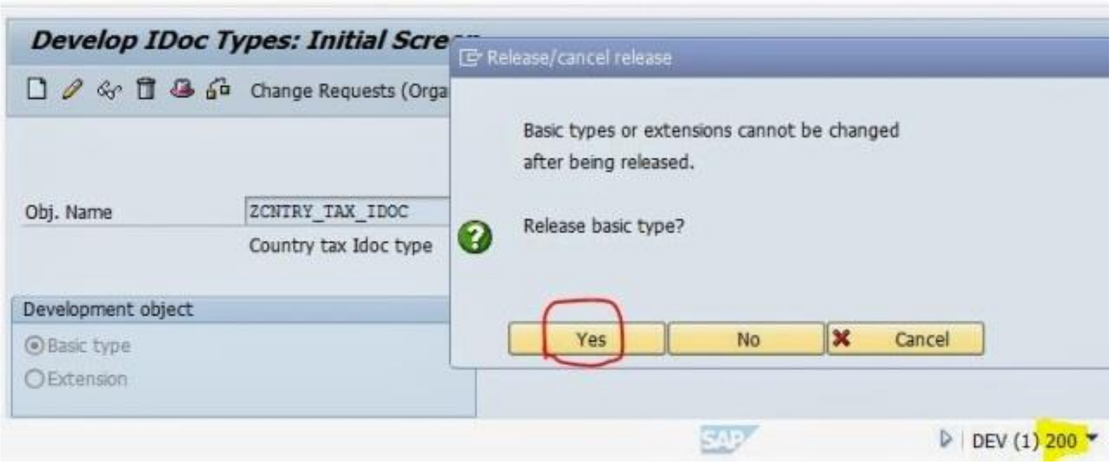
Country tax Idoc type

Development object

● Basic type

○ Extension

**Step44. Click on Yes button to continue.**



Develop IDoc Types: Initial Scre

Change Requests (Orga

Obj. Name: ZCNTRY_TAX_IDOC

Country tax Idoc type

Development object

● Basic type

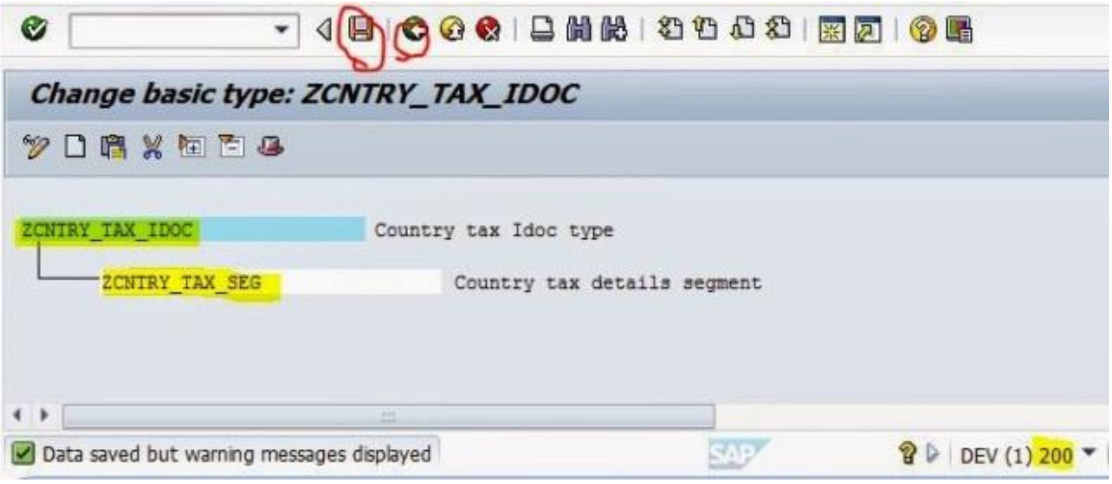
○ Extension

Release/cancel release

Basic types or extensions cannot be changed after being released.

Release basic type?

**Step45. Save it and click on Back button,**



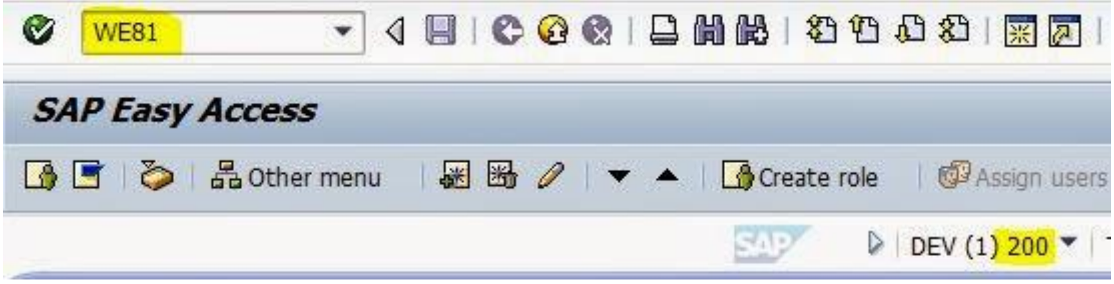
Change basic type: ZCNTRY_TAX_IDOC

ZCNTRY_TAX_IDOC Country tax Idoc type

ZCNTRY_TAX_SEG Country tax details segment

Data saved but warning messages displayed

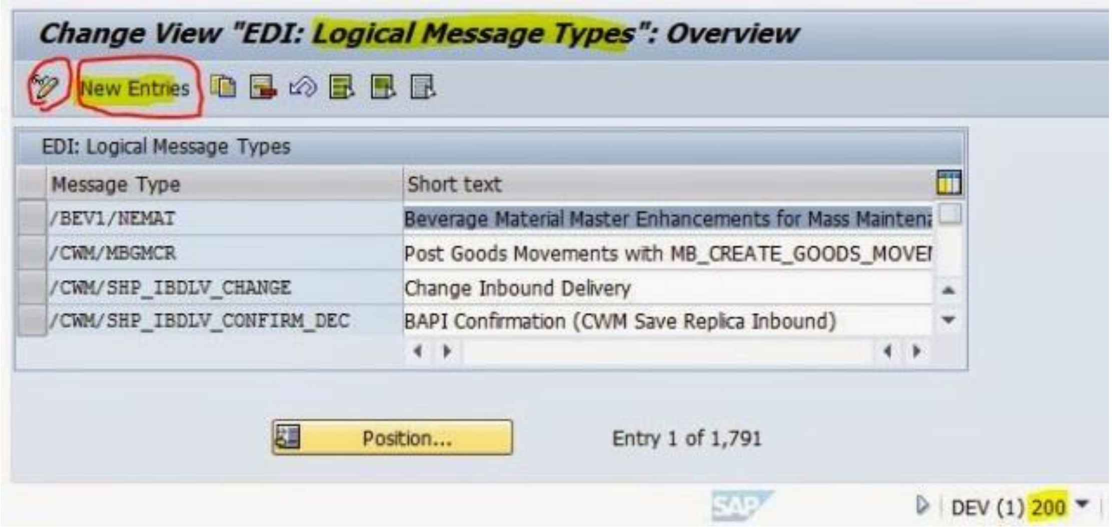
**Step46. Now go to Tcode- SE81 to create a message type.**



WE81

SAP Easy Access

**Step47. Click on Edit and the New Entries button.**



Change View "EDI: Logical Message Types": Overview

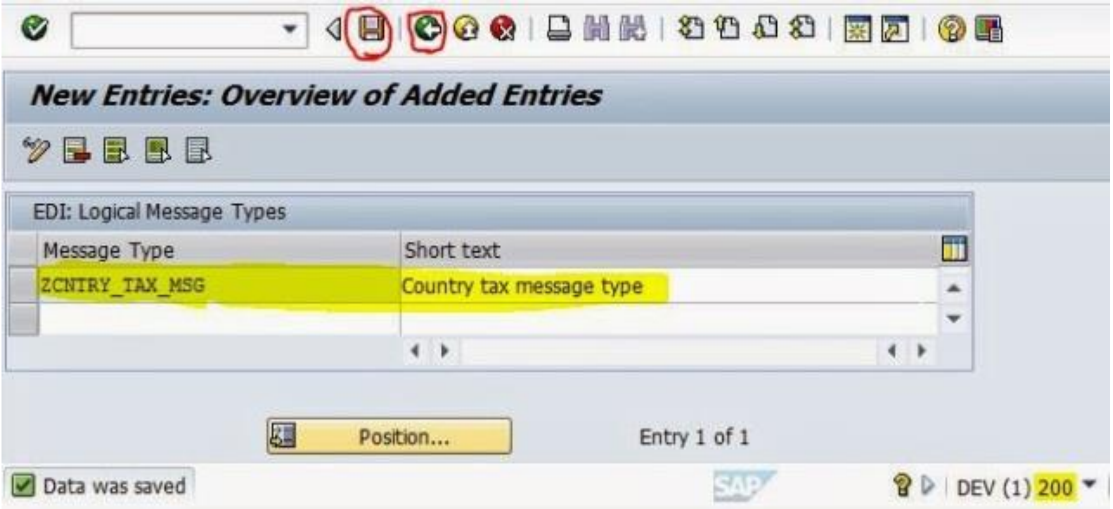
New Entries

EDI: Logical Message Types

| Message Type | Short text |
|---|---|
| /BEV1/NEMAT | Beverage Material Master Enhancements for Mass Maintena |
| /CWM/MBGMCR | Post Goods Movements with MB_CREATE_GOODS_MOVEI |
| /CWM/SHP_IBDLV_CHANGE | Change Inbound Delivery |
| /CWM/SHP_IBDLV_CONFIRM_DEC | BAPI Confirmation (CWM Save Replica Inbound) |

Position... Entry 1 of 1,791

**Step48. Provide a message type and short text and save it. At last click on Back button.**



New Entries: Overview of Added Entries

EDI: Logical Message Types

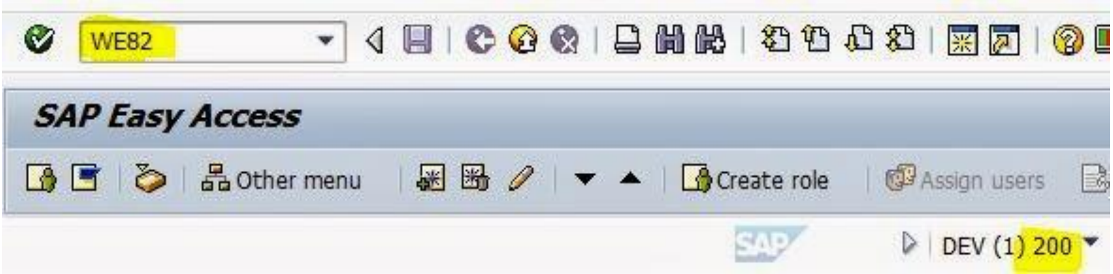
| Message Type | Short text |
|---|---|
| ZCNTRY_TAX_MSG | Country tax message type |

Entry 1 of 1

Data was saved

DEV (1) 200

**Step49. Now go to Tcode- WE82 to assign message type to Idoc.**

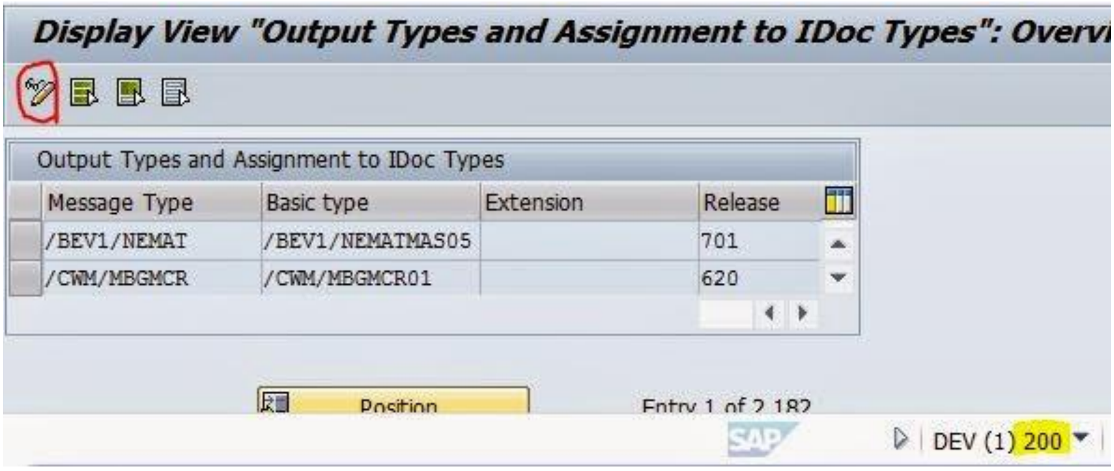


WE82

SAP Easy Access

DEV (1) 200

**Step50. Click on the Edit button.**



Display View "Output Types and Assignment to IDoc Types": Overvi

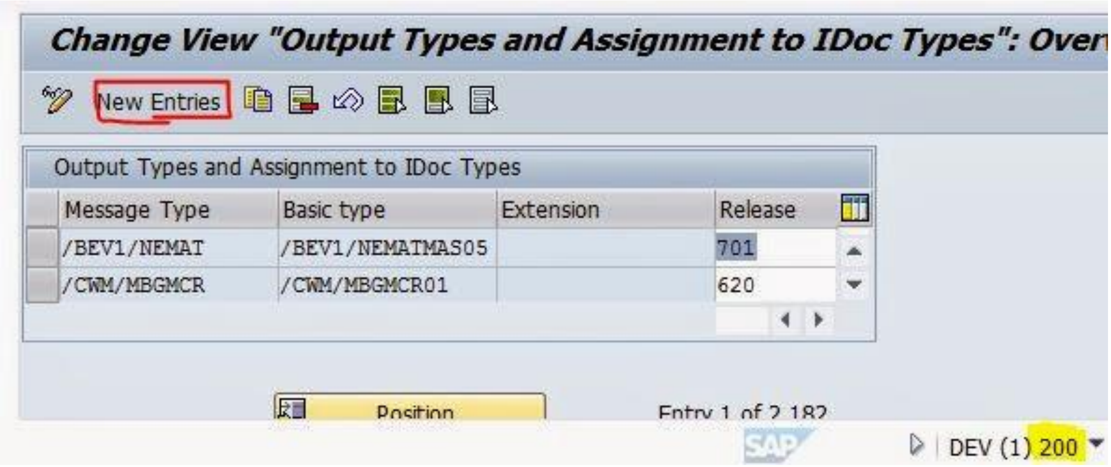
Output Types and Assignment to IDoc Types

| Message Type | Basic type | Extension | Release |
|---|---|---|---|
| /BEV1/NEMAT | /BEV1/NEMATMAS05 | | 701 |
| /CWM/MBGMCR | /CWM/MBGMCR01 | | 620 |

Entry 1 of 2,182

DEV (1) 200

**Step51. Click on New Entries button.**

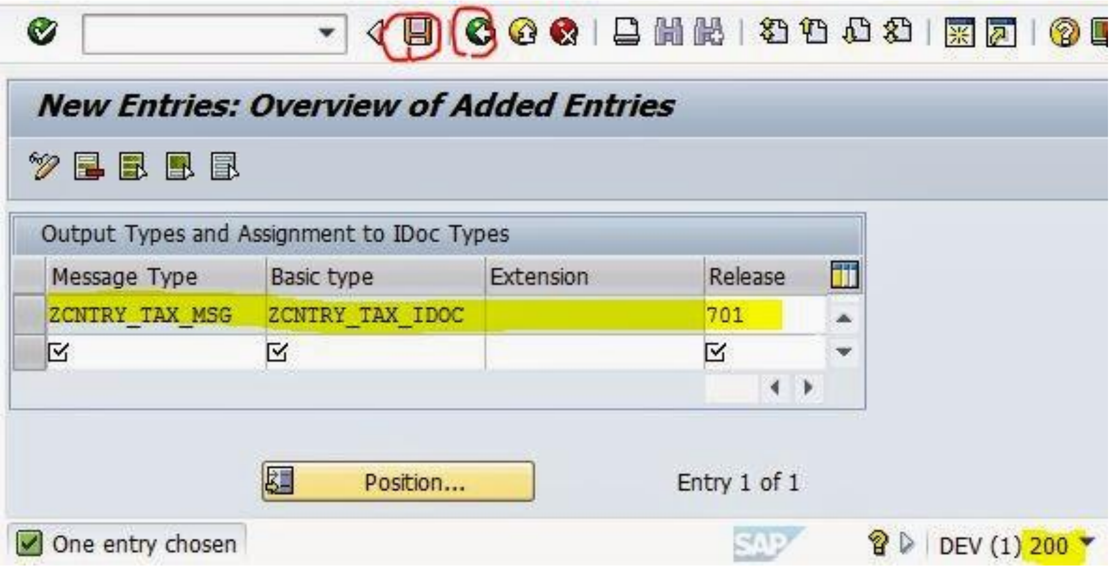


Change View "Output Types and Assignment to IDoc Types": Over

Output Types and Assignment to IDoc Types

| Message Type | Basic type | Extension | Release |
|---|---|---|---|
| /BEV1/NEMAT | /BEV1/NEMATMAS05 | | 701 |
| /CWM/MBGMCR | /CWM/MBGMCR01 | | 620 |

**Step52. Provide the above created message type, Idoc name and release and Save it. At last click on Back button.**



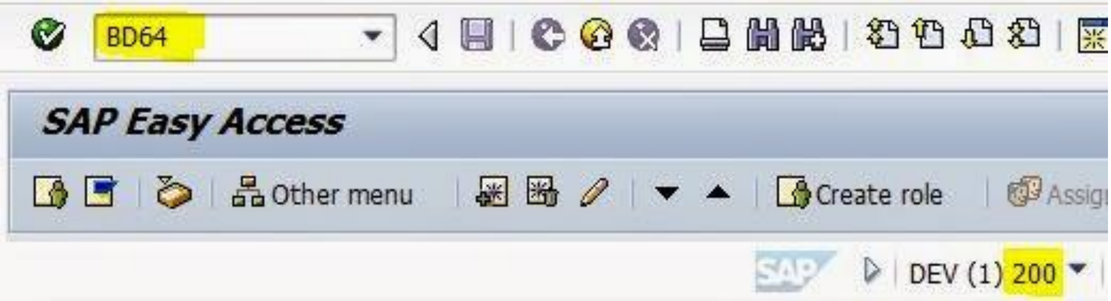
New Entries: Overview of Added Entries

Output Types and Assignment to IDoc Types

| Message Type | Basic type | Extension | Release |
|---|---|---|---|
| ZCNTRY_TAX_MSG | ZCNTRY_TAX_IDOC | | 701 |
| ☑ | ☑ | | ☑ |

One entry chosen

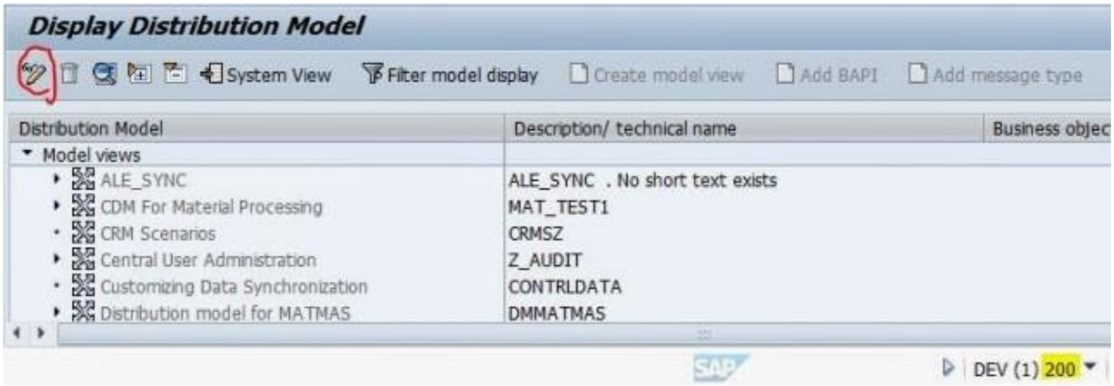
**Step53. Now go to Tcode- BD64 to create Customer Distribution Model.**



BD64

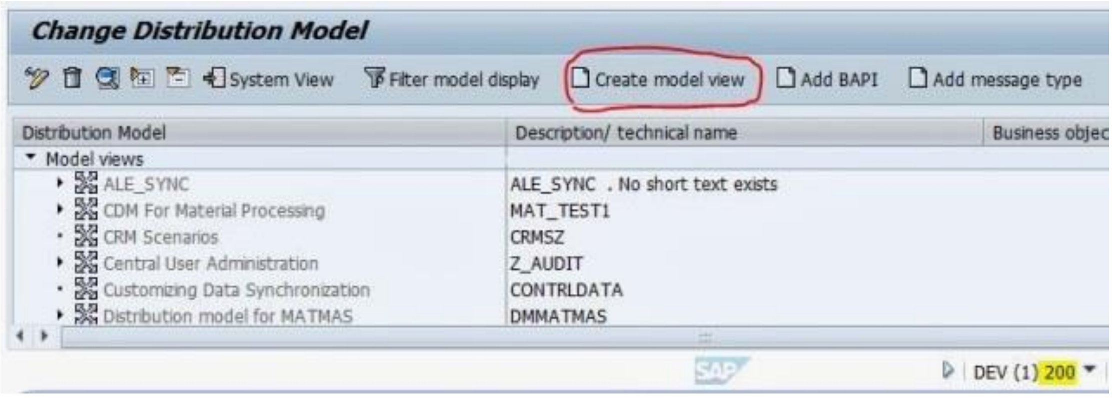
SAP Easy Access

**Step54. Click on Edit button.**



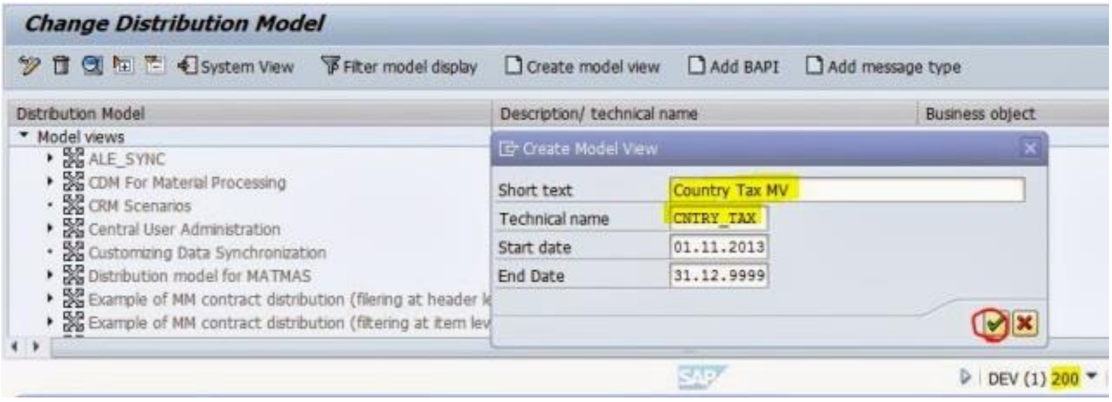
| Distribution Model | Description/ technical name | Business objec |
|---|---|---|
| Model views | | |
| ALE_SYNC | ALE_SYNC . No short text exists | |
| CDM For Material Processing | MAT_TEST1 | |
| CRM Scenarios | CRMSZ | |
| Central User Administration | Z_AUDIT | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |

**Step55. Now click on Create Model view.**



| Distribution Model | Description/ technical name | Business objec |
|---|---|---|
| Model views | | |
| ALE_SYNC | ALE_SYNC . No short text exists | |
| CDM For Material Processing | MAT_TEST1 | |
| CRM Scenarios | CRMSZ | |
| Central User Administration | Z_AUDIT | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |

**Step56. Provide Technical name, short text and click on Continue button.**

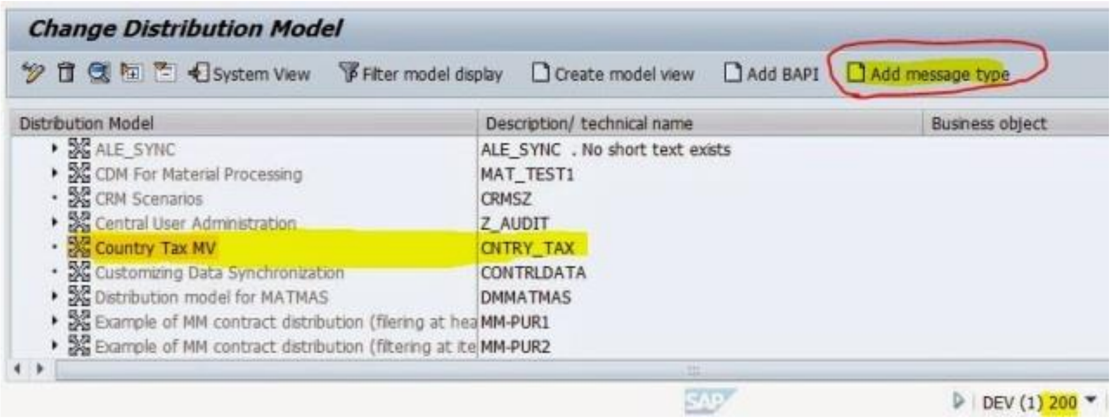


| Distribution Model | Description/ technical name | Business object |
|---|---|---|
| Model views | | |
| ALE_SYNC | | |
| CDM For Material Processing | | |
| CRM Scenarios | | |
| Central User Administration | | |
| Customizing Data Synchronization | | |
| Distribution model for MATMAS | | |
| Example of MM contract distribution (filering at header le | | |
| Example of MM contract distribution (filtering at item lev | | |

Create Model View

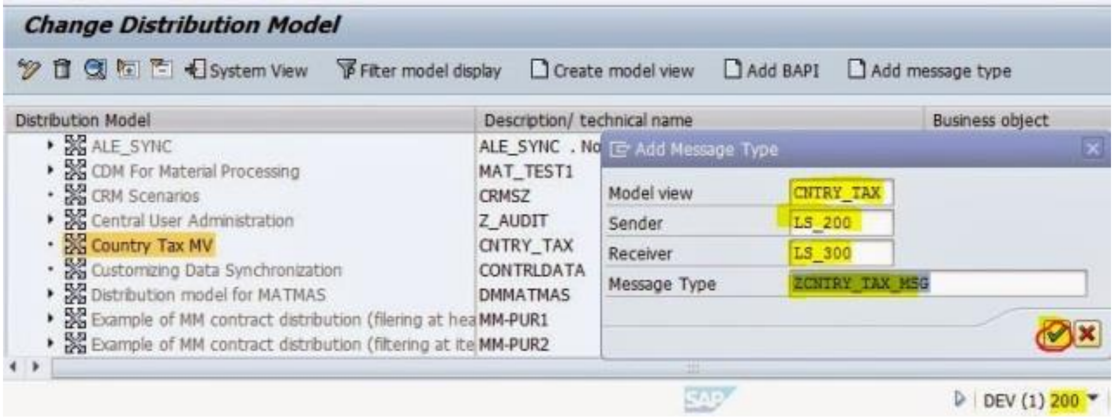
| | |
|---|---|
| Short text | Country Tax MV |
| Technical name | CNTRY_TAX |
| Start date | 01.11.2013 |
| End Date | 31.12.9999 |

**Step57. Now click on Add Message Type button.**



| Distribution Model | Description/ technical name | Business object |
|---|---|---|
| ALE_SYNC | ALE_SYNC . No short text exists | |
| CDM For Material Processing | MAT_TEST1 | |
| CRM Scenarios | CRMSZ | |
| Central User Administration | Z_AUDIT | |
| Country Tax MV | CNTRY_TAX | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |
| Example of MM contract distribution (filering at hea | MM-PUR1 | |
| Example of MM contract distribution (filtering at ite | MM-PUR2 | |

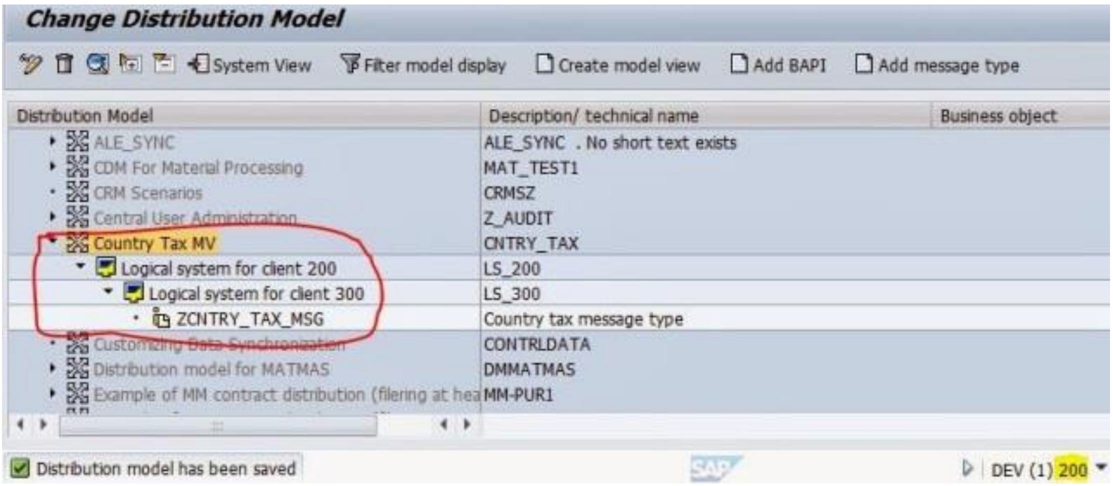
**Step58. Provide the sender logical system, receiver logical system, message type and click on Yes button.**



| Distribution Model | Description/ technical name | Business object |
|---|---|---|
| ALE_SYNC | ALE_SYNC . No | |
| CDM For Material Processing | MAT_TEST1 | |
| CRM Scenarios | CRMSZ | |
| Central User Administration | Z_AUDIT | |
| Country Tax MV | CNTRY_TAX | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |
| Example of MM contract distribution (filering at hea | MM-PUR1 | |
| Example of MM contract distribution (filtering at ite | MM-PUR2 | |

| | |
|---|---|
| Model view | CNTRY_TAX |
| Sender | LS_200 |
| Receiver | LS_300 |
| Message Type | ZCNTRY_TAX_MSG |

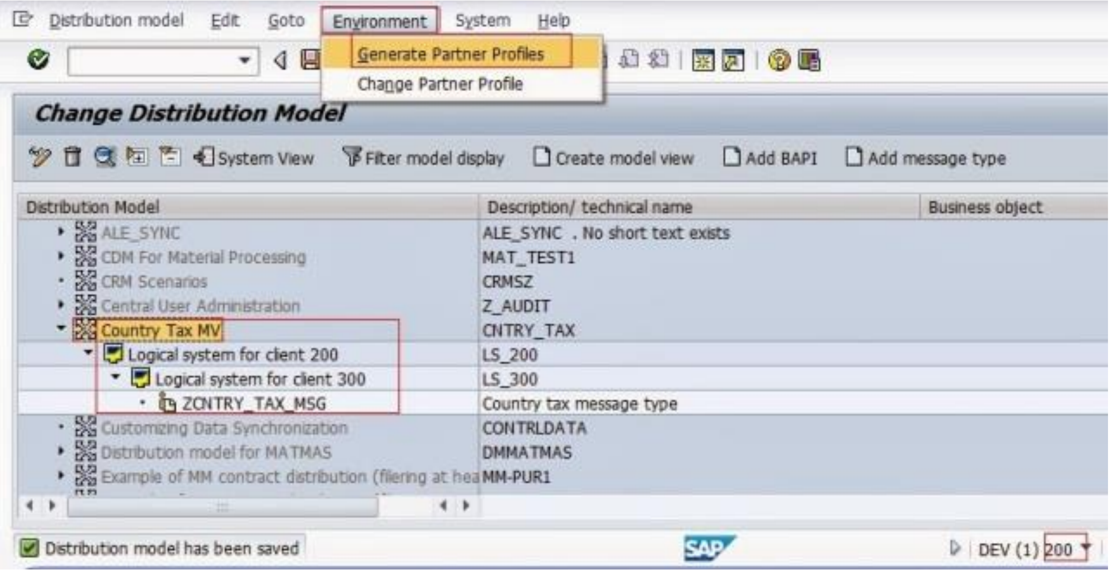
**Step59. Now the Customer Distribution model looks like as shown below.**



| Distribution Model | Description/ technical name | Business object |
|---|---|---|
| ALE_SYNC | ALE_SYNC . No short text exists | |
| CDM For Material Processing | MAT_TEST1 | |
| CRM Scenarios | CRMSZ | |
| Central User Administration | Z_AUDIT | |
| Country Tax MV | CNTRY_TAX | |
| Logical system for client 200 | LS_200 | |
| Logical system for client 300 | LS_300 | |
| ZCNTRY_TAX_MSG | Country tax message type | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |
| Example of MM contract distribution (filering at hea | MM-PUR1 | |

Distribution model has been saved

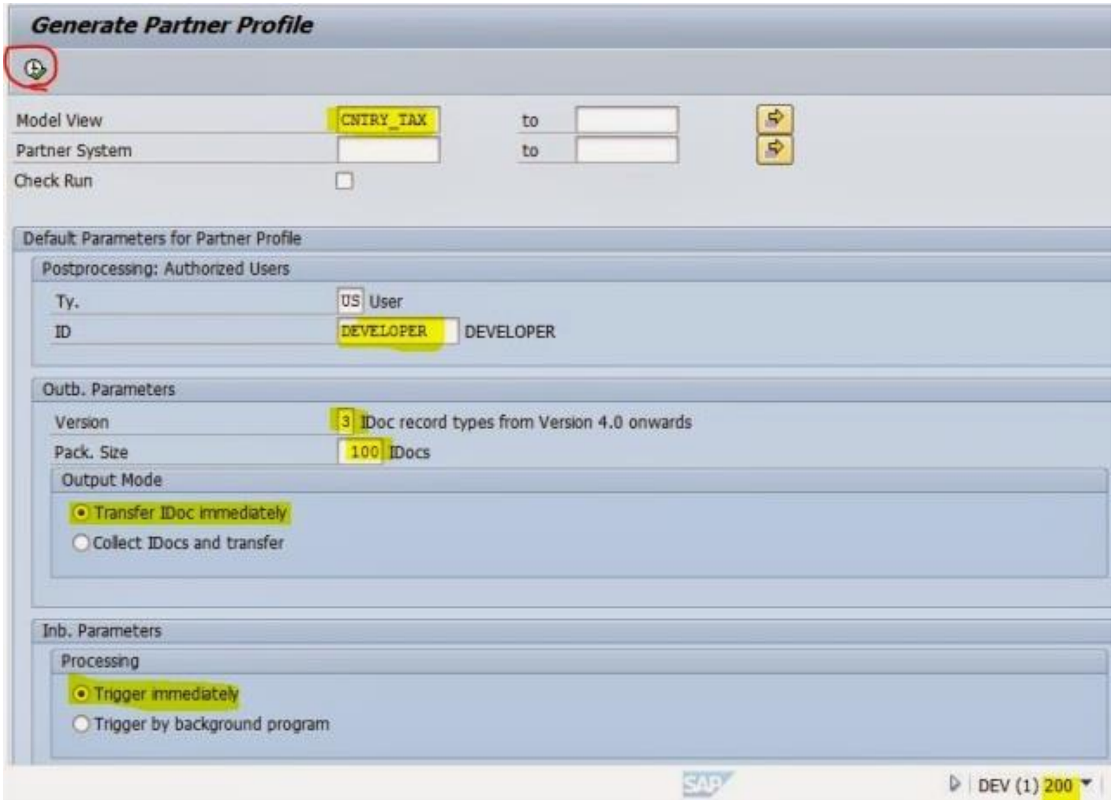
**Step60. Select the distribution model and from the Environment menu click on generate partner profiles.**



| Distribution Model | Description/ technical name | Business object |
|---|---|---|
| ALE_SYNC | ALE_SYNC . No short text exists | |
| CDM For Material Processing | MAT_TEST1 | |
| CRM Scenarios | CRMSZ | |
| Central User Administration | Z_AUDIT | |
| Country Tax MV | CNTRY_TAX | |
| Logical system for client 200 | LS_200 | |
| Logical system for client 300 | LS_300 | |
| ZCNTRY_TAX_MSG | Country tax message type | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |
| Example of MM contract distribution (filering at hea | MM-PUR1 | |

Distribution model has been saved

**Step61. Execute the program.**



Generate Partner Profile

Model View CNTRY_TAX to
Partner System to
Check Run □

Default Parameters for Partner Profile

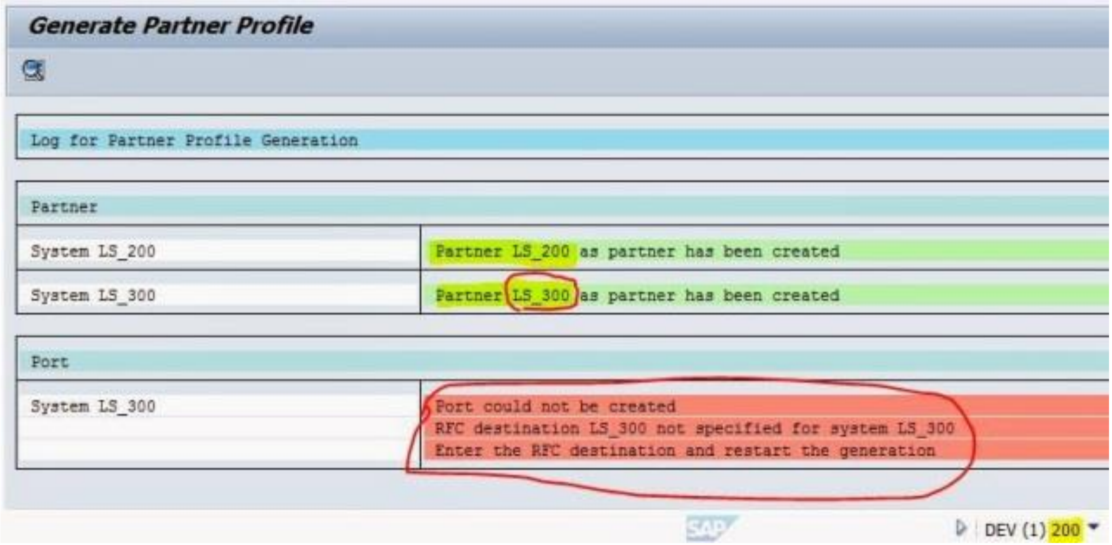
Postprocessing: Authorized Users
Ty. US User
ID DEVELOPER DEVELOPER

Outb. Parameters
Version 3 IDoc record types from Version 4.0 onwards
Pack. Size 100 IDocs
Output Mode
● Transfer IDoc immediately
○ Collect IDocs and transfer

Inb. Parameters
Processing
● Trigger immediately
○ Trigger by background program

DEV (1) 200

**Step62. It generates the partner profiles LS_300 and LS_200.**

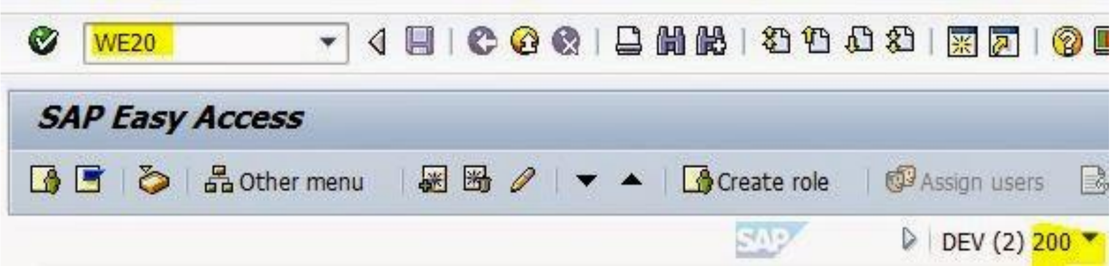


Generate Partner Profile

```
Log for Partner Profile Generation

Partner
System LS_200     Partner LS_200 as partner has been created
System LS_300     Partner LS_300 as partner has been created

Port
System LS_300     Port could not be created
                  RFC destination LS_300 not specified for system LS_300
                  Enter the RFC destination and restart the generation
```

DEV (1) 200

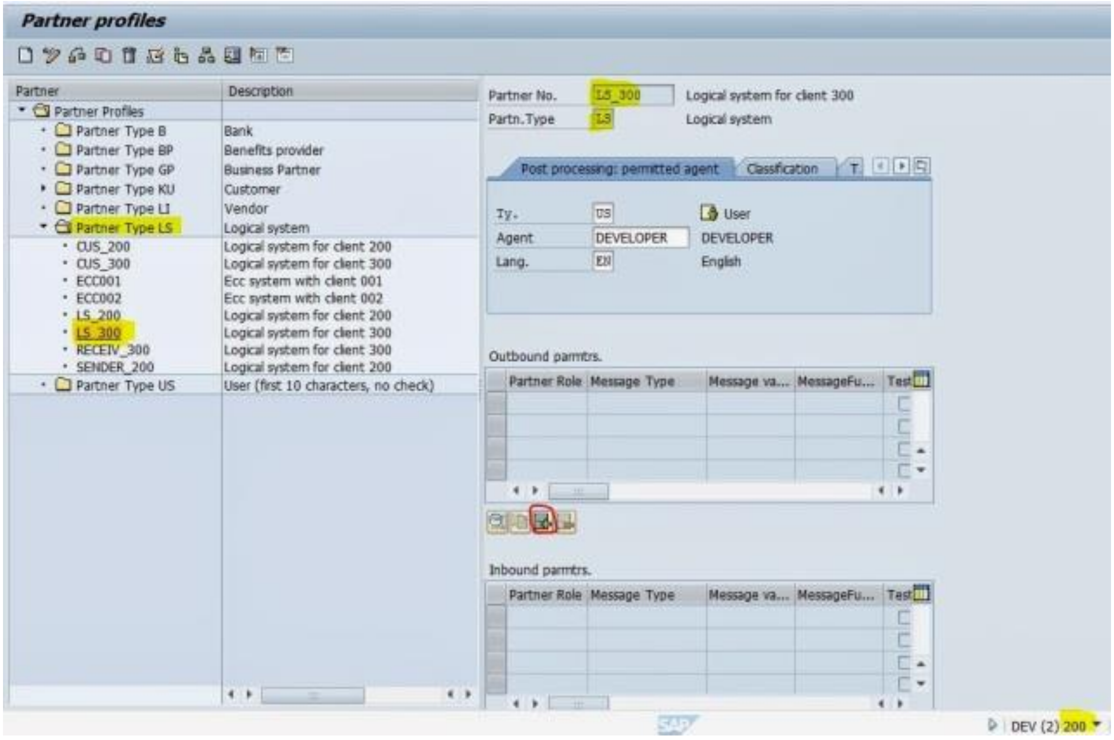
**Step63. Lets go to the Tcode WE20 to edit the generated partner profiles.**



WE20

SAP Easy Access

**Step64. Under the partner type LS, select the LS_300. Click on  the Add button as highlighted under Outbound Parameters.**



Partner profiles

| 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 |
| CUS_200 | Logical system for client 200 |
| CUS_300 | Logical system for client 300 |
| ECC001 | Ecc system with client 001 |
| ECC002 | Ecc system with client 002 |
| LS_200 | Logical system for client 200 |
| LS_300 | Logical system for client 300 |
| RECEIV_300 | Logical system for client 300 |
| SENDER_200 | Logical system for client 200 |
| Partner Type US | User (first 10 characters, no check) |

Partner No. LS_300 Logical system for client 300
Partn.Type LS Logical system

Post processing: permitted agent | Classification | T

Ty. US User
Agent DEVELOPER DEVELOPER
Lang. EN English

Outbound parmtrs.

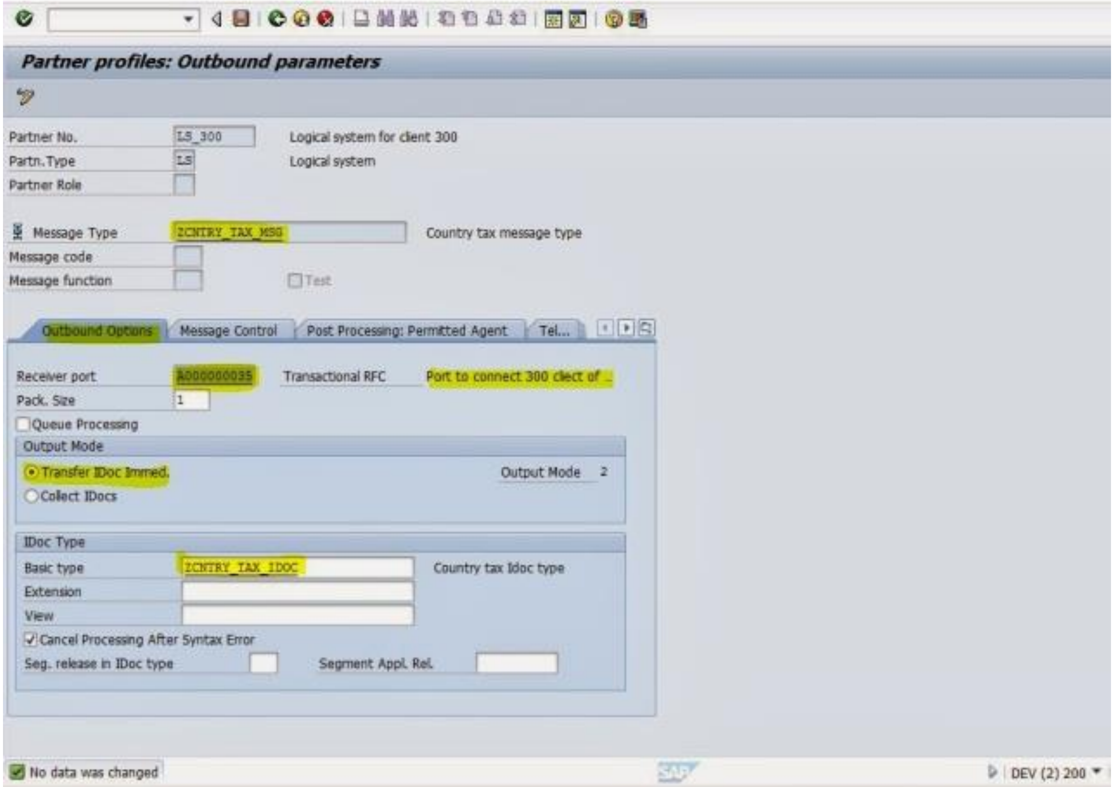
| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|

Inbound parmtrs.

| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|

DEV (2) 200

**Step65. Provide the message type, port and Idoc type, save it and click on back button.**



Partner profiles: Outbound parameters

Partner No. LS_300 Logical system for client 300
Partn.Type LS Logical system
Partner Role

Message Type ZCNTRY_TAX_MSG Country tax message type
Message code
Message function □ Test

Outbound Options | Message Control | Post Processing: Permitted Agent | Tel...

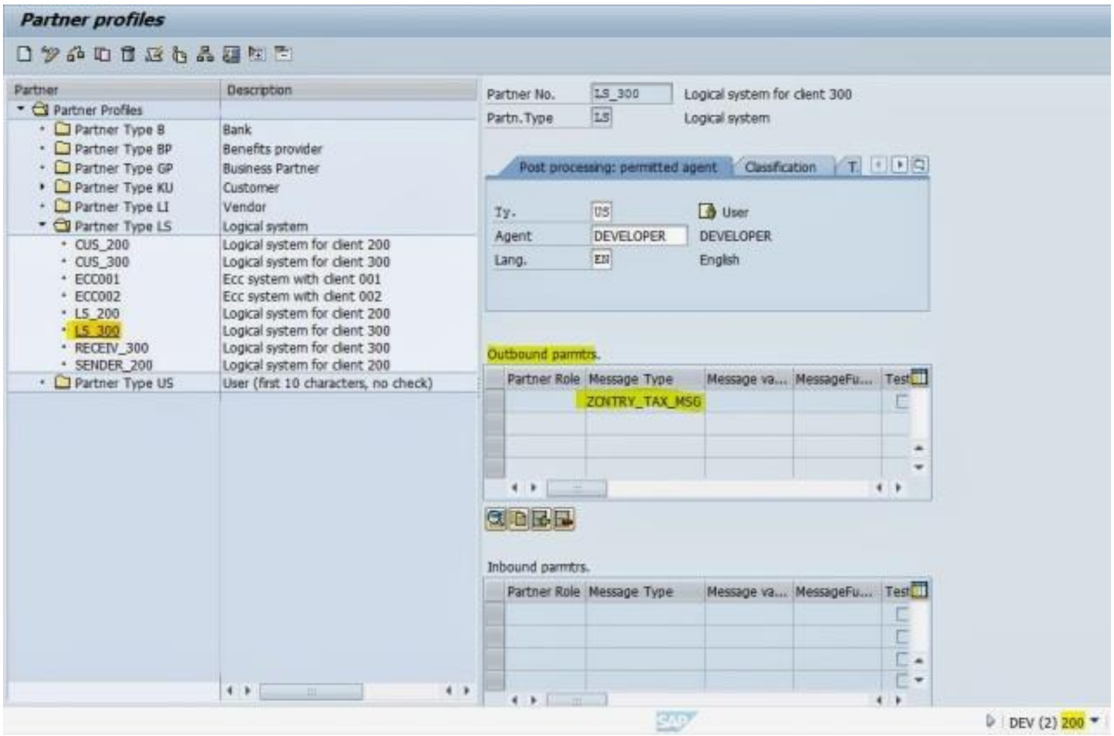
Receiver port A000000035 Transactional RFC Port to connect 300 clect of ...
Pack. Size 1
□ Queue Processing

Output Mode
● Transfer IDoc Immed. Output Mode 2
○ Collect IDocs

IDoc Type
Basic type ZCNTRY_TAX_IDOC Country tax Idoc type
Extension
View
☑ Cancel Processing After Syntax Error
Seg. release in IDoc type Segment Appl. Rel.

No data was changed DEV (2) 200

**Step66. Now the message type is added under the outbound parameter.**



Partner profiles

| 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 |
| CUS_200 | Logical system for client 200 |
| CUS_300 | Logical system for client 300 |
| ECC001 | Ecc system with client 001 |
| ECC002 | Ecc system with client 002 |
| LS_200 | Logical system for client 200 |
| LS_300 | Logical system for client 300 |
| RECEIV_300 | Logical system for client 300 |
| SENDER_200 | Logical system for client 200 |
| Partner Type US | User (first 10 characters, no check) |

Partner No. LS_300 Logical system for client 300

Partn.Type LS Logical system

Post processing: permitted agent | Classification

Ty. US User

Agent DEVELOPER DEVELOPER

Lang. EN English

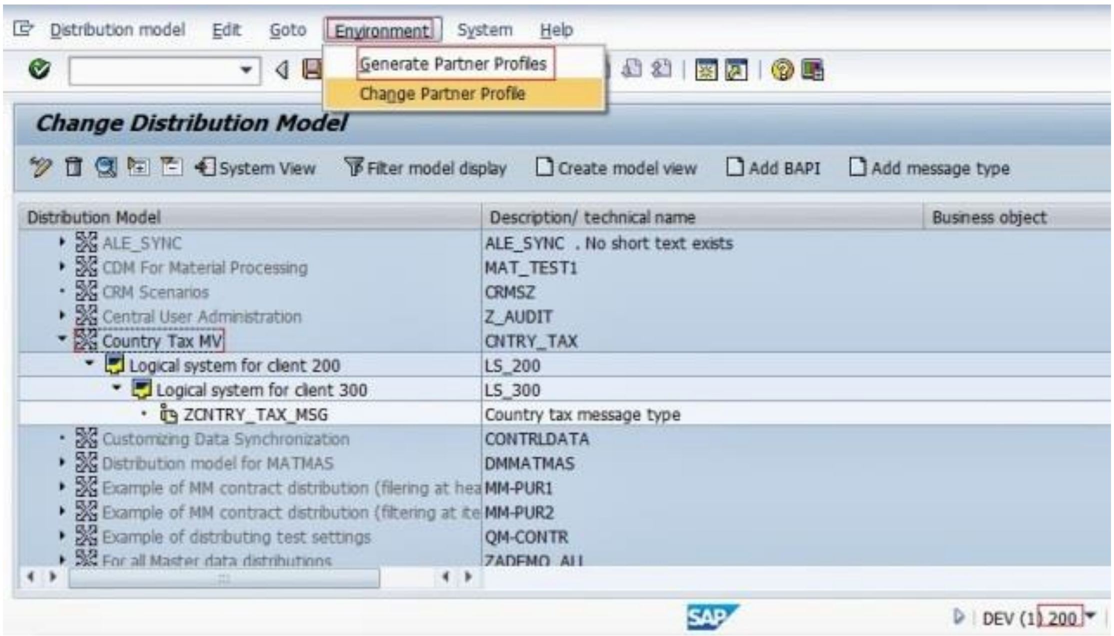
Outbound parmtrs.

| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|
| | ZCNTRY_TAX_MSG | | | □ |

Inbound parmtrs.

| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|

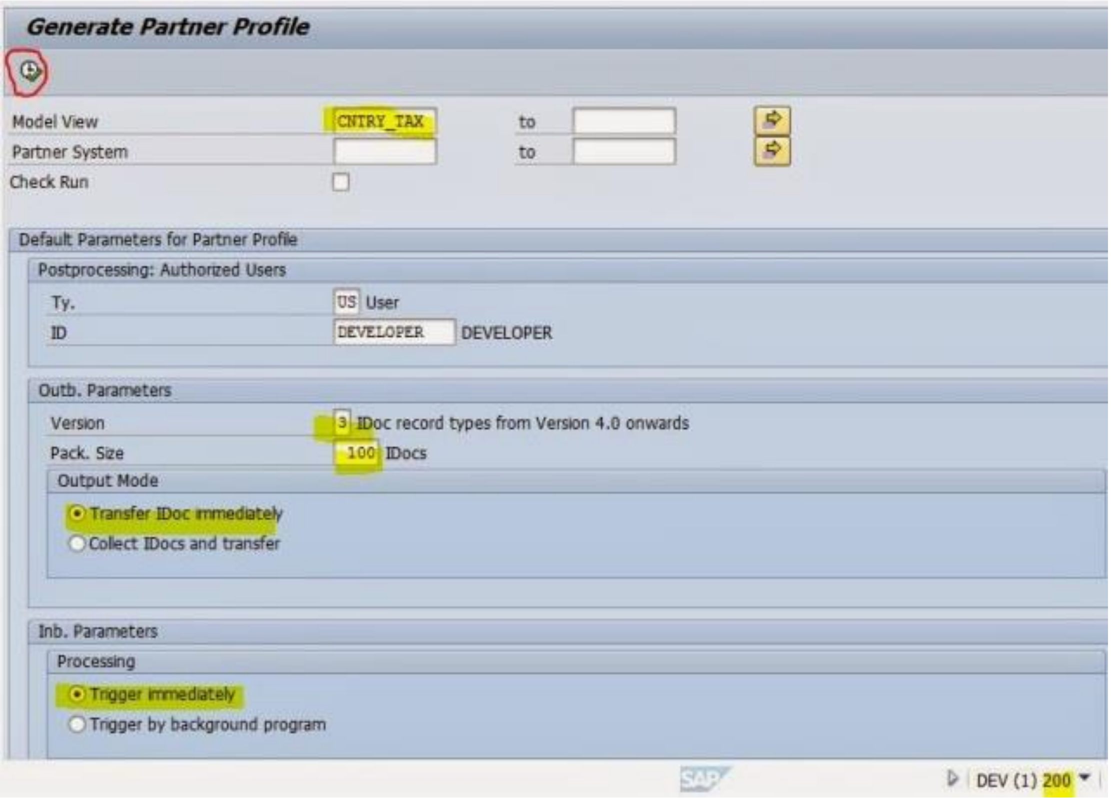
**Step67. Go to Tcode- BD64, select the customer distribution model and from the environment menu click on Generate Partner profile.**



Change Distribution Model

| Distribution Model | Description/ technical name | Business object |
|---|---|---|
| ALE_SYNC | ALE_SYNC . No short text exists | |
| CDM For Material Processing | MAT_TEST1 | |
| CRM Scenarios | CRMSZ | |
| Central User Administration | Z_AUDIT | |
| Country Tax MV | CNTRY_TAX | |
| Logical system for client 200 | LS_200 | |
| Logical system for client 300 | LS_300 | |
| ZCNTRY_TAX_MSG | Country tax message type | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |
| Example of MM contract distribution (filering at hea | MM-PUR1 | |
| Example of MM contract distribution (filtering at ite | MM-PUR2 | |
| Example of distributing test settings | QM-CONTR | |
| For all Master data distributions | ZADEMO_ALL | |

**Step68. Execute the below program.**



Generate Partner Profile

Model View: CNTRY_TAX to

Partner System: to

Check Run: □

Default Parameters for Partner Profile

Postprocessing: Authorized Users

Ty.: US User

ID: DEVELOPER DEVELOPER

Outb. Parameters

Version: 3 IDoc record types from Version 4.0 onwards

Pack. Size: 100 IDocs

Output Mode

● Transfer IDoc immediately

○ Collect IDocs and transfer

Inb. Parameters

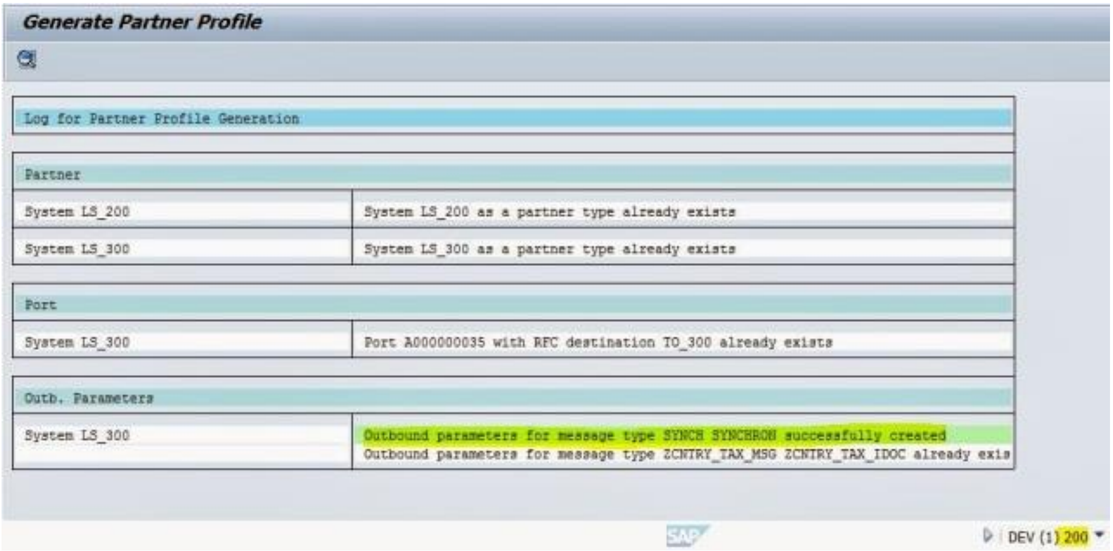
Processing

● Trigger immediately

○ Trigger by background program

DEV (1) 200

**Step69. There is no red line. So all teh set up is fine.**

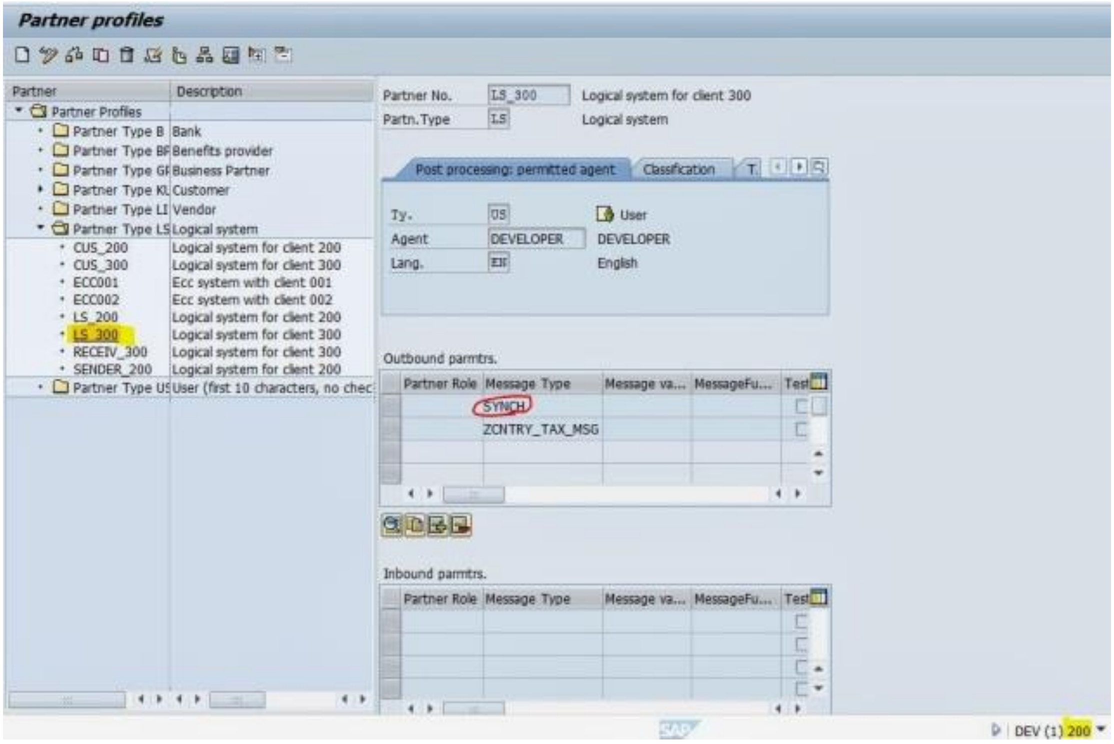


Generate Partner Profile

```
Log for Partner Profile Generation

Partner
System LS_200        System LS_200 as a partner type already exists
System LS_300        System LS_300 as a partner type already exists

Port
System LS_300        Port A000000035 with RFC destination TO_300 already exists

Outb. Parameters
System LS_300        Outbound parameters for message type SYNCH SYNCHRON successfully created
                     Outbound parameters for message type ZCNTRY_TAX_MSG ZCNTRY_TAX_IDOC already exis
```

DEV (1) 200

**Step70. Now go to Tcode – We20 to see again the partner profile. The SYNCH message type is automatically added to the partner profile LS_300 when it is generated. Now double click on teh message type SYNCH.**



Partner profiles

| Partner | Description |
|---|---|
| Partner Profiles | |
| Partner Type B | Bank |
| Partner Type BF | Benefits provider |
| Partner Type GF | Business Partner |
| Partner Type KL | Customer |
| Partner Type LI | Vendor |
| Partner Type LS | Logical system |
| CUS_200 | Logical system for client 200 |
| CUS_300 | Logical system for client 300 |
| ECC001 | Ecc system with client 001 |
| ECC002 | Ecc system with client 002 |
| LS_200 | Logical system for client 200 |
| LS_300 | Logical system for client 300 |
| RECEIV_300 | Logical system for client 300 |
| SENDER_200 | Logical system for client 200 |
| Partner Type US | User (first 10 characters, no chec |

Partner No. LS_300 Logical system for client 300
Partn.Type LS Logical system

Post processing: permitted agent | Classification | T.

Ty. US User
Agent DEVELOPER DEVELOPER
Lang. EN English

Outbound parmtrs.

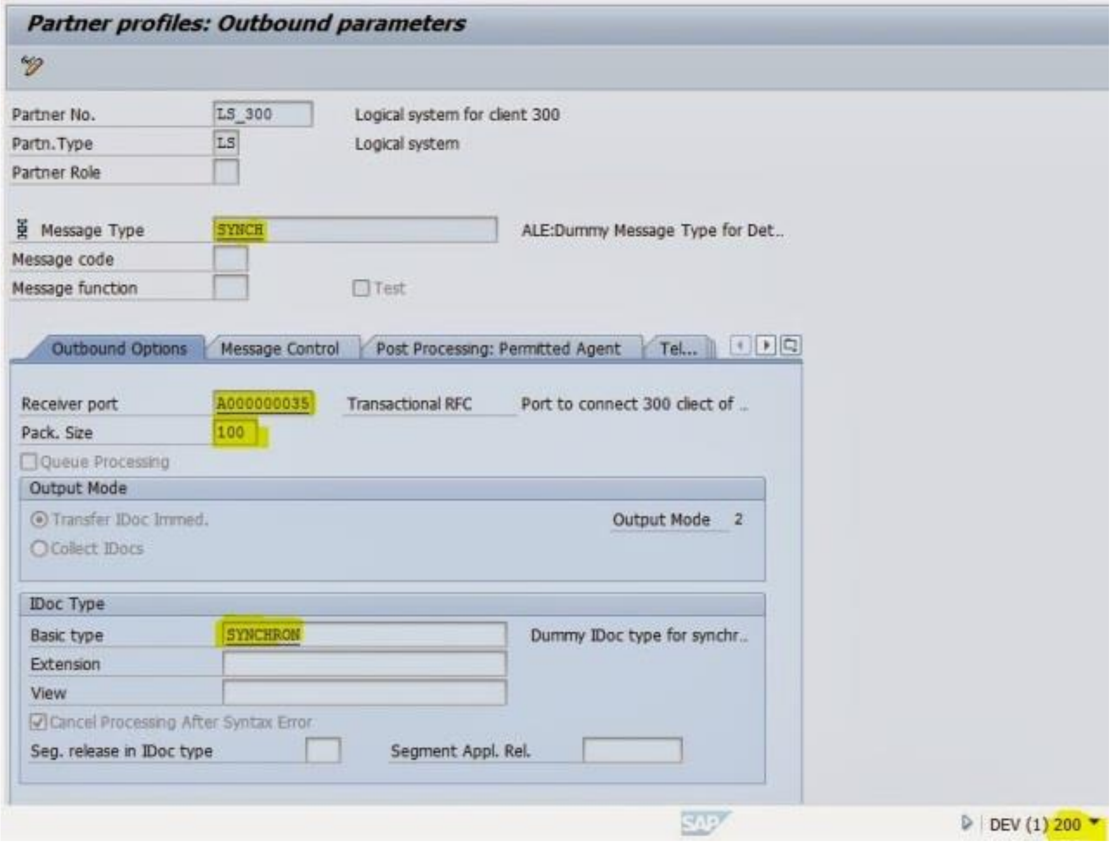
| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|
| | SYNCH | | | □ |
| | ZCNTRY_TAX_MSG | | | □ |

Inbound parmtrs.

| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|

DEV (1) 200

**Step71. The port and other details are automatically automatically assigned. Save it and go back.**



Partner profiles: Outbound parameters

Partner No. LS_300 Logical system for client 300
Partn.Type LS Logical system
Partner Role

Message Type SYNCH ALE:Dummy Message Type for Det..
Message code
Message function □ Test

Outbound Options | Message Control | Post Processing: Permitted Agent | Tel...

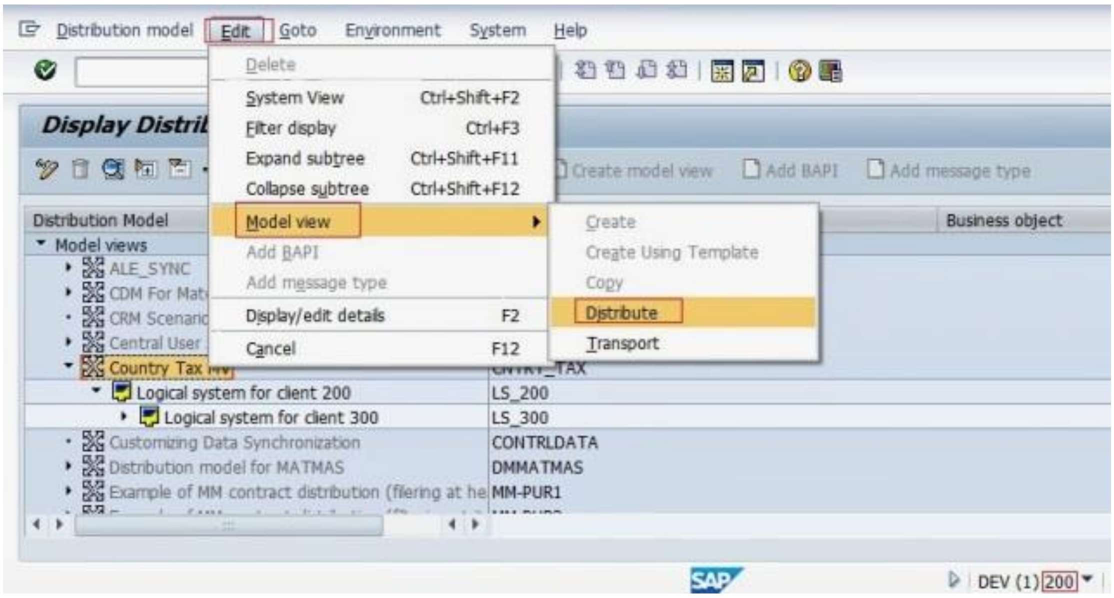
Receiver port A000000035 Transactional RFC Port to connect 300 clect of ..
Pack. Size 100
□ Queue Processing

Output Mode
● Transfer IDoc Immed. Output Mode 2
○ Collect IDocs

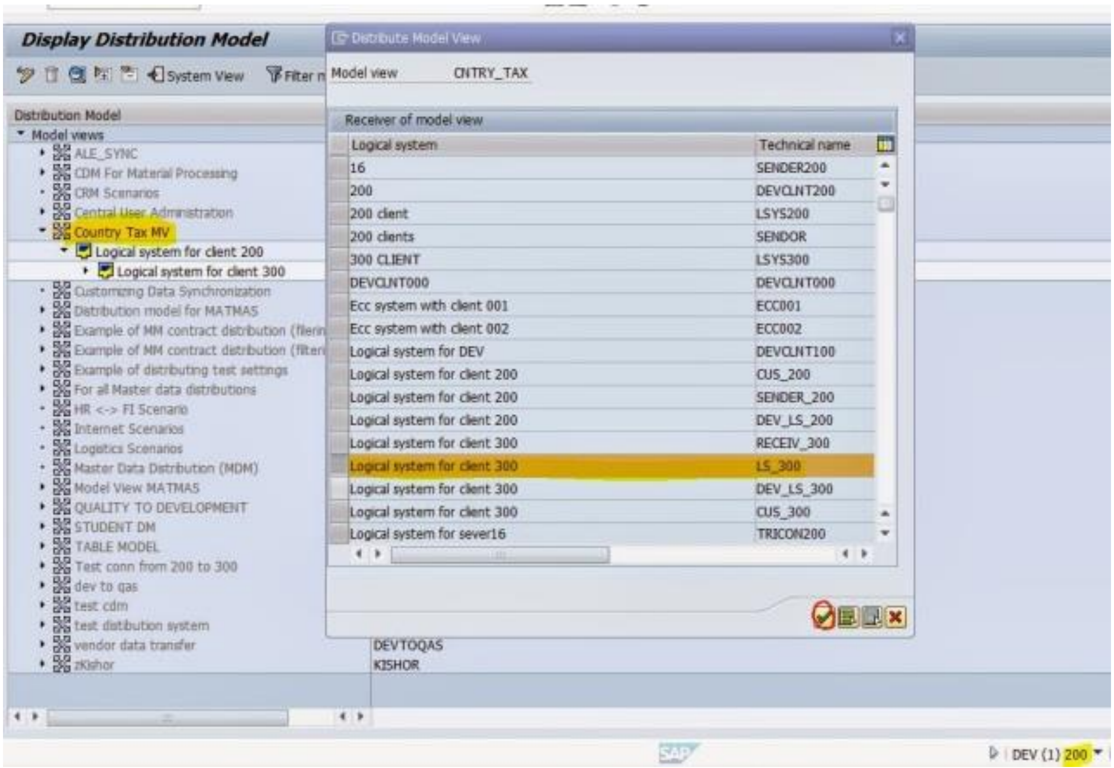
IDoc Type
Basic type SYNCHRON Dummy IDoc type for synchr..
Extension
View
☑ Cancel Processing After Syntax Error
Seg. release in IDoc type Segment Appl. Rel.

DEV (1) 200

**Step72. Now go to teh tcode- BD64, Select the Customer distribution model and Navigate along the highlighted path to distribute the Distribution model.**



**Step73. Select the receiver model view as LS_300.**

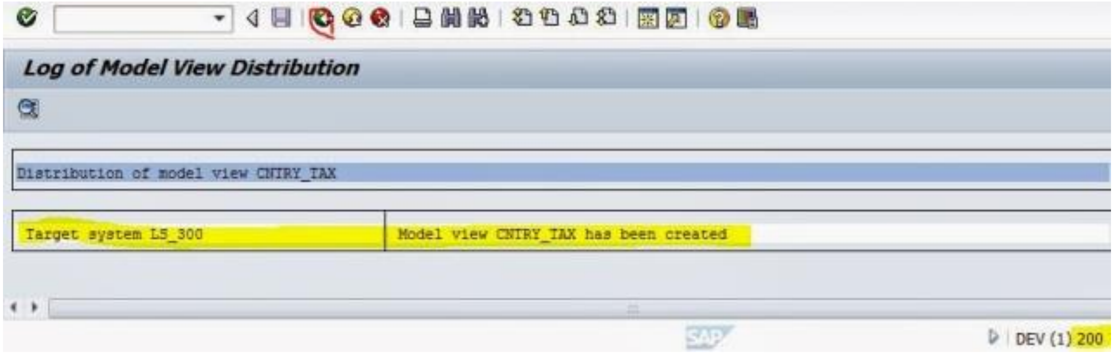

Display Distribution Model
System View
Filter m
Distribution Model
Model views
ALE_SYNC
CDM For Material Processing
CRM Scenarios
Central User Administration
Country Tax MV
Logical system for client 200
Logical system for client 300
Customizing Data Synchronization
Distribution model for MATMAS
Example of MM contract distribution (filerin
Example of MM contract distribution (filteri
Example of distributing test settings
For all Master data distributions
HR <-> FI Scenario
Internet Scenarios
Logistics Scenarios
Master Data Distribution (MDM)
Model View MATMAS
QUALITY TO DEVELOPMENT
STUDENT DM
TABLE MODEL
Test conn from 200 to 300
dev to qas
test cdm
test distibution system
vendor data transfer
zKishor
DEVTOQAS
KISHOR
Distribute Model View
DEV (1) 200

Model view CNTRY_TAX

Receiver of model view

| Logical system | Technical name |
|---|---|
| 16 | SENDER200 |
| 200 | DEVCLNT200 |
| 200 client | LSYS200 |
| 200 clients | SENDOR |
| 300 CLIENT | LSYS300 |
| DEVCLNT000 | DEVCLNT000 |
| Ecc system with client 001 | ECC001 |
| Ecc system with client 002 | ECC002 |
| Logical system for DEV | DEVCLNT100 |
| Logical system for client 200 | CUS_200 |
| Logical system for client 200 | SENDER_200 |
| Logical system for client 200 | DEV_LS_200 |
| Logical system for client 300 | RECEIV_300 |
| Logical system for client 300 | LS_300 |
| Logical system for client 300 | DEV_LS_300 |
| Logical system for client 300 | CUS_300 |
| Logical system for sever16 | TRICON200 |

**Step74. Now the distribution model is created in the target client (300).**

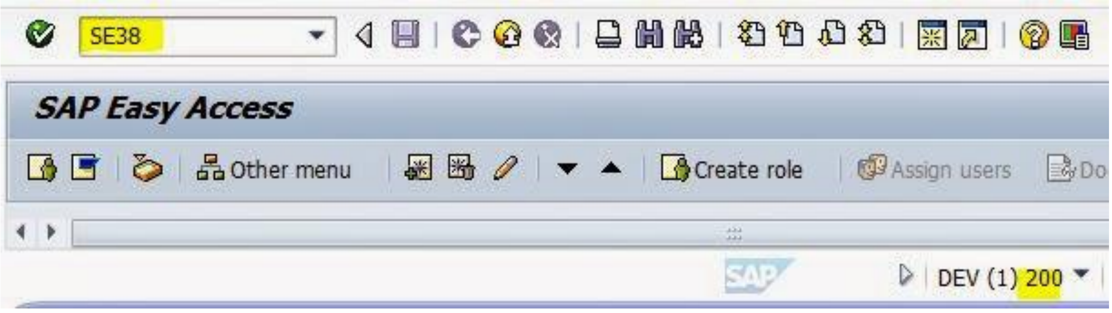


Log of Model View Distribution

Distribution of model view CNTRY_TAX

Target system LS_300 | Model view CNTRY_TAX has been created

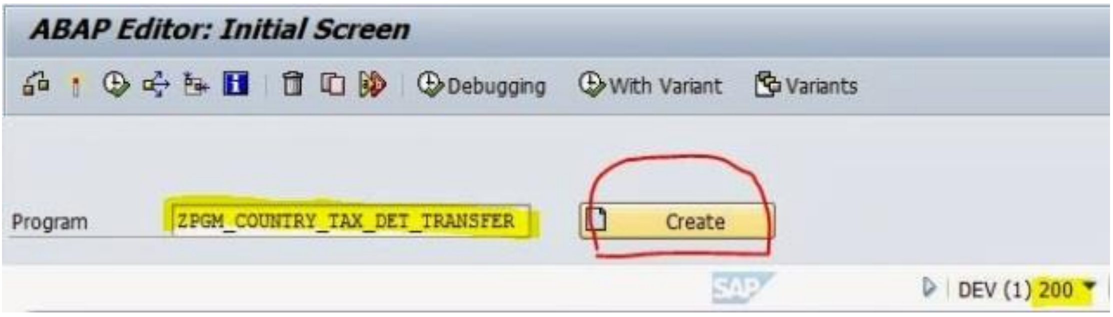
Step75. Now go to Tcode- SE38.



SE38

SAP Easy Access

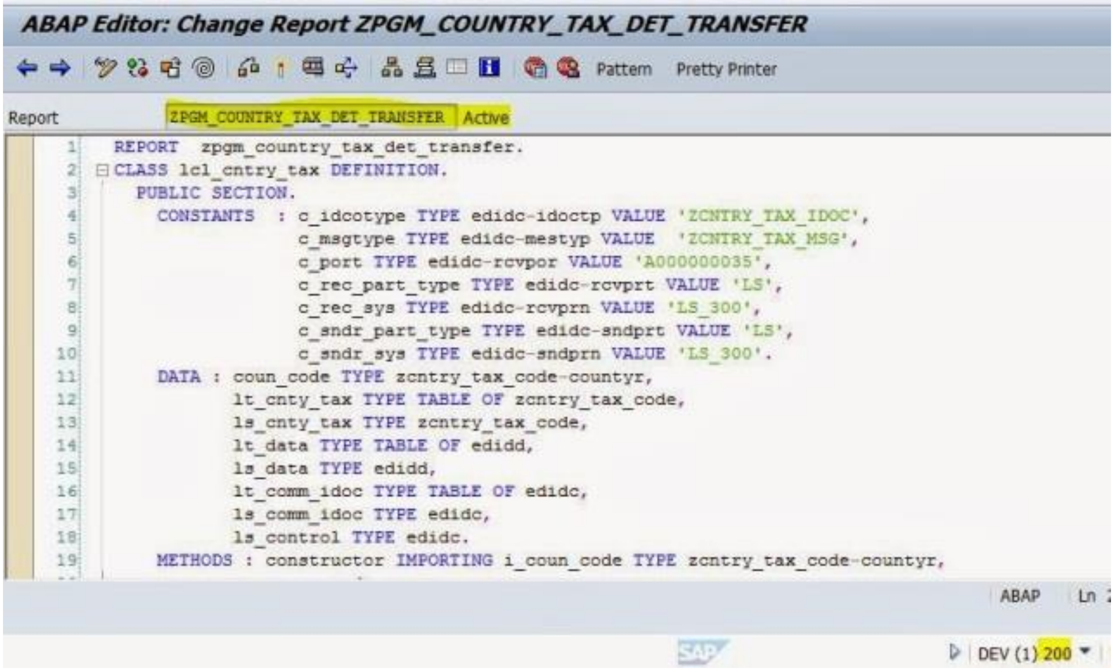
Step76. Provide a report name and create  it.



ABAP Editor: Initial Screen

Program ZPGM_COUNTRY_TAX_DET_TRANSFER

Step77. Provide the below code and activate it.


ABAP Editor: Change Report ZPGM_COUNTRY_TAX_DET_TRANSFER
Pattern
Pretty Printer
Report
ZPGM_COUNTRY_TAX_DET_TRANSFER
Active
ABAP
Ln
DEV (1) 200

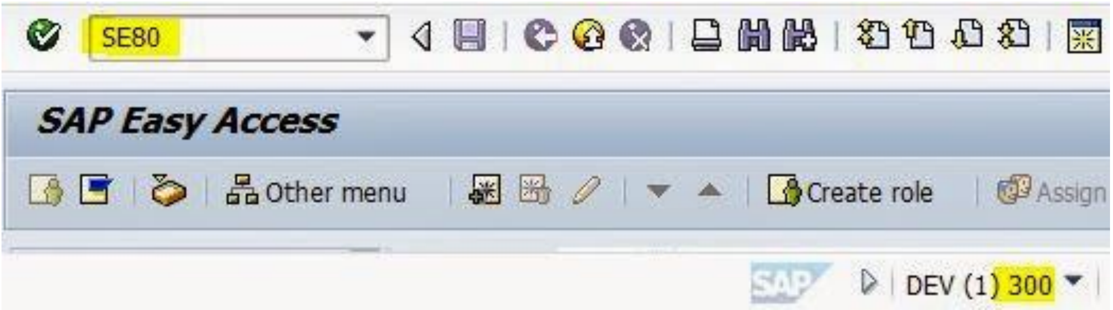
```
REPORT  zpgm_country_tax_det_transfer.
CLASS lcl_cntry_tax DEFINITION.
  PUBLIC SECTION.
    CONSTANTS  : c_idcotype TYPE edidc-idoctp VALUE 'ZCNTRY_TAX_IDOC',
                 c_msgtype TYPE edidc-mestyp VALUE  'ZCNTRY_TAX_MSG',
                 c_port TYPE edidc-rcvpor VALUE 'A000000035',
                 c_rec_part_type TYPE edidc-rcvprt VALUE 'LS',
                 c_rec_sys TYPE edidc-rcvprn VALUE 'LS_300',
                 c_sndr_part_type TYPE edidc-sndprt VALUE 'LS',
                 c_sndr_sys TYPE edidc-sndprn VALUE 'LS_300'.
    DATA : coun_code TYPE zcntry_tax_code-countyr,
           lt_cnty_tax TYPE TABLE OF zcntry_tax_code,
           ls_cnty_tax TYPE zcntry_tax_code,
           lt_data TYPE TABLE OF edidd,
           ls_data TYPE edidd,
           lt_comm_idoc TYPE TABLE OF edidc,
           ls_comm_idoc TYPE edidc,
           ls_control TYPE edidc.
    METHODS : constructor IMPORTING i_coun_code TYPE zcntry_tax_code-countyr,
```

```
REPORT  zpgm_country_tax_det_transfer.

CLASS lcl_cntry_tax DEFINITION.

  PUBLIC SECTION.

    CONSTANTS  : c_idcotype TYPE edidc-idoctp VALUE ‘ZCNTRY_TAX_IDOC’,

                 c_msgtype TYPE edidc-mestyp VALUE  ‘ZCNTRY_TAX_MSG’,

                 c_port TYPE edidc-rcvpor VALUE ‘A000000035’,

                 c_rec_part_type TYPE edidc-rcvprt VALUE ‘LS’,

                 c_rec_sys TYPE edidc-rcvprn VALUE ‘LS_300’,

                c_sndr_part_type TYPE edidc-sndprt VALUE ‘LS’,

                c_sndr_sys TYPE edidc-sndprn VALUE ‘LS_300’.

    DATA : coun_code TYPE zcntry_tax_code-countyr,

           lt_cnty_tax TYPE TABLE OF zcntry_tax_code,

           ls_cnty_tax TYPE zcntry_tax_code,

           lt_data TYPE TABLE OF edidd,

           ls_data TYPE edidd,

           lt_comm_idoc TYPE TABLE OF edidc,

           ls_comm_idoc TYPE edidc,

           ls_control TYPE edidc.
  METHODS : constructor IMPORTING i_coun_code TYPE zcntry_tax_code-countyr,
            prepare_data,
            prepare_cntl_data,
            send_idoc.
ENDCLASS.          “lcl_cntry_tax DEFINITION
CLASS lcl_cntry_tax IMPLEMENTATION.
 METHOD constructor.
  coun_code = i_coun_code.
 ENDMETHOD.          “constructor
 METHOD prepare_data.
  DATA : ls_seg TYPE zcntry_tax_seg.
  SELECT  * FROM zcntry_tax_code INTO TABLE  lt_cnty_tax WHERE countyr = coun_code.
  IF sy-subrc = 0.
   LOOP AT lt_cnty_tax INTO ls_cnty_tax.
    ls_data-segnam  = ‘ZCNTRY_TAX_SEG’. ” segment name
    ls_seg-countyr = ls_cnty_tax-countyr.
    ls_seg-tax_code = ls_cnty_tax-tax_code.
    ls_data-sdata = ls_seg.
    APPEND ls_data TO lt_data.
    CLEAR ls_data.
   ENDLOOP.
  ELSE.
   MESSAGE ‘No record found’ TYPE ‘E’.
  ENDIF.
 ENDMETHOD.          “prepare_data
 METHOD prepare_cntl_data.
  ls_control-idoctp = c_idcotype.
  ls_control-mestyp = c_msgtype.
  ls_control-rcvpor = c_port.
  ls_control-rcvprt = c_rec_part_type .
  ls_control-rcvprn = c_rec_sys.
  ls_control-sndprt = c_sndr_part_type .
  ls_control-sndprn = c_sndr_sys.
 ENDMETHOD.                  “prepare_cntl_data
 METHOD send_idoc.
  CALL FUNCTION ‘MASTER_IDOC_DISTRIBUTE’
   EXPORTING
    master_idoc_control          = ls_control
*     OBJ_TYPE                   = ”
*     CHNUM                      = ”
   TABLES
    communication_idoc_control     = lt_comm_idoc
    master_idoc_data             = lt_data
   EXCEPTIONS
    error_in_idoc_control         = 1
    error_writing_idoc_status      = 2
    error_in_idoc_data           = 3
    sending_logical_system_unknown = 4
    OTHERS                     = 5.
  IF sy-subrc = 0.
   LOOP AT lt_comm_idoc INTO ls_comm_idoc.
    WRITE      ‘Idoc Generated : ‘, ls_comm_idoc-docnum.
   ENDLOOP.
  ENDIF.
 ENDMETHOD.                  “send_idoc
ENDCLASS.                  “lcl_cntry_tax IMPLEMENTATION
START-OF-SELECTION.
 PARAMETERS : p_cntry TYPE zcntry_tax_code-countyr.
 DATA  : obj TYPE REF TO lcl_cntry_tax.
 CREATE OBJECT obj
  EXPORTING
   i_coun_code = p_cntry.
```

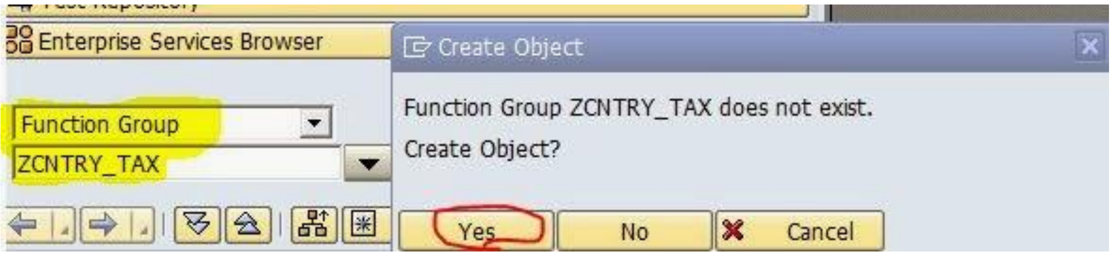
```
CALL METHOD obj->prepare_data.

CALL METHOD obj->prepare_cntl_data.

CALL METHOD obj->send_idoc.
```

---

**Step78. All the set up with the custom report in the source client 200 is ready. Now logon to the client 300. Go to Tcode- SE80.**



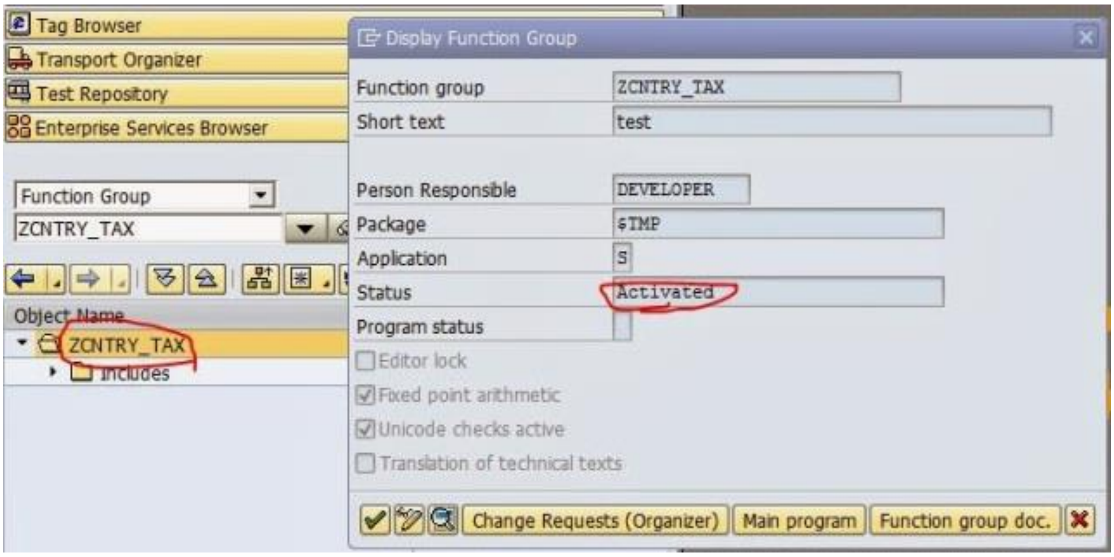
**Step79. Create a function group. Provide the name and hit the enter button. From the popup click on Yes button.**



Function Group
ZCNTRY_TAX

Function Group ZCNTRY_TAX does not exist.
Create Object?

**Step80. Activate the Function Group.**



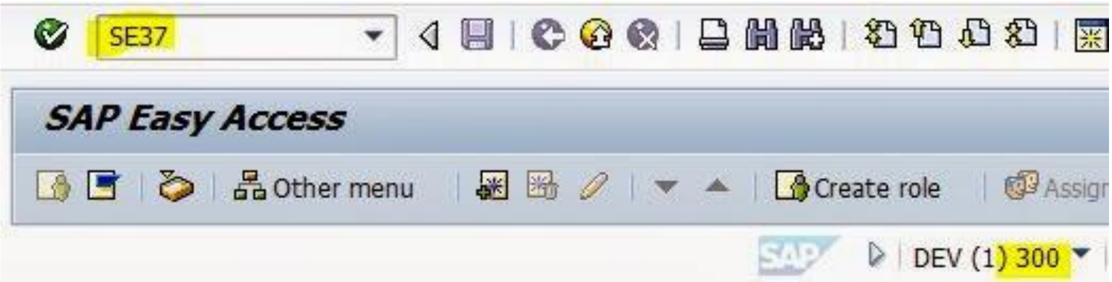
Function Group
ZCNTRY_TAX

ZCNTRY_TAX

| | |
|---|---|
| Function group | ZCNTRY_TAX |
| Short text | test |
| Person Responsible | DEVELOPER |
| Package | $TMP |
| Application | S |
| Status | Activated |
| Program status | |

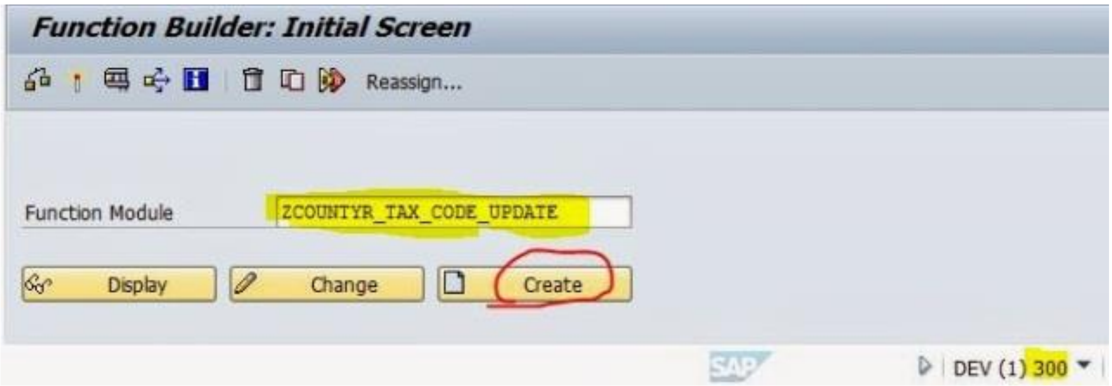
□ Editor lock
☑ Fixed point arithmetic
☑ Unicode checks active
□ Translation of technical texts

**Step81. Now go to Tcode- SE37 .**



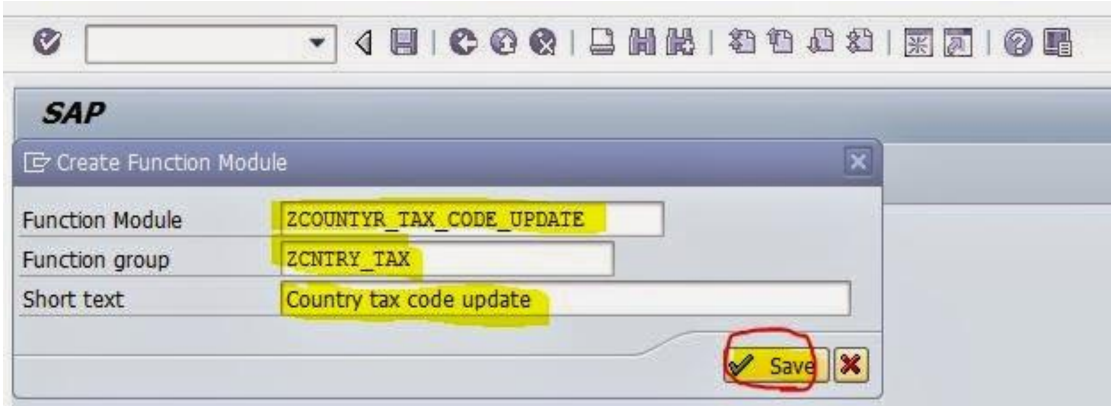
SE37

Step82. Now Provide a function module name and click on the create button.



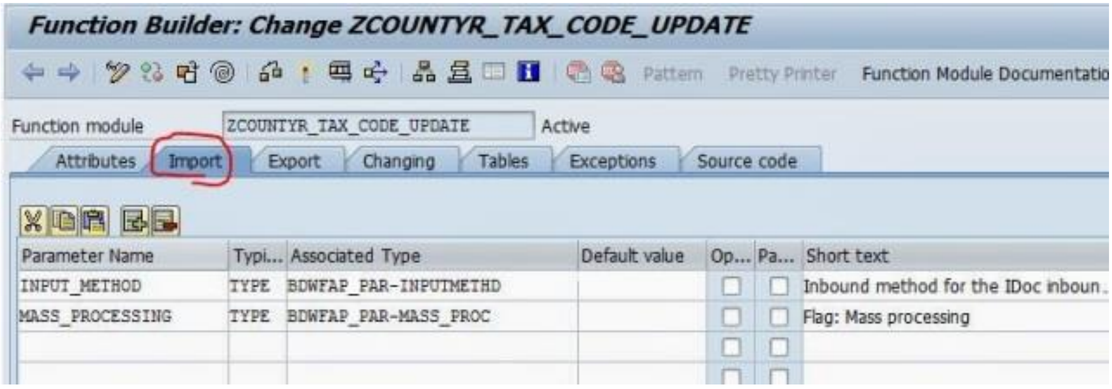
Function Module ZCOUNTYR_TAX_CODE_UPDATE

Step83. Provide the above created function group name and description and click on Save button.



| | |
|---|---|
| Function Module | ZCOUNTYR_TAX_CODE_UPDATE |
| Function group | ZCNTRY_TAX |
| Short text | Country tax code update |

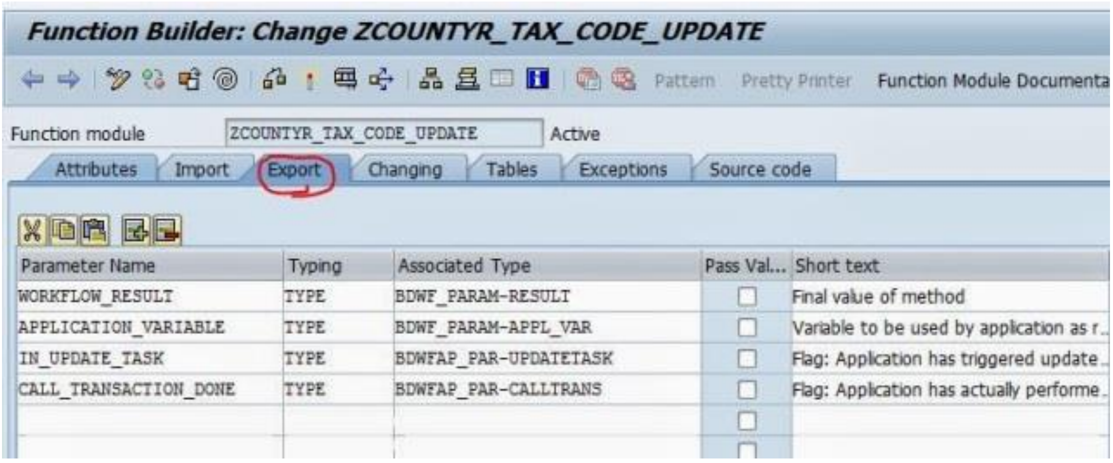
Step84. Provide the Importing parameters as shown below.



Function module ZCOUNTYR_TAX_CODE_UPDATE Active

| Parameter Name | Typi... | Associated Type | Default value | Op... | Pa... | Short text |
|---|---|---|---|---|---|---|
| INPUT_METHOD | TYPE | BDWFAP_PAR-INPUTMETHD | | □ | □ | Inbound method for the IDoc inboun |
| MASS_PROCESSING | TYPE | BDWFAP_PAR-MASS_PROC | | □ | □ | Flag: Mass processing |
| | | | | □ | □ | |

Step85. Provide the Exporting parameters as shown below.

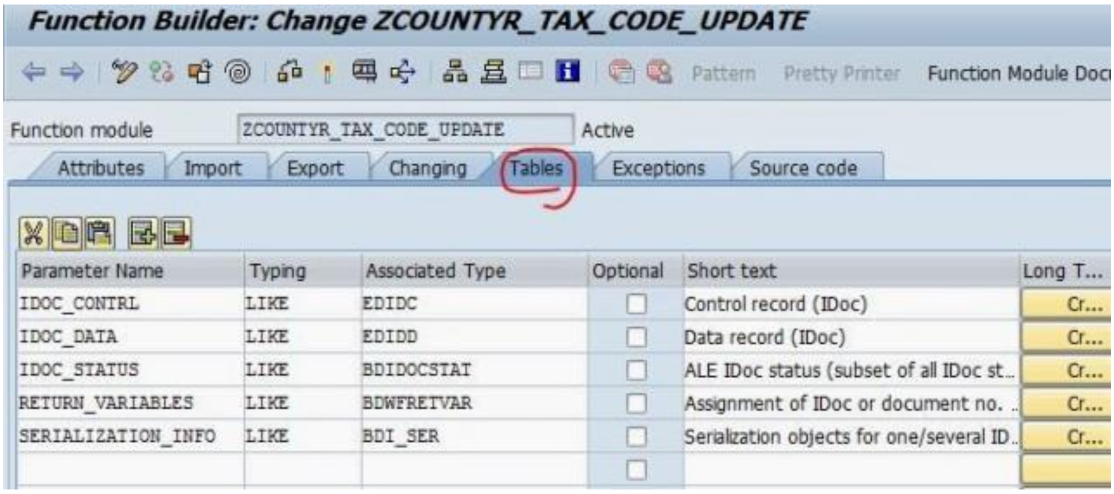


Function Builder: Change ZCOUNTYR_TAX_CODE_UPDATE

Function module ZCOUNTYR_TAX_CODE_UPDATE Active

| Parameter Name | Typing | Associated Type | Pass Val... | Short text |
|---|---|---|---|---|
| WORKFLOW_RESULT | TYPE | BDWF_PARAM-RESULT | □ | Final value of method |
| APPLICATION_VARIABLE | TYPE | BDWF_PARAM-APPL_VAR | □ | Variable to be used by application as r... |
| IN_UPDATE_TASK | TYPE | BDWFAP_PAR-UPDATETASK | □ | Flag: Application has triggered update... |
| CALL_TRANSACTION_DONE | TYPE | BDWFAP_PAR-CALLTRANS | □ | Flag: Application has actually performe... |
| | | | □ | |
| | | | □ | |

**Step86. Provide the parameters under the Tables section as shown below.**

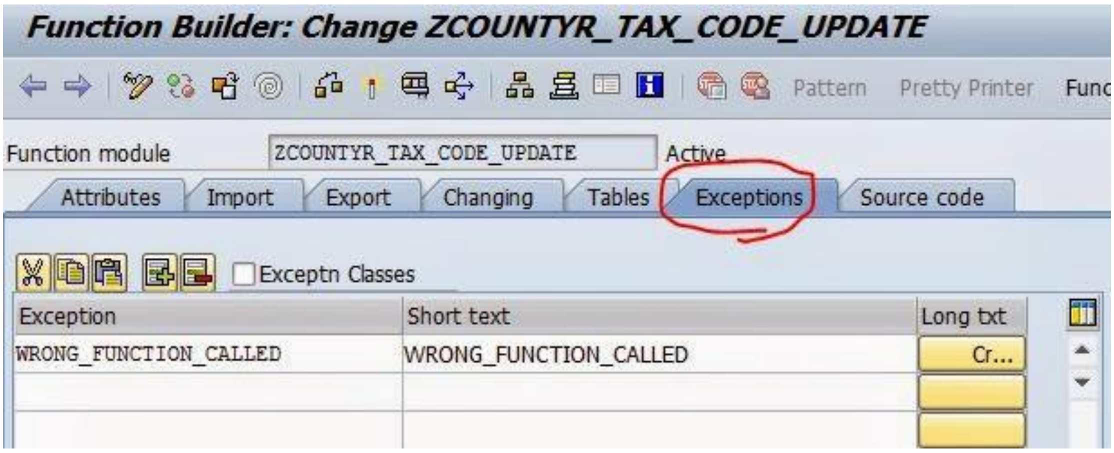


Function Builder: Change ZCOUNTYR_TAX_CODE_UPDATE

Function module ZCOUNTYR_TAX_CODE_UPDATE Active

| Parameter Name | Typing | Associated Type | Optional | Short text | Long T... |
|---|---|---|---|---|---|
| IDOC_CONTRL | LIKE | EDIDC | □ | Control record (IDoc) | Cr... |
| IDOC_DATA | LIKE | EDIDD | □ | Data record (IDoc) | Cr... |
| IDOC_STATUS | LIKE | BDIDOCSTAT | □ | ALE IDoc status (subset of all IDoc st... | Cr... |
| RETURN_VARIABLES | LIKE | BDWFRETVAR | □ | Assignment of IDoc or document no. ... | Cr... |
| SERIALIZATION_INFO | LIKE | BDI_SER | □ | Serialization objects for one/several ID... | Cr... |
| | | | □ | | |

**Step87. Provide the Exception as shown below.**



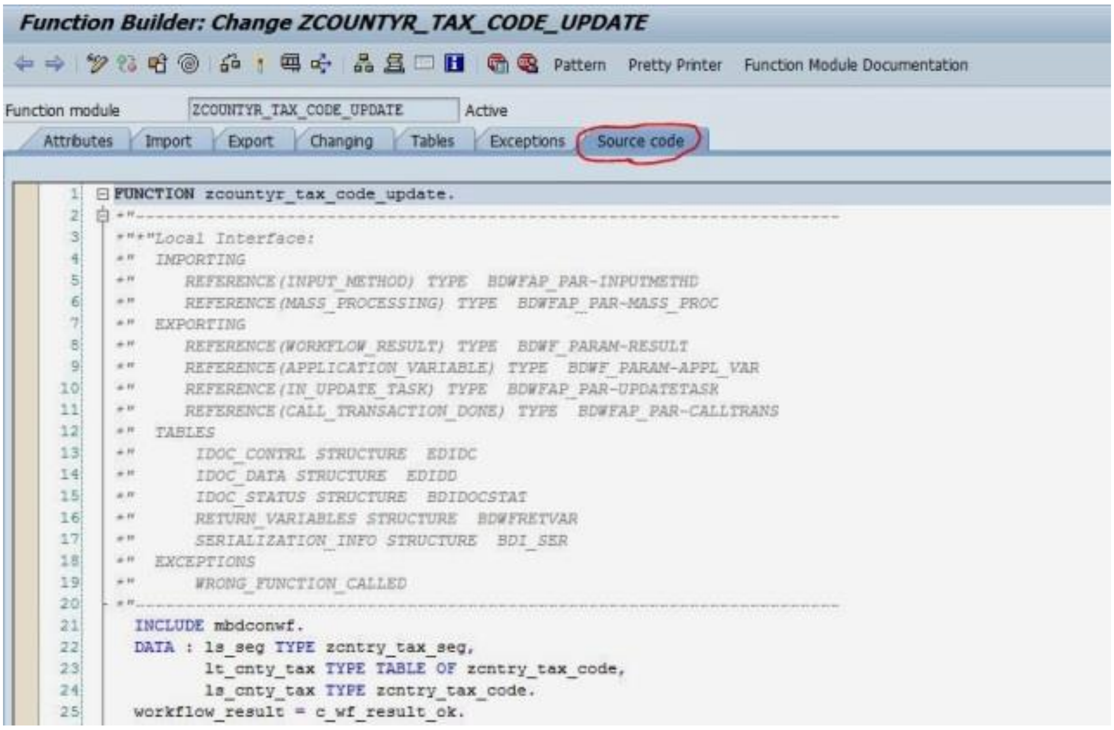
Function Builder: Change ZCOUNTYR_TAX_CODE_UPDATE

Function module ZCOUNTYR_TAX_CODE_UPDATE Active

□ Exceptn Classes

| Exception | Short text | Long txt |
|---|---|---|
| WRONG_FUNCTION_CALLED | WRONG_FUNCTION_CALLED | Cr... |
| | | |
| | | |

**Step88. Maintain the below code.**


Function Builder: Change ZCOUNTYR_TAX_CODE_UPDATE
Pattern
Pretty Printer
Function Module Documentation
Attributes
Import
Export
Changing
Tables
Exceptions
Source code

Function module ZCOUNTYR_TAX_CODE_UPDATE Active

```
FUNCTION zcountyr_tax_code_update.
*"----------------------------------------------------------------------
*"*"Local Interface:
*"  IMPORTING
*"     REFERENCE(INPUT_METHOD) TYPE  BDWFAP_PAR-INPUTMETHD
*"     REFERENCE(MASS_PROCESSING) TYPE  BDWFAP_PAR-MASS_PROC
*"  EXPORTING
*"     REFERENCE(WORKFLOW_RESULT) TYPE  BDWF_PARAM-RESULT
*"     REFERENCE(APPLICATION_VARIABLE) TYPE  BDWF_PARAM-APPL_VAR
*"     REFERENCE(IN_UPDATE_TASK) TYPE  BDWFAP_PAR-UPDATETASK
*"     REFERENCE(CALL_TRANSACTION_DONE) TYPE  BDWFAP_PAR-CALLTRANS
*"  TABLES
*"      IDOC_CONTRL STRUCTURE  EDIDC
*"      IDOC_DATA STRUCTURE  EDIDD
*"      IDOC_STATUS STRUCTURE  BDIDOCSTAT
*"      RETURN_VARIABLES STRUCTURE  BDWFRETVAR
*"      SERIALIZATION_INFO STRUCTURE  BDI_SER
*"  EXCEPTIONS
*"      WRONG_FUNCTION_CALLED
*"----------------------------------------------------------------------
  INCLUDE mbdconwf.
  DATA : ls_seg TYPE zcntry_tax_seg,
         lt_cnty_tax TYPE TABLE OF zcntry_tax_code,
         ls_cnty_tax TYPE zcntry_tax_code.
  workflow_result = c_wf_result_ok.
```

**FUNCTION zcountyr_tax_code_update.**

***”—————————————————————————-**

***”*”Local Interface:**

***”  IMPORTING**

***”     REFERENCE(INPUT_METHOD) TYPE  BDWFAP_PAR-INPUTMETHD**

***”     REFERENCE(MASS_PROCESSING) TYPE  BDWFAP_PAR-MASS_PROC**

***”  EXPORTING**

***”     REFERENCE(WORKFLOW_RESULT) TYPE  BDWF_PARAM-RESULT**

***”     REFERENCE(APPLICATION_VARIABLE) TYPE  BDWF_PARAM-APPL_VAR**

***”     REFERENCE(IN_UPDATE_TASK) TYPE  BDWFAP_PAR-UPDATETASK**

***”     REFERENCE(CALL_TRANSACTION_DONE) TYPE  BDWFAP_PAR-CALLTRANS**

***”  TABLES**

***”      IDOC_CONTRL STRUCTURE  EDIDC**

***”      IDOC_DATA STRUCTURE  EDIDD**

***”      IDOC_STATUS STRUCTURE  BDIDOCSTAT**

***”      RETURN_VARIABLES STRUCTURE  BDWFRETVAR**

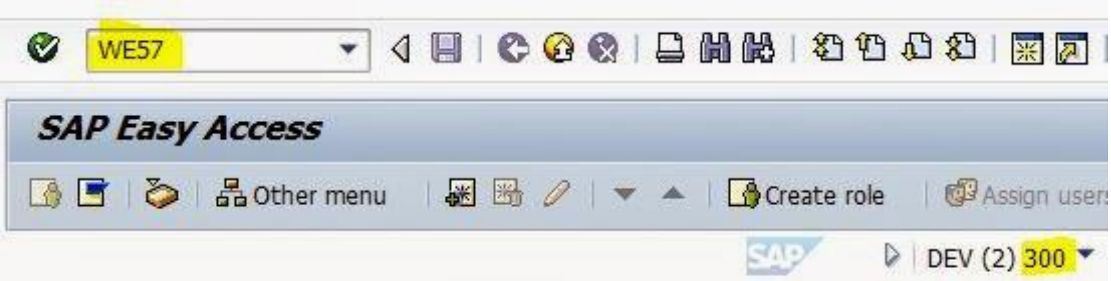
***”      SERIALIZATION_INFO STRUCTURE  BDI_SER**

***”  EXCEPTIONS**

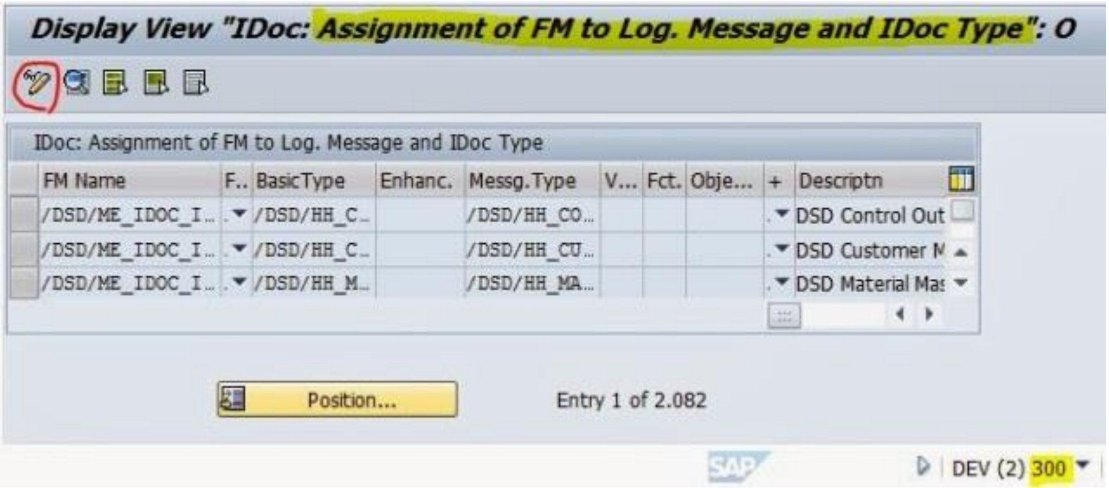
***”      WRONG_FUNCTION_CALLED**

```
*"——————————————————————————————-
 INCLUDE mbdconwf.
 DATA : ls_seg TYPE zcntry_tax_seg,
     lt_cnty_tax TYPE TABLE OF zcntry_tax_code,
     ls_cnty_tax TYPE zcntry_tax_code.
 workflow_result = c_wf_result_ok.
 LOOP AT idoc_contrl.
  IF idoc_contrl-mestyp = 'ZCNTRY_TAX_MSG'.
   LOOP AT  idoc_data.
    ls_seg = idoc_data-sdata.
    MOVE-CORRESPONDING ls_seg TO ls_cnty_tax.
    APPEND ls_cnty_tax TO lt_cnty_tax.
   ENDLOOP.
   MODIFY zcntry_tax_code FROM TABLE lt_cnty_tax.
   IF sy-subrc = 0.
    idoc_status-docnum = idoc_contrl-docnum.
    idoc_status-status = '53'.
    idoc_status-msgv1 =  ls_cnty_tax-countyr.
    APPEND idoc_status.
    CLEAR idoc_status.
   ELSE.
    idoc_status-docnum = idoc_contrl-docnum.
    idoc_status-status = '51'.
    idoc_status-msgv1 =  ls_cnty_tax-countyr.
    APPEND idoc_status.
    CLEAR idoc_status.
    workflow_result = c_wf_result_error.
    return_variables-wf_param = 'Error_Idocs'.
    return_variables-doc_number = idoc_contrl-docnum.
    APPEND return_variables.
    CLEAR return_variables.
    ENDIF.
   ENDIF.
  ENDLOOP.
ENDFUNCTION.
```

Step89. Now go to Tcode- WE57 to assign the Function module to the Idoc type.



Step90. Click on Edit button.



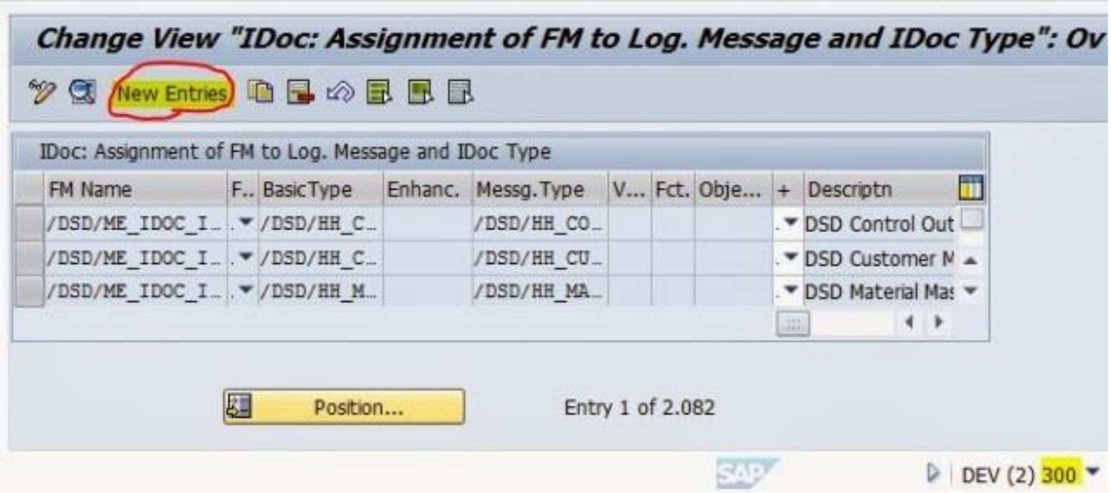
Display View "IDoc: Assignment of FM to Log. Message and IDoc Type": O

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

| FM Name | F.. | BasicType | Enhanc. | Messg.Type | V... | Fct. | Obje... | + | Descriptn |
|---|---|---|---|---|---|---|---|---|---|
| /DSD/ME_IDOC_I... | | /DSD/HH_C... | | /DSD/HH_CO... | | | | | DSD Control Out |
| /DSD/ME_IDOC_I... | | /DSD/HH_C... | | /DSD/HH_CU... | | | | | DSD Customer M |
| /DSD/ME_IDOC_I... | | /DSD/HH_M... | | /DSD/HH_MA... | | | | | DSD Material Mas |

Entry 1 of 2.082

Step91. Clcik on New Entries.



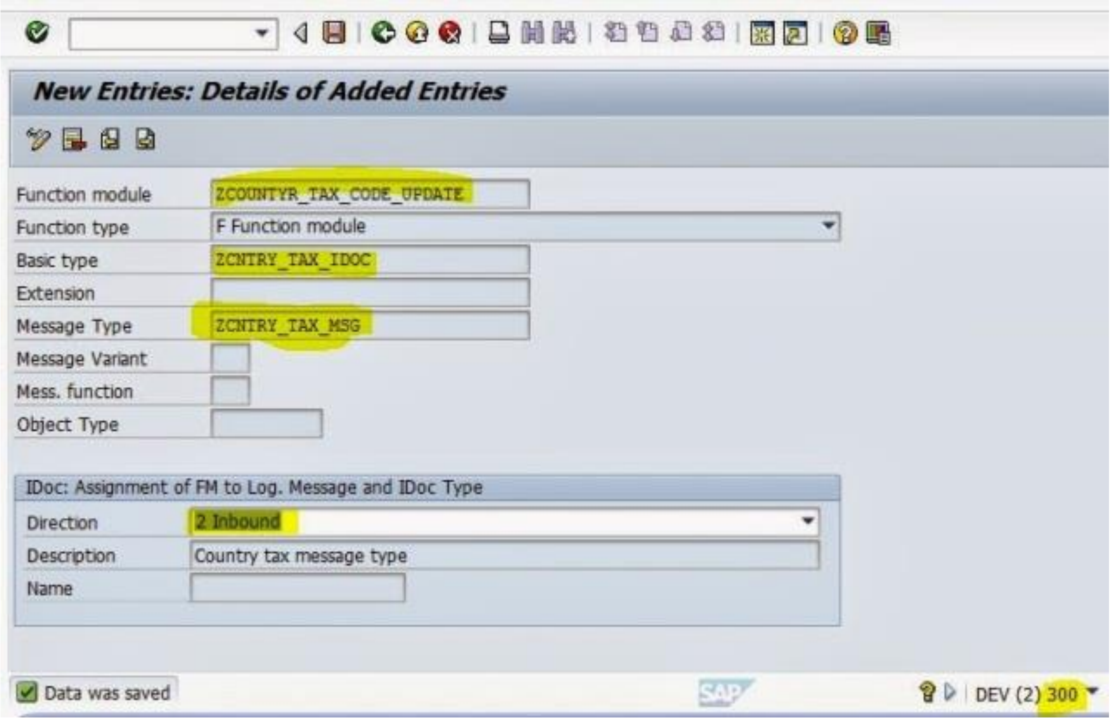
Change View "IDoc: Assignment of FM to Log. Message and IDoc Type": Ov

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

| FM Name | F.. | BasicType | Enhanc. | Messg.Type | V... | Fct. | Obje... | + | Descriptn |
|---|---|---|---|---|---|---|---|---|---|
| /DSD/ME_IDOC_I... | | /DSD/HH_C... | | /DSD/HH_CO... | | | | | DSD Control Out |
| /DSD/ME_IDOC_I... | | /DSD/HH_C... | | /DSD/HH_CU... | | | | | DSD Customer M |
| /DSD/ME_IDOC_I... | | /DSD/HH_M... | | /DSD/HH_MA... | | | | | DSD Material Mas |

Entry 1 of 2.082

Step92. Provide the Function module name, Idoc name and message type. Save it and clcik on back button.



New Entries: Details of Added Entries

| | |
|---|---|
| Function module | ZCOUNTYR_TAX_CODE_UPDATE |
| Function type | F Function module |
| Basic type | ZCNTRY_TAX_IDOC |
| Extension | |
| Message Type | ZCNTRY_TAX_MSG |
| Message Variant | |
| Mess. function | |
| Object Type | |

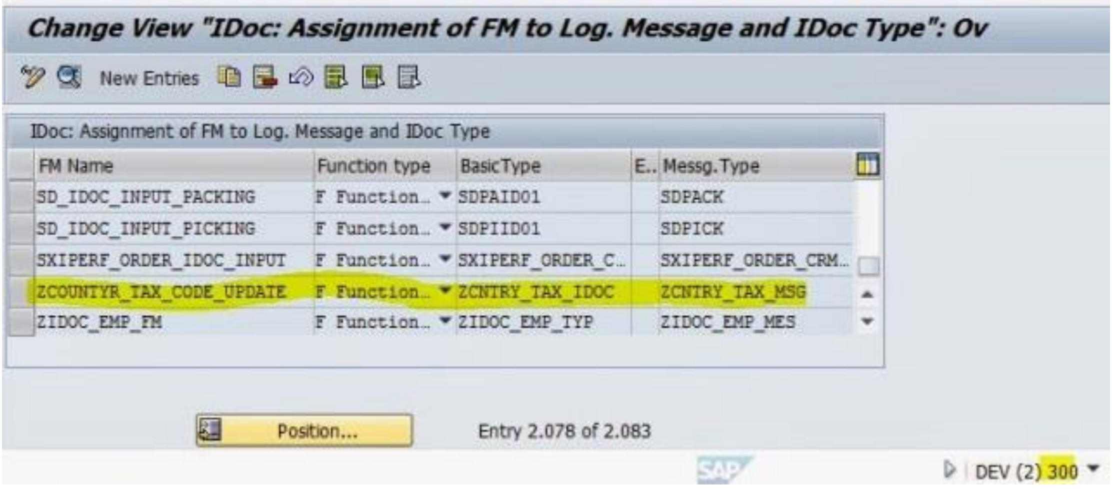
IDoc: Assignment of FM to Log. Message and IDoc Type

| | |
|---|---|
| Direction | 2 Inbound |
| Description | Country tax message type |
| Name | |

Data was saved

DEV (2) 300

Step93. The line appears in the list.



Change View "IDoc: Assignment of FM to Log. Message and IDoc Type": Ov

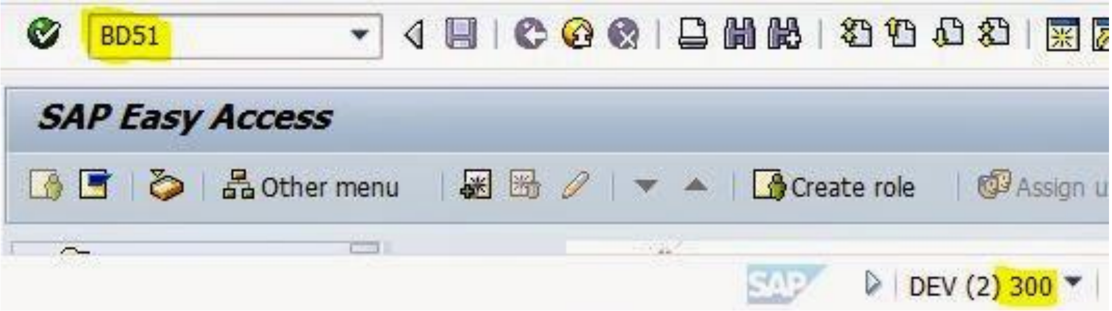
IDoc: Assignment of FM to Log. Message and IDoc Type

| FM Name | Function type | BasicType | E.. | Messg.Type |
|---|---|---|---|---|
| SD_IDOC_INPUT_PACKING | F Function... | SDPAID01 | | SDPACK |
| SD_IDOC_INPUT_PICKING | F Function... | SDPIID01 | | SDPICK |
| SXIPERF_ORDER_IDOC_INPUT | F Function... | SXIPERF_ORDER_C... | | SXIPERF_ORDER_CRM... |
| ZCOUNTYR_TAX_CODE_UPDATE | F Function... | ZCNTRY_TAX_IDOC | | ZCNTRY_TAX_MSG |
| ZIDOC_EMP_FM | F Function... | ZIDOC_EMP_TYP | | ZIDOC_EMP_MES |

Entry 2.078 of 2.083

DEV (2) 300

Step94. Now go to Tcode- BD51 .

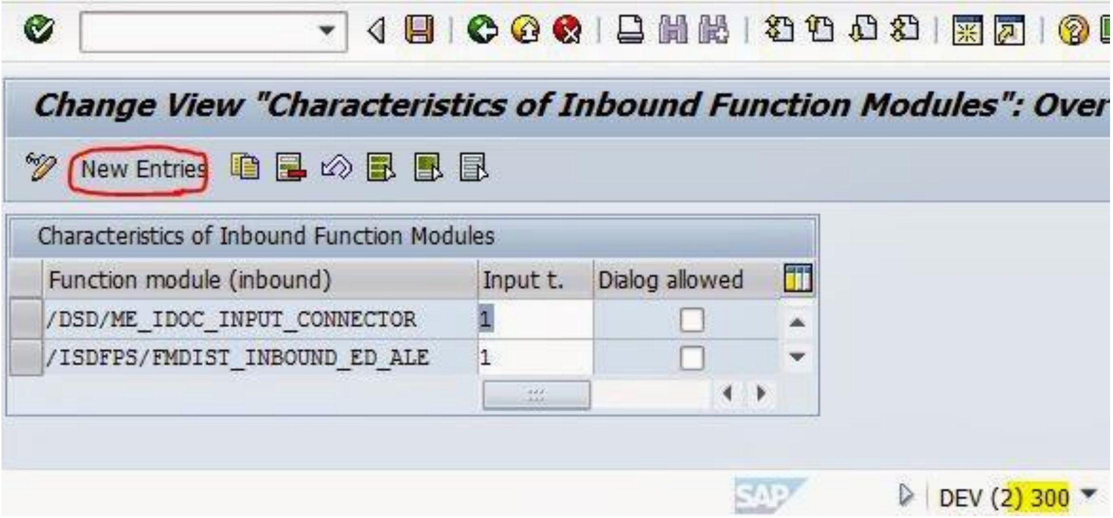


BD51

SAP Easy Access

DEV (2) 300

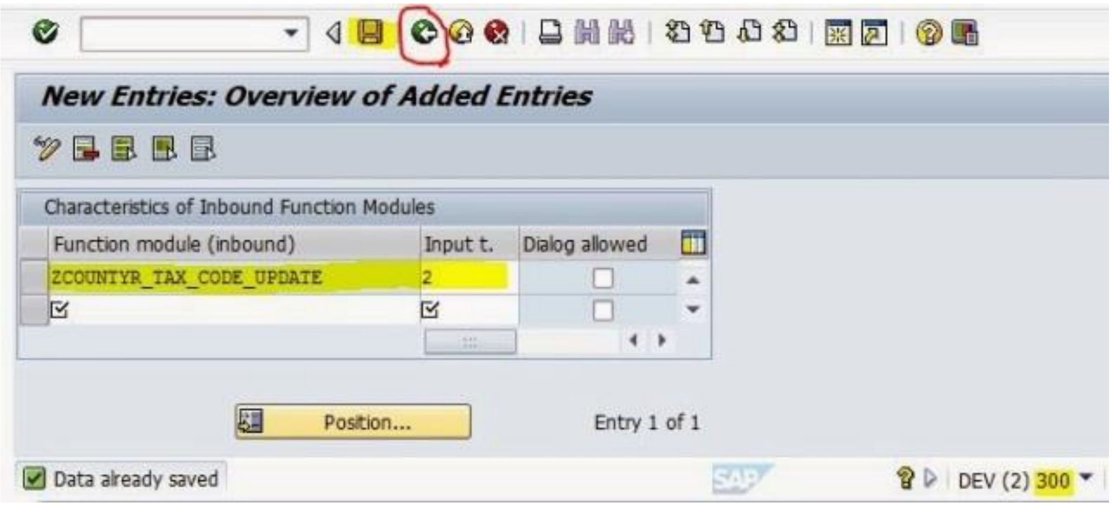
Step95. click On New Entries.


Change View "Characteristics of Inbound Function Modules": Over

Characteristics of Inbound Function Modules

| Function module (inbound) | Input t. | Dialog allowed |
|---|---|---|
| /DSD/ME_IDOC_INPUT_CONNECTOR | 1 | □ |
| /ISDFPS/FMDIST_INBOUND_ED_ALE | 1 | □ |

Step96. Provide the Function Module name and maintain value as 2 under input section as a inbound function module.


New Entries: Overview of Added Entries

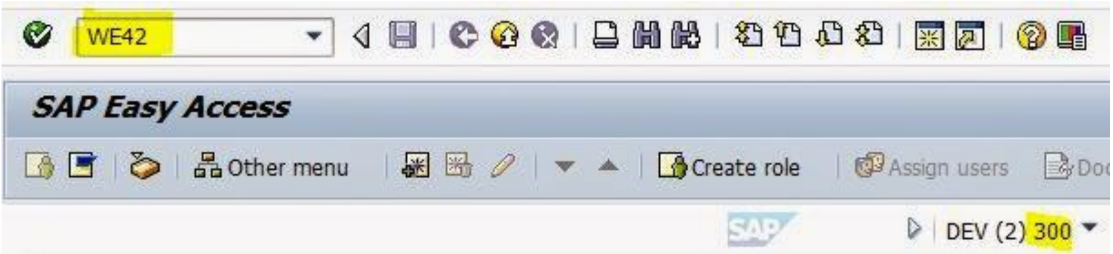
Characteristics of Inbound Function Modules

| Function module (inbound) | Input t. | Dialog allowed |
|---|---|---|
| ZCOUNTYR_TAX_CODE_UPDATE | 2 | □ |
| ☑ | ☑ | □ |

Entry 1 of 1

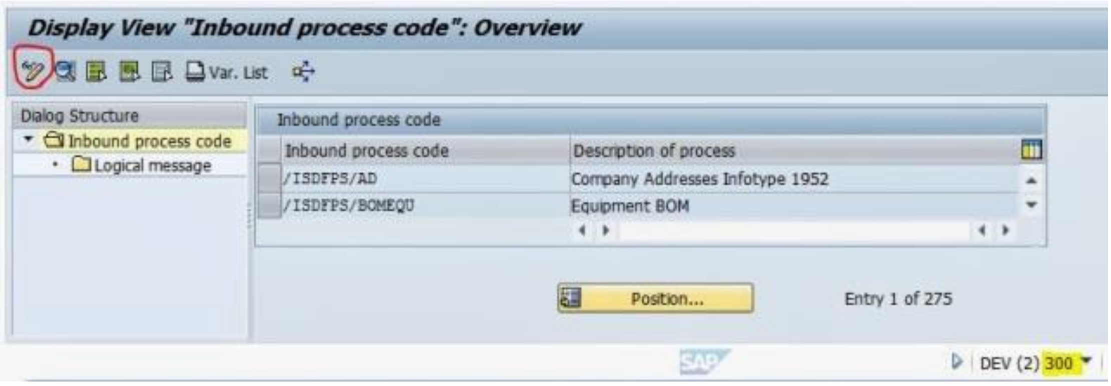
Data already saved

Step97. Now go to Tcode- WE42.


WE42

SAP Easy Access

Step98. Click on Edit button.



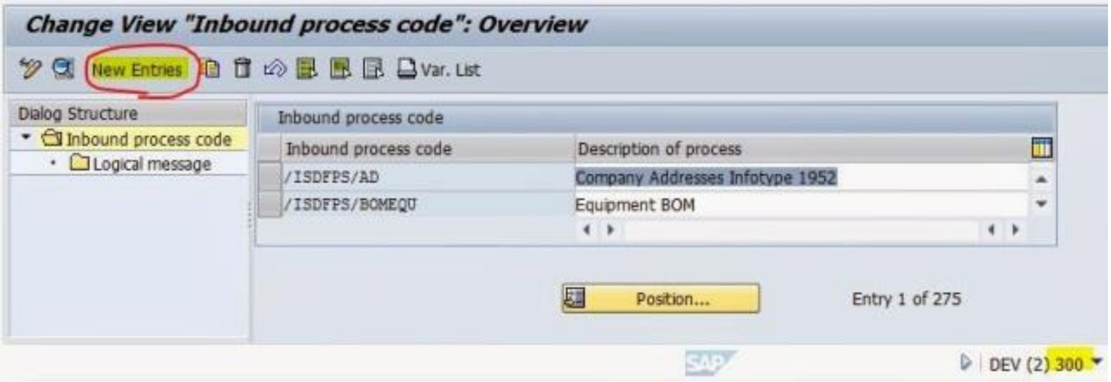
Display View "Inbound process code": Overview

Inbound process code

| Inbound process code | Description of process |
|---|---|
| /ISDFPS/AD | Company Addresses Infotype 1952 |
| /ISDFPS/BOMEQU | Equipment BOM |

Entry 1 of 275

**Step99. Click on New Entries button.**



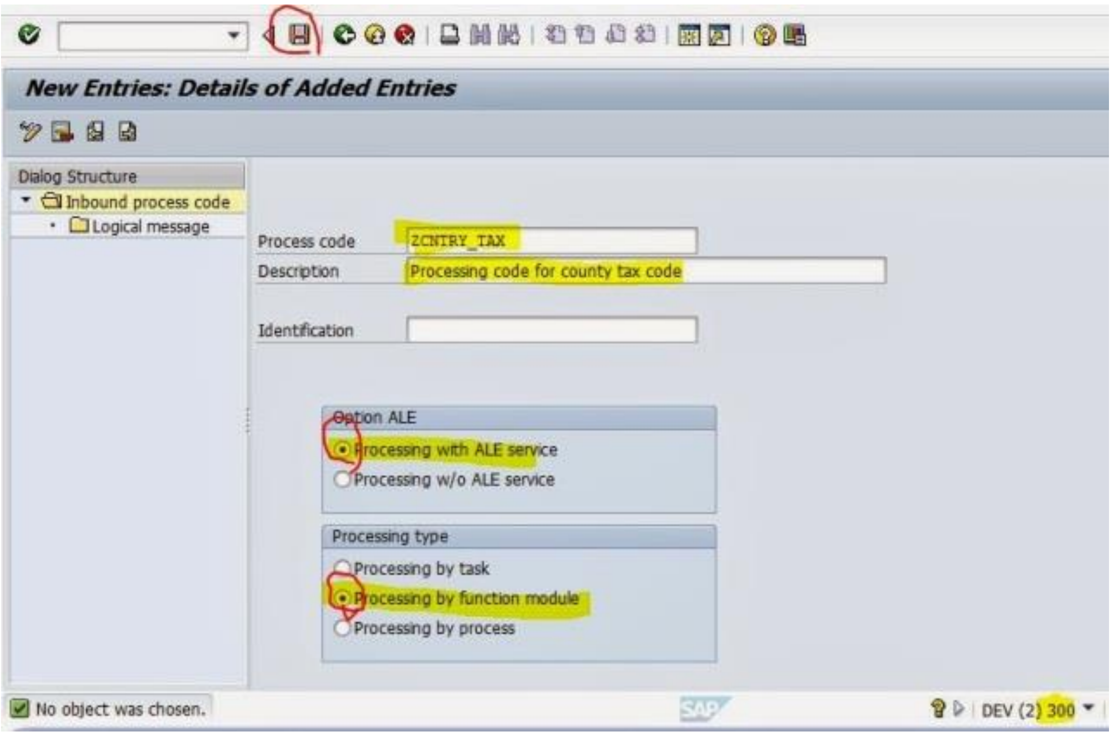
Change View "Inbound process code": Overview

Inbound process code

| Inbound process code | Description of process |
|---|---|
| /ISDFPS/AD | Company Addresses Infotype 1952 |
| /ISDFPS/BOMEQU | Equipment BOM |

Entry 1 of 275

**Step100. Provide a Processing Code name with description. Select the radio buttons as highlighted below. Save it and click on back button.**



New Entries: Details of Added Entries

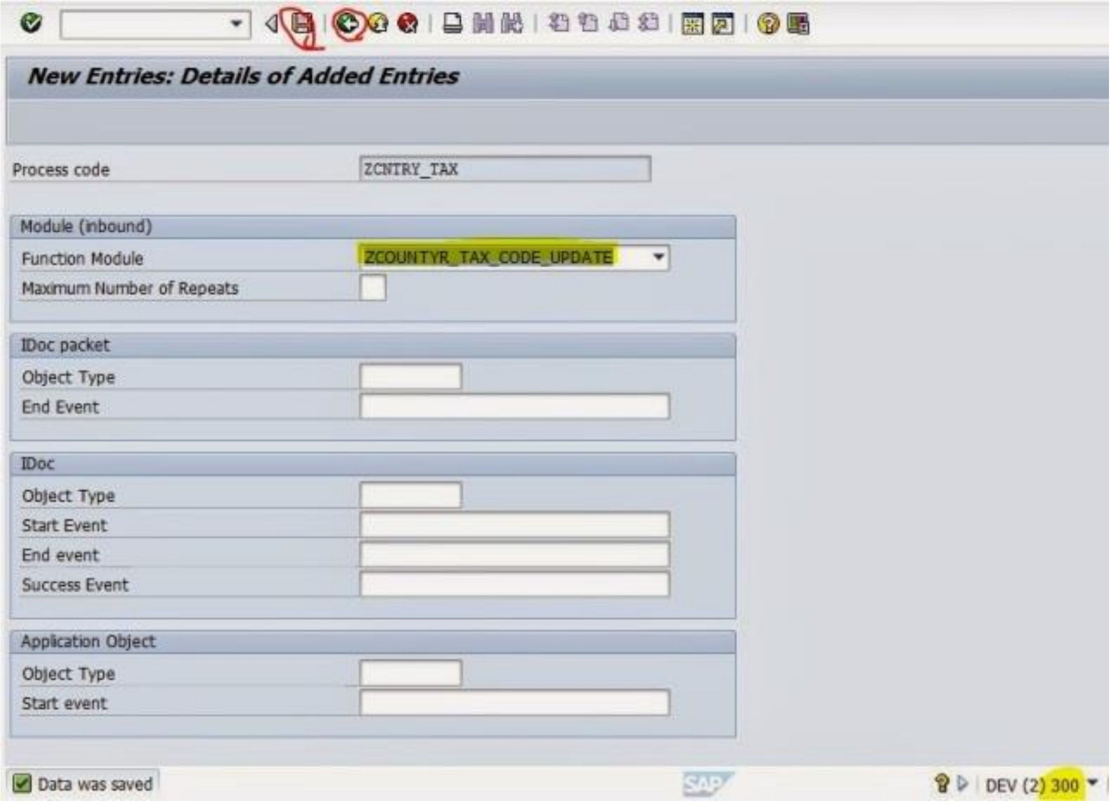
Process code: ZCNTRY_TAX
Description: Processing code for county tax code
Identification:

Option ALE
● Processing with ALE service
○ Processing w/o ALE service

Processing type
○ Processing by task
● Processing by function module
○ Processing by process

No object was chosen.

**Step101. Here provide the Inbound function module name, save it and click on Back button.**



New Entries: Details of Added Entries

Process code: ZCNTRY_TAX

Module (inbound)

Function Module: ZCOUNTYR_TAX_CODE_UPDATE

Maximum Number of Repeats: □

IDoc packet

Object Type:

End Event:

IDoc

Object Type:

Start Event:

End event:

Success Event:

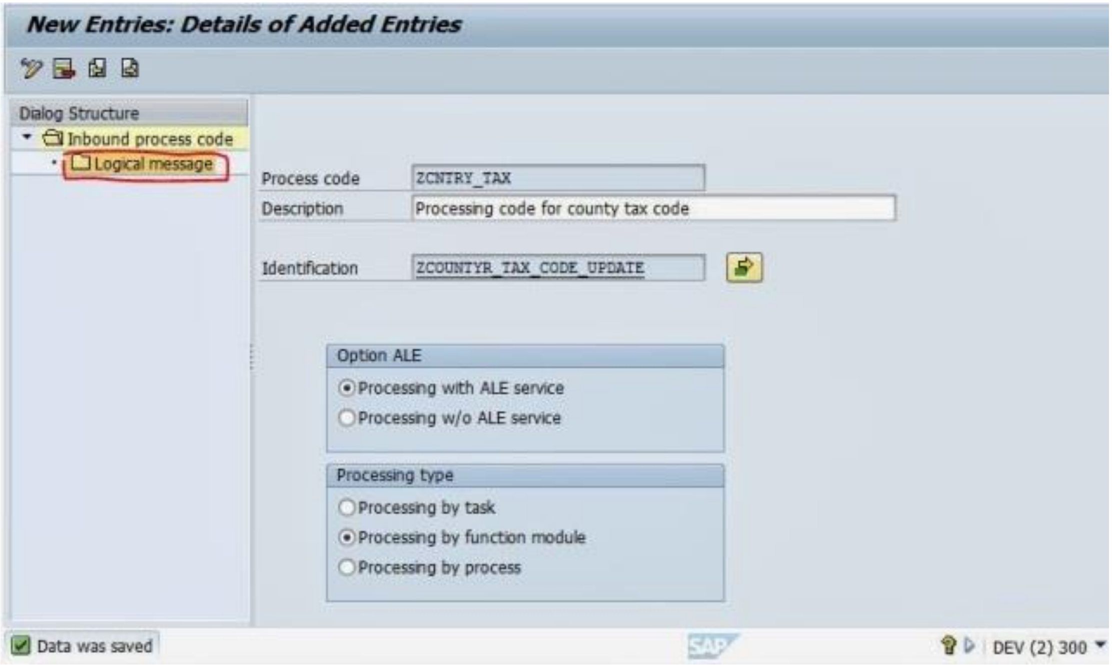
Application Object

Object Type:

Start event:

☑ Data was saved

**Step102. Now double click on the logical message  line highlighted in the left hand side.**



New Entries: Details of Added Entries

Dialog Structure

- Inbound process code
  - Logical message

Process code: ZCNTRY_TAX

Description: Processing code for county tax code

Identification: ZCOUNTYR_TAX_CODE_UPDATE

Option ALE

● Processing with ALE service

○ Processing w/o ALE service

Processing type

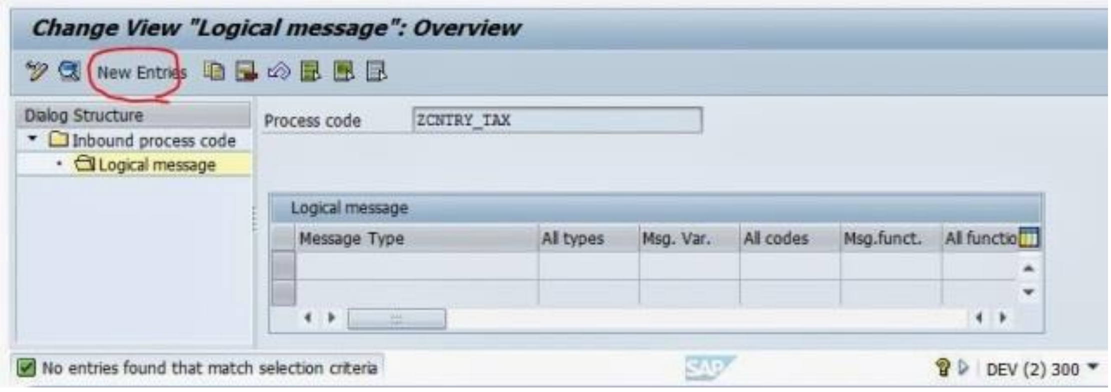
○ Processing by task

● Processing by function module

○ Processing by process

☑ Data was saved

**Step103. Click on New Entries button.**



Change View "Logical message": Overview

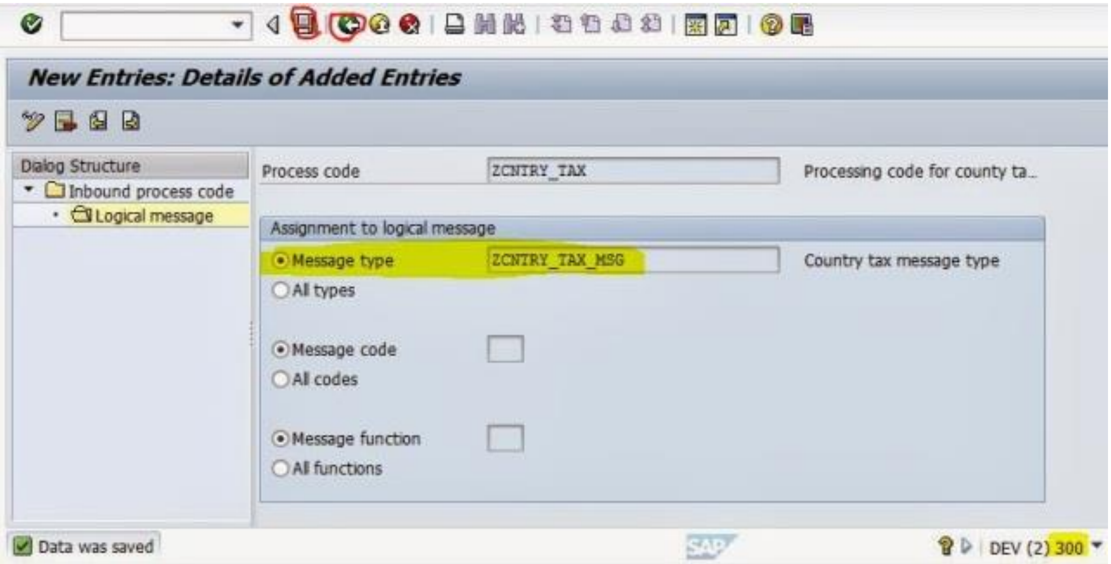
Process code ZCNTRY_TAX

| Message Type | All types | Msg. Var. | All codes | Msg.funct. | All functio |
|---|---|---|---|---|---|

No entries found that match selection criteria

DEV (2) 300

Step104. Provide the message type name, save it and click on the Back button.



New Entries: Details of Added Entries

Process code ZCNTRY_TAX Processing code for county ta...

Assignment to logical message

● Message type ZCNTRY_TAX_MSG Country tax message type
○ All types

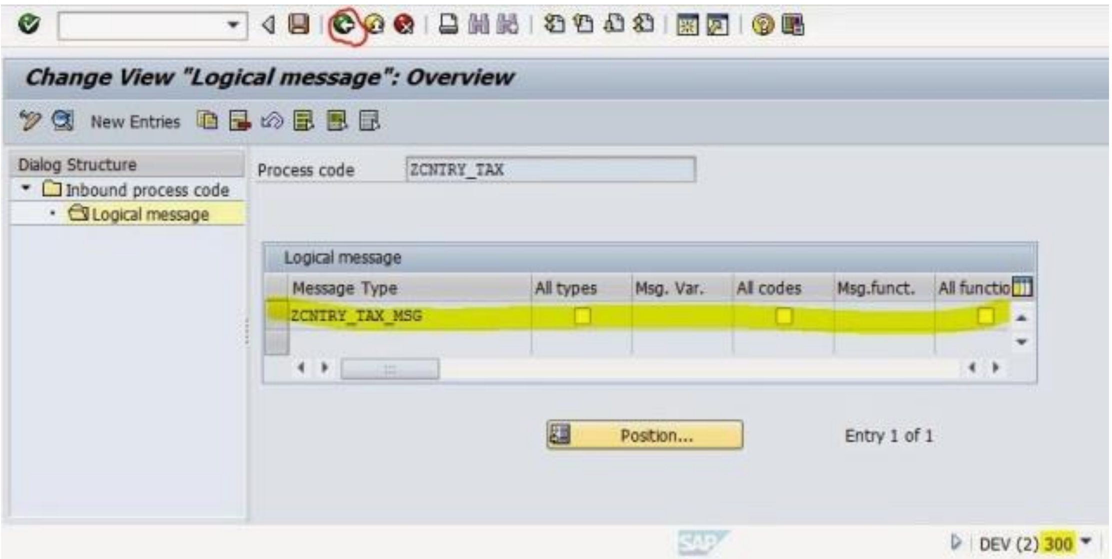
● Message code □
○ All codes

● Message function □
○ All functions

Data was saved

DEV (2) 300

Step105. Click on the Back button.



Change View "Logical message": Overview

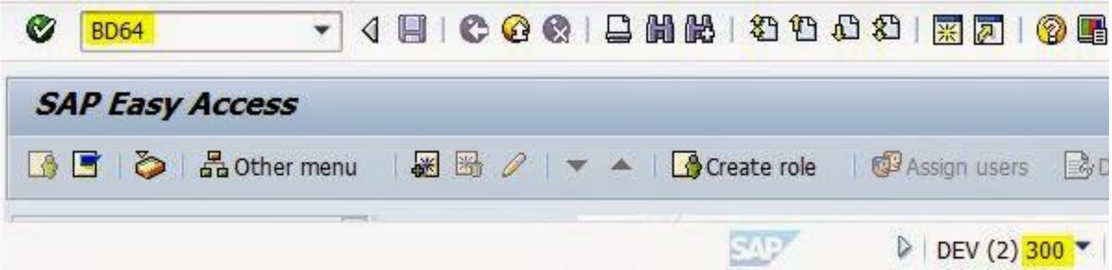
Process code ZCNTRY_TAX

| Message Type | All types | Msg. Var. | All codes | Msg.funct. | All functio |
|---|---|---|---|---|---|
| ZCNTRY_TAX_MSG | □ | | □ | | □ |

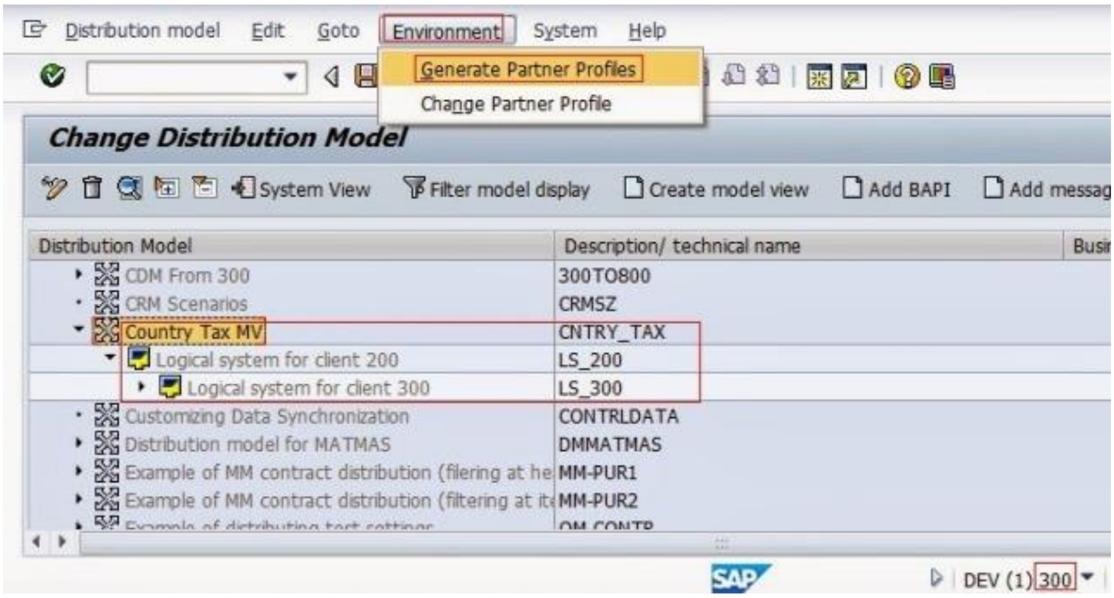
Entry 1 of 1

DEV (2) 300

Step106. Now go to Tcode- BD64 to generate the partner profile from the Customer distribution model.

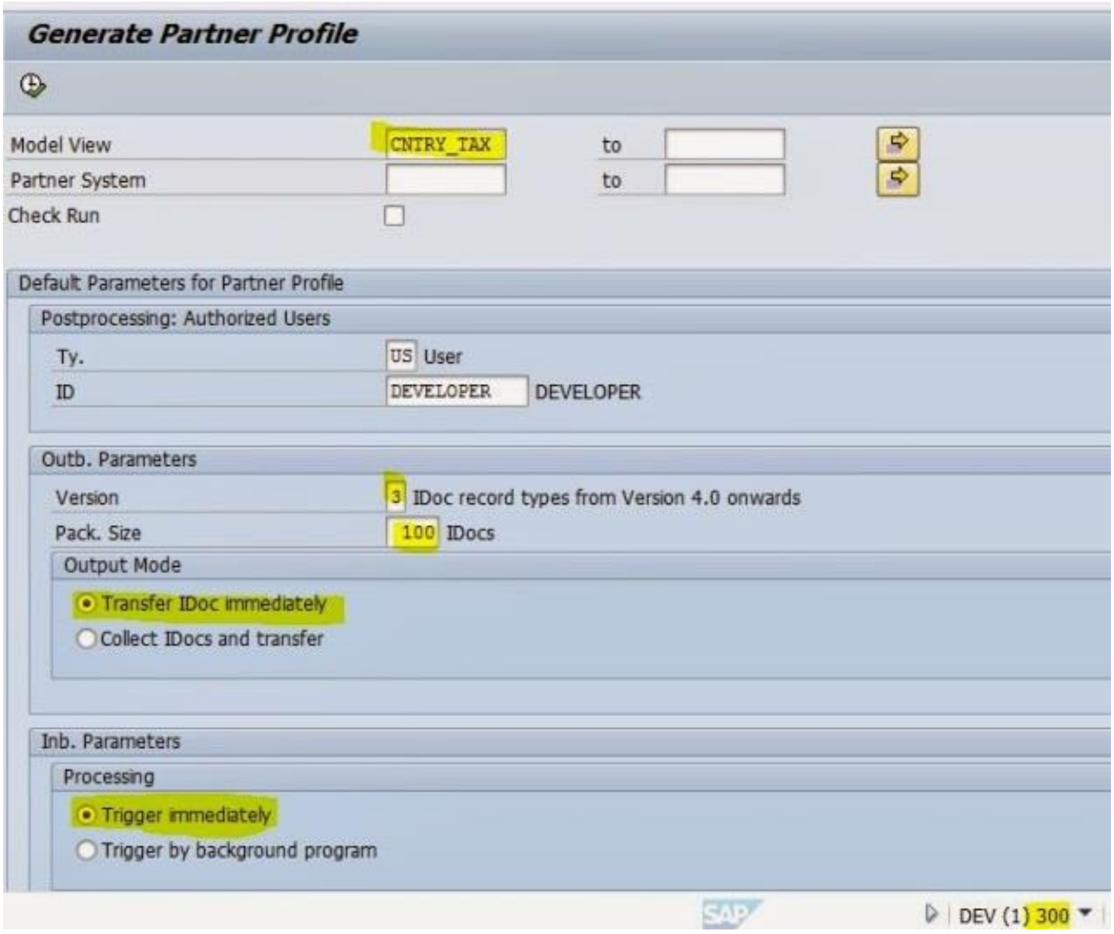


**Step107. Select the Distribution model, From the Environment menu click on Generate Partner profile .**



| Distribution Model | Description/ technical name | Busi |
|---|---|---|
| CDM From 300 | 300TO800 | |
| CRM Scenarios | CRMSZ | |
| Country Tax MV | CNTRY_TAX | |
| Logical system for client 200 | LS_200 | |
| Logical system for client 300 | LS_300 | |
| Customizing Data Synchronization | CONTRLDATA | |
| Distribution model for MATMAS | DMMATMAS | |
| Example of MM contract distribution (filering at he | MM-PUR1 | |
| Example of MM contract distribution (filtering at it | MM-PUR2 | |
| Example of distributing text settings | OM CONTR | |

**Step108. Execute the program.**



Generate Partner Profile

|  |  |  |  |
|---|---|---|---|
| Model View | CNTRY_TAX | to |  |
| Partner System |  | to |  |
| Check Run | □ |  |  |

Default Parameters for Partner Profile

Postprocessing: Authorized Users

|  |  |  |
|---|---|---|
| Ty. | US | User |
| ID | DEVELOPER | DEVELOPER |

Outb. Parameters

|  |  |  |
|---|---|---|
| Version | 3 | IDoc record types from Version 4.0 onwards |
| Pack. Size | 100 | IDocs |

Output Mode

● Transfer IDoc immediately

○ Collect IDocs and transfer

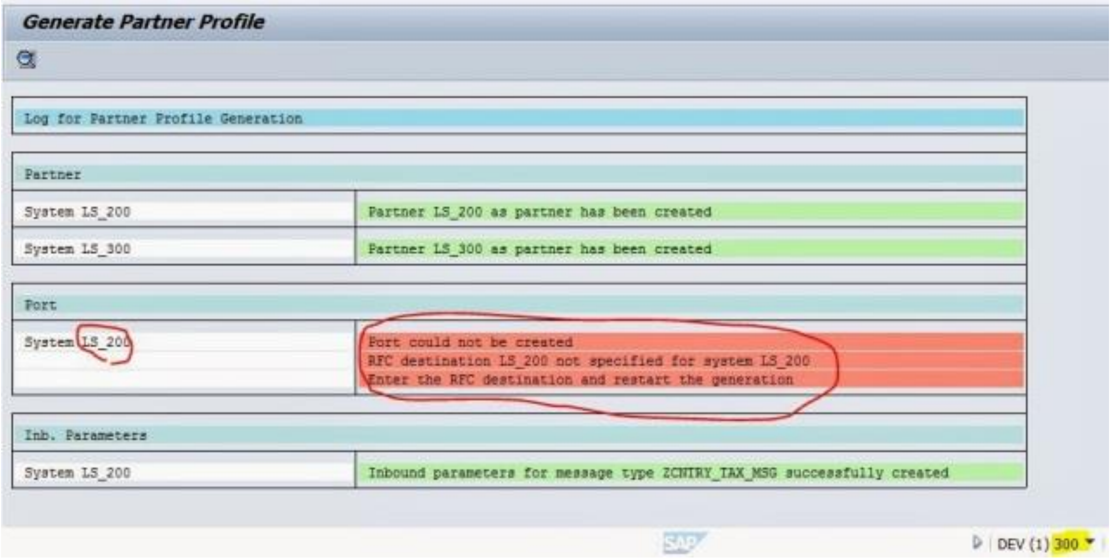
Inb. Parameters

Processing

● Trigger immediately

○ Trigger by background program

**Step109. The partner profiles LS_200 and Ls_300 are generated.**



Generate Partner Profile

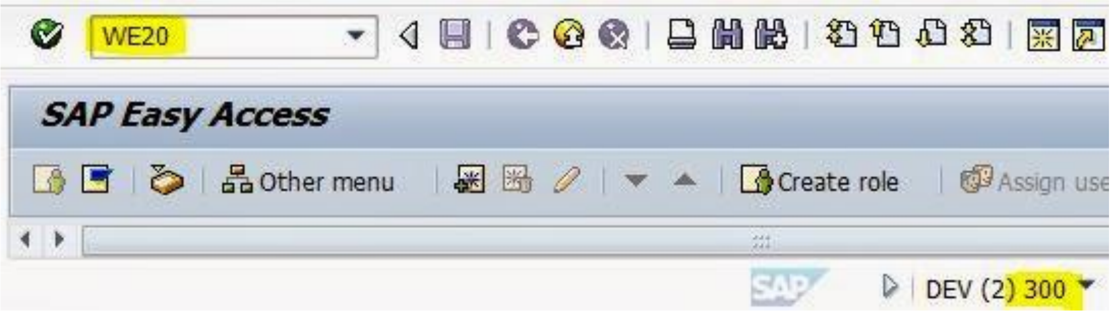
Log for Partner Profile Generation

| Partner |  |
|---|---|
| System LS_200 | Partner LS_200 as partner has been created |
| System LS_300 | Partner LS_300 as partner has been created |

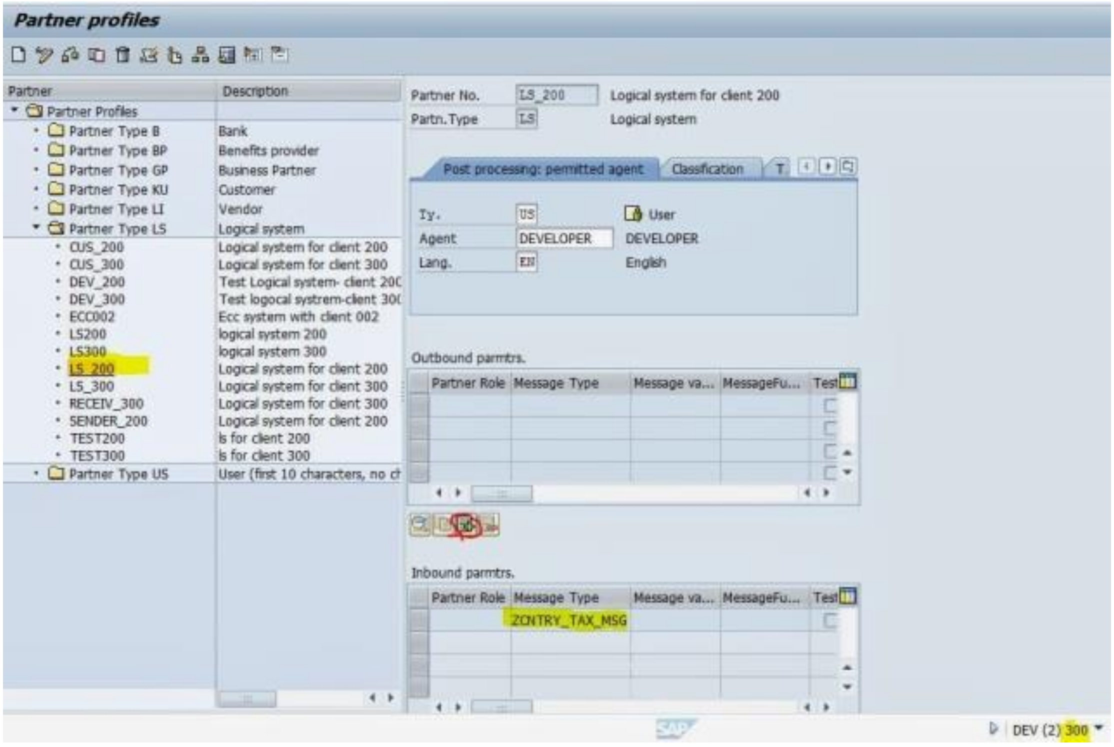
| Port |  |
|---|---|
| System LS_200 | Port could not be created<br>RFC destination LS_200 not specified for system LS_200<br>Enter the RFC destination and restart the generation |

| Inb. Parameters |  |
|---|---|
| System LS_200 | Inbound parameters for message type ZCNTRY_TAX_MSG successfully created |

**Step110. Now go to Tcode- WE20 to see the partner profiles.**



**Step111. Under the Partner Type 'LS' select the partner profile LS_200. The message type ZCNTRY_TAX_MSG is already added in the Inbound parameters section. Now click on the Add button under the Outbound parameters section.**



| 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 |
| CUS_200 | Logical system for client 200 |
| CUS_300 | Logical system for client 300 |
| DEV_200 | Test Logical system- client 20C |
| DEV_300 | Test logocal systrem-client 30C |
| ECC002 | Ecc system with client 002 |
| LS200 | logical system 200 |
| LS300 | logical system 300 |
| LS_200 | Logical system for client 200 |
| LS_300 | Logical system for client 300 |
| RECEIV_300 | Logical system for client 300 |
| SENDER_200 | Logical system for client 200 |
| TEST200 | ls for client 200 |
| TEST300 | ls for client 300 |
| Partner Type US | User (first 10 characters, no ch |

Partner No. LS_200 Logical system for client 200
Partn.Type LS Logical system

Ty. US User
Agent DEVELOPER DEVELOPER
Lang. EN English

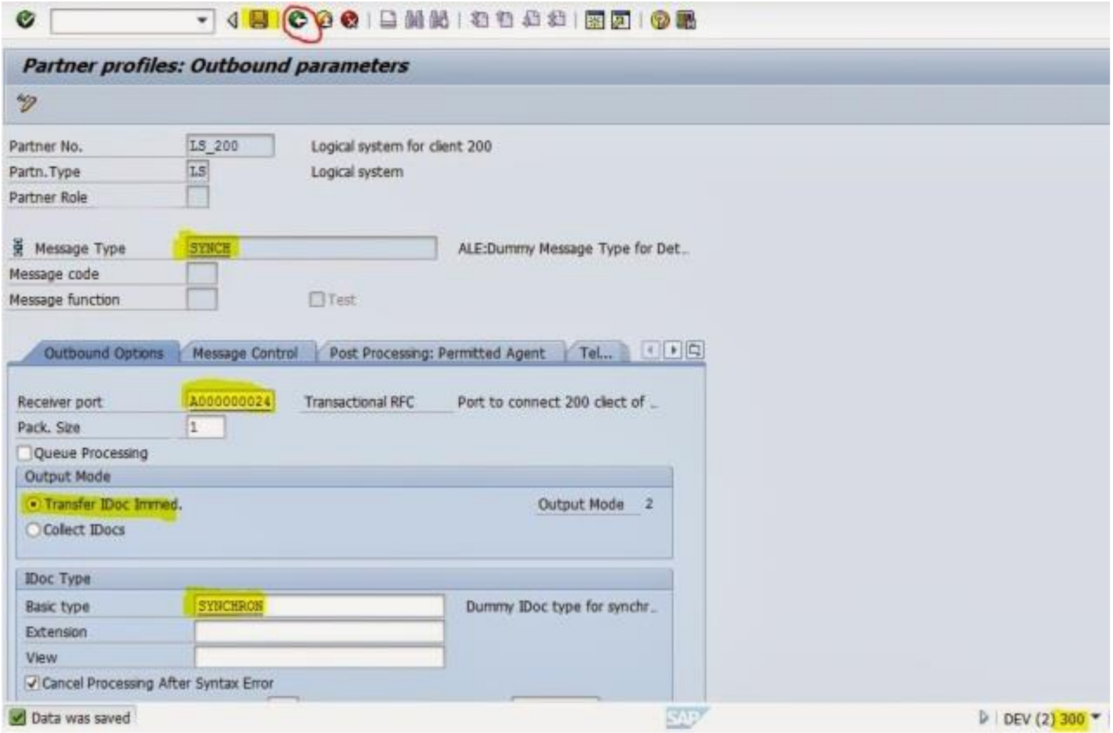
Outbound parmtrs.

| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|
| | | | | □ |
| | | | | □ |
| | | | | □ |
| | | | | □ |

Inbound parmtrs.

| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|
| | ZCNTRY_TAX_MSG | | | □ |

**Step112. Provide the message type, port, basic idoc type, save it and at last click on back button.**



**Partner profiles: Outbound parameters**

| | | |
|---|---|---|
| Partner No. | LS_200 | Logical system for client 200 |
| Partn.Type | LS | Logical system |
| Partner Role | | |
| Message Type | SYNCH | ALE:Dummy Message Type for Det.. |
| Message code | | |
| Message function | | □ Test |

Receiver port A000000024 Transactional RFC Port to connect 200 clect of ..

Pack. Size 1

□ Queue Processing

Output Mode

● Transfer IDoc Immed. Output Mode 2

○ Collect IDocs

IDoc Type

Basic type SYNCHRON Dummy IDoc type for synchr..

Extension

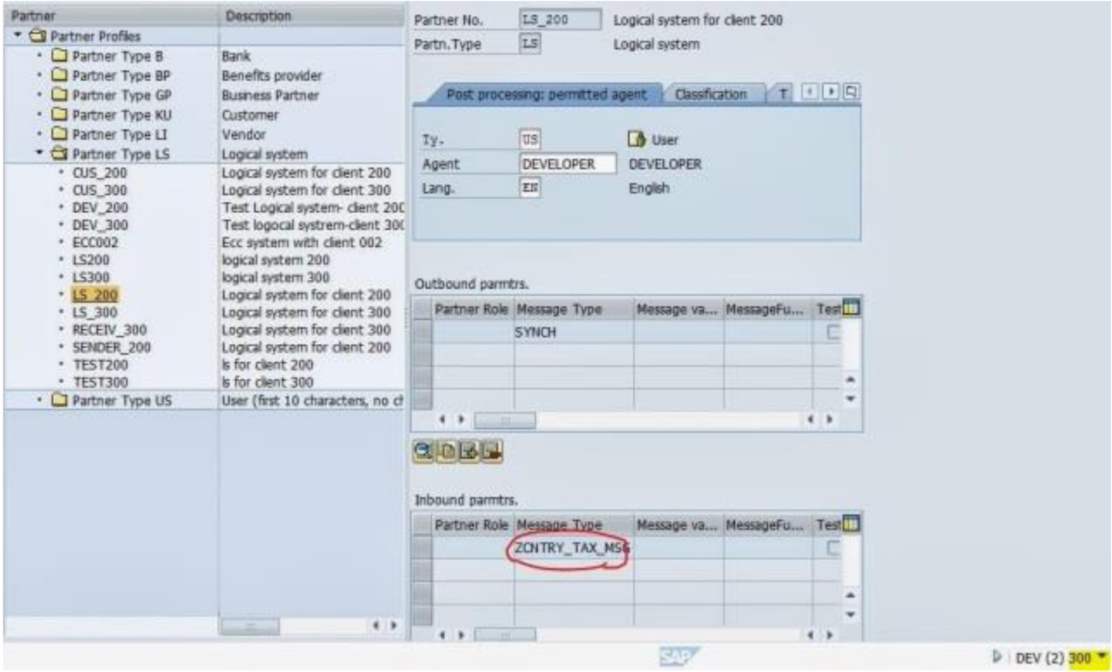
View

☑ Cancel Processing After Syntax Error

Data was saved

DEV (2) 300

**Step113. Now double click on the Message type ZCNTRY_TAX_MSG under the Inbound parameters.**



| 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 |
| CUS_200 | Logical system for client 200 |
| CUS_300 | Logical system for client 300 |
| DEV_200 | Test Logical system- client 200 |
| DEV_300 | Test logocal systrem-client 300 |
| ECC002 | Ecc system with client 002 |
| LS200 | logical system 200 |
| LS300 | logical system 300 |
| LS_200 | Logical system for client 200 |
| LS_300 | Logical system for client 300 |
| RECEIV_300 | Logical system for client 300 |
| SENDER_200 | Logical system for client 200 |
| TEST200 | ls for client 200 |
| TEST300 | ls for client 300 |
| Partner Type US | User (first 10 characters, no ch |

| | | |
|---|---|---|
| Partner No. | LS_200 | Logical system for client 200 |
| Partn.Type | LS | Logical system |

| | | |
|---|---|---|
| Ty. | US | User |
| Agent | DEVELOPER | DEVELOPER |
| Lang. | EN | English |

Outbound parmtrs.

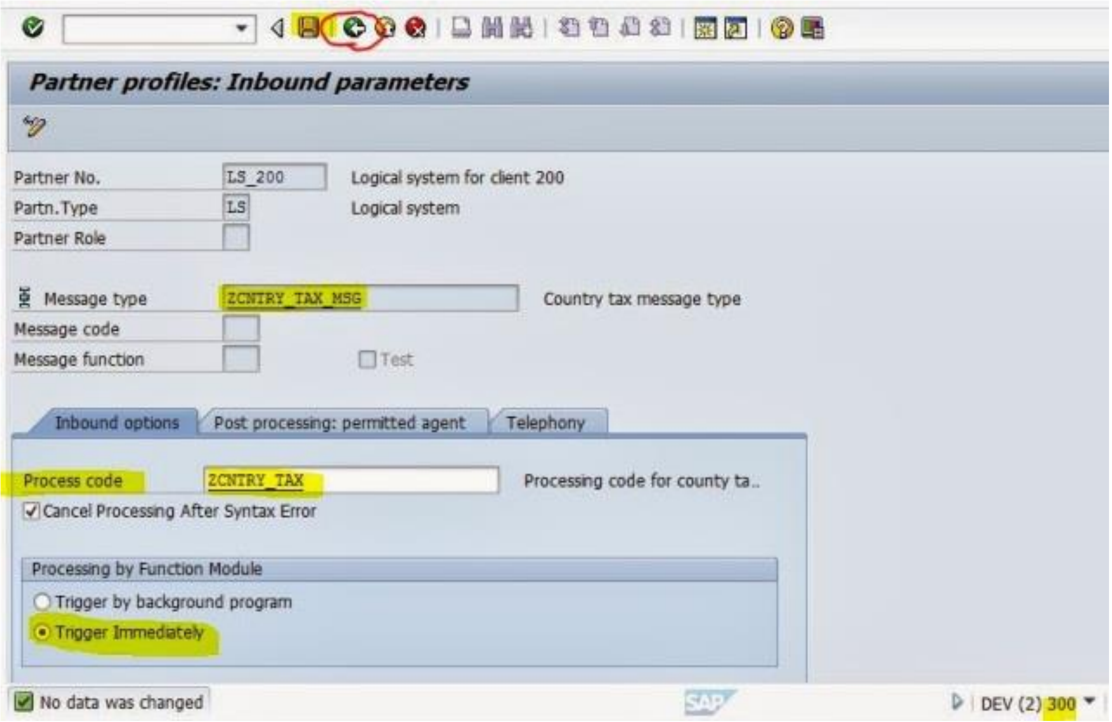
| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|
| | SYNCH | | | □ |

Inbound parmtrs.

| Partner Role | Message Type | Message va... | MessageFu... | Test |
|---|---|---|---|---|
| | ZCNTRY_TAX_MSG | | | □ |

DEV (2) 300

**Step114. The generated processing code is assigned to the message type.**



**Partner profiles: Inbound parameters**

Partner No. LS_200 Logical system for client 200

Partn.Type LS Logical system

Partner Role

Message type ZCNTRY_TAX_MSG Country tax message type

Message code

Message function

□ Test

Inbound options | Post processing: permitted agent | Telephony

Process code ZCNTRY_TAX Processing code for county ta..

☑ Cancel Processing After Syntax Error

Processing by Function Module

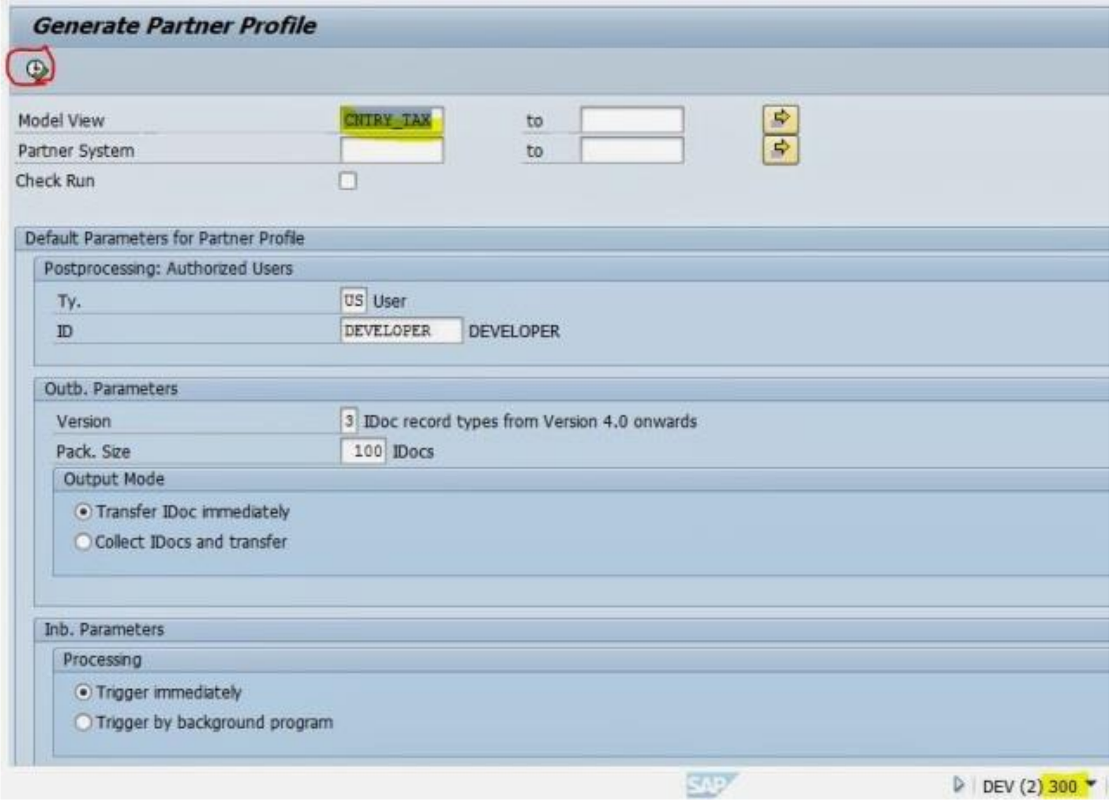
○ Trigger by background program

● Trigger Immediately

No data was changed

DEV (2) 300

**Step115. Now go to Tcode- BD64 to generate the partner profile from the Customer distribution model. Select the Distribution model, From the Environment menu click on Generate Partner profile . Execute the program.**



**Generate Partner Profile**

Model View CNTRY_TAX to

Partner System to

Check Run □

Default Parameters for Partner Profile

Postprocessing: Authorized Users

Ty. US User

ID DEVELOPER DEVELOPER

Outb. Parameters

Version 3 IDoc record types from Version 4.0 onwards

Pack. Size 100 IDocs

Output Mode

● Transfer IDoc immediately

○ Collect IDocs and transfer

Inb. Parameters

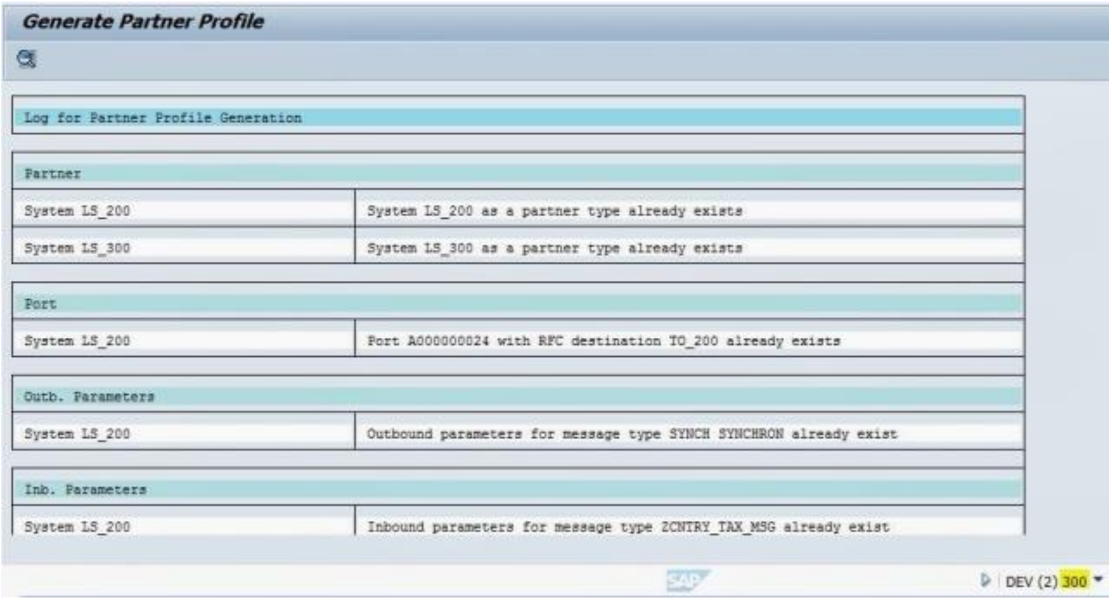
Processing

● Trigger immediately

○ Trigger by background program

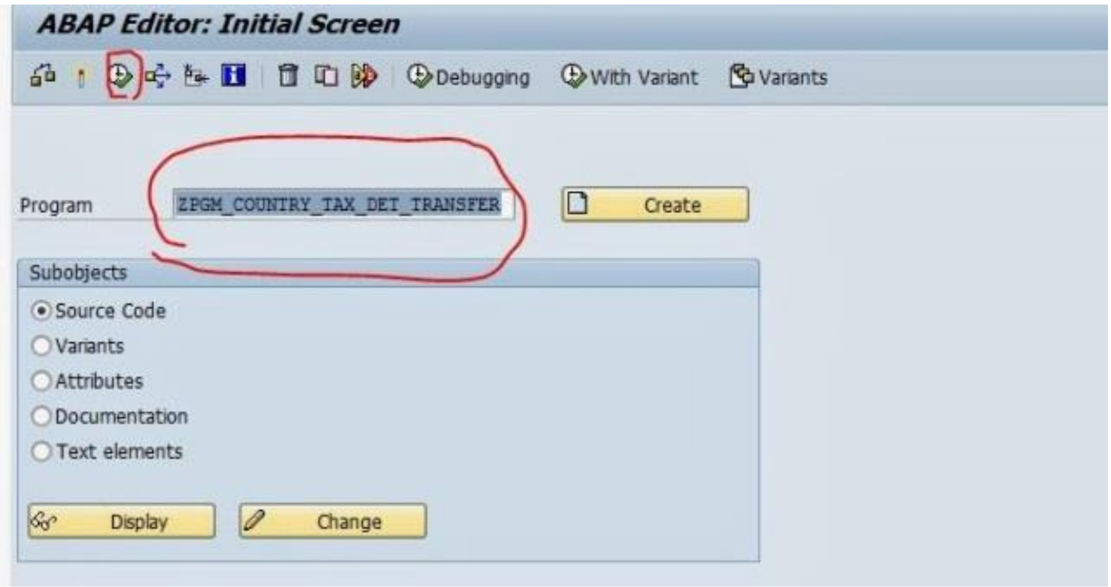
DEV (2) 300

**Step116. There are no red lines. So all set up is done in the client 300 to receive the Idoc.**



```
Log for Partner Profile Generation

Partner
System LS_200          System LS_200 as a partner type already exists
System LS_300          System LS_300 as a partner type already exists

Port
System LS_200          Port A000000024 with RFC destination TO_200 already exists

Outb. Parameters
System LS_200          Outbound parameters for message type SYNCH SYNCHRON already exist

Inb. Parameters
System LS_200          Inbound parameters for message type ZCNTRY_TAX_MSG already exist
```

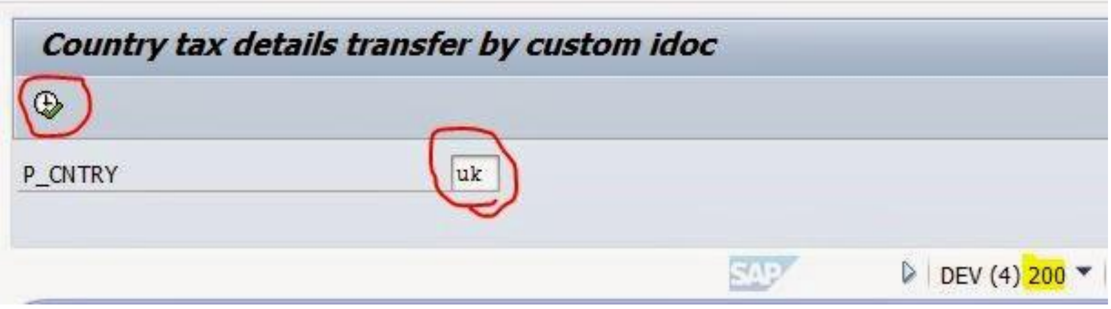
**Step117. Now go the client-200 and execute the report.**



Program ZPGM_COUNTRY_TAX_DET_TRANSFER

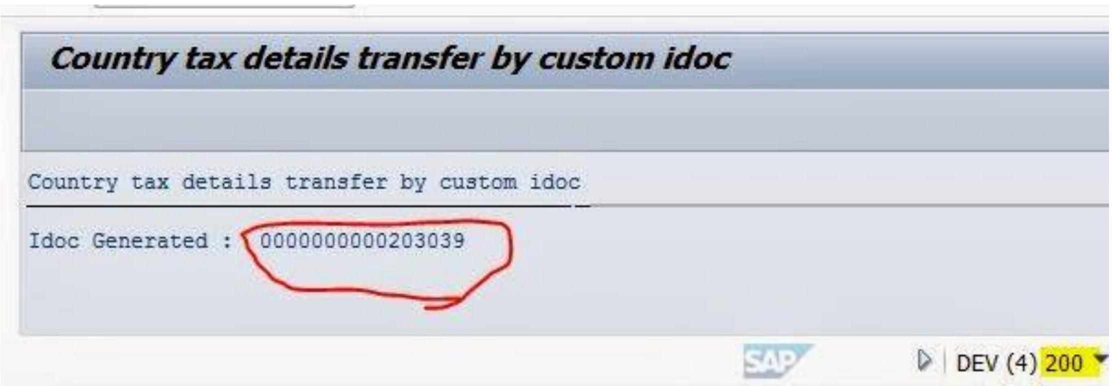
● Source Code
○ Variants
○ Attributes
○ Documentation
○ Text elements

**Step118. Provide the country name and execute it.**



P_CNTRY uk

**Step119. The idoc generated 203039.**

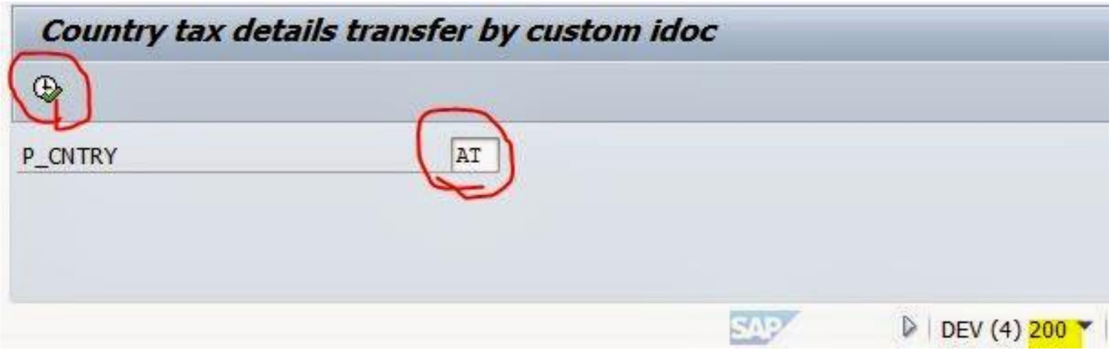


Country tax details transfer by custom idoc

Country tax details transfer by custom idoc

Idoc Generated : 0000000000203039

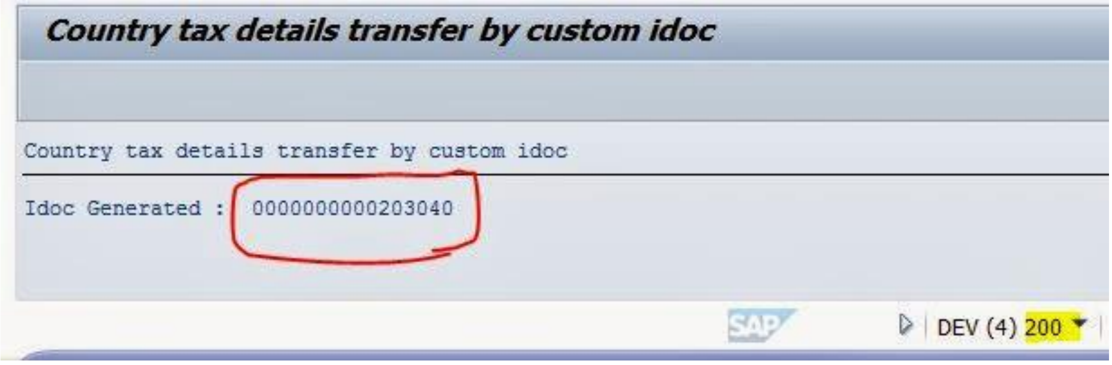
**Step120. Again execute the report and Provide the country name and execute it.**



Country tax details transfer by custom idoc

P_CNTRY AT

**Step121. The idoc generated 203040.**

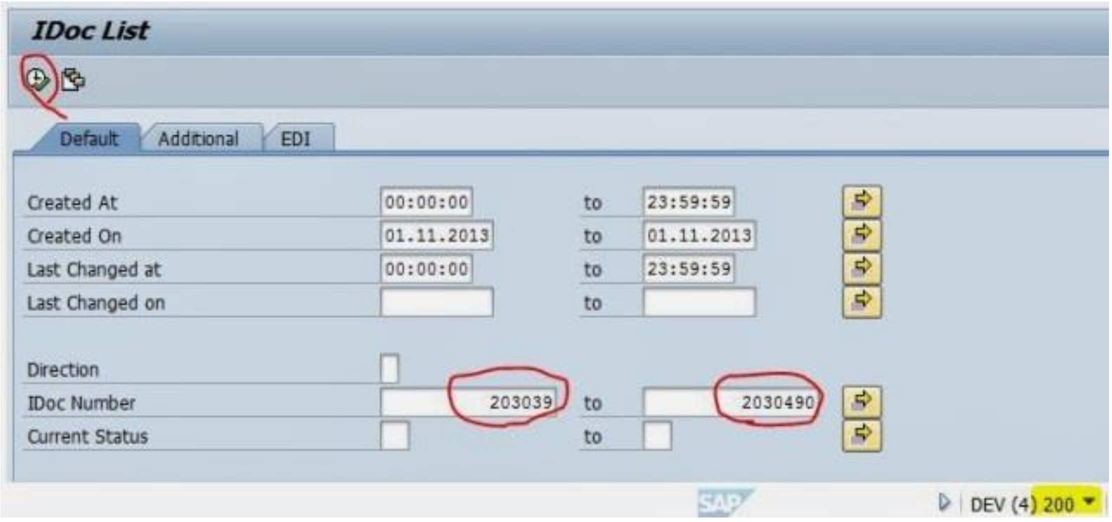


Country tax details transfer by custom idoc

Country tax details transfer by custom idoc

Idoc Generated : 0000000000203040

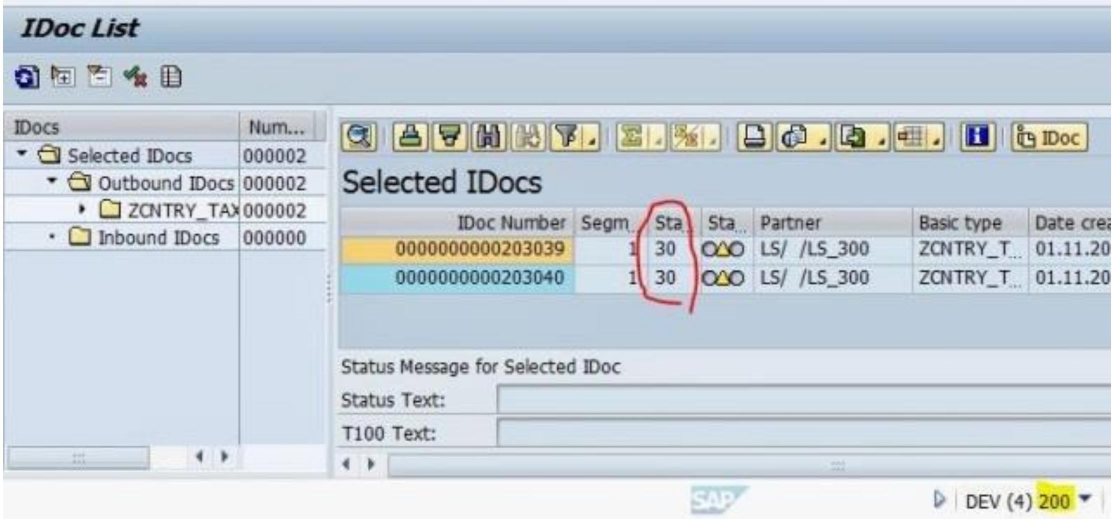
**Step122. Now go to Tcode- WE02 in the client 200 to see the Idoc details.Provide the idoc numbers and execute it.**



IDoc List

| | | | |
|---|---|---|---|
| Created At | 00:00:00 | to | 23:59:59 |
| Created On | 01.11.2013 | to | 01.11.2013 |
| Last Changed at | 00:00:00 | to | 23:59:59 |
| Last Changed on | | to | |
| Direction | | | |
| IDoc Number | 203039 | to | 2030490 |
| Current Status | | to | |

**Step123. Now the idoc status is 30(ready for dispatch to the port). If the status is 03 then idoc is dispatched.**



IDoc List

| IDocs | Num... |
|---|---|
| Selected IDocs | 000002 |
| Outbound IDocs | 000002 |
| ZCNTRY_TAX | 000002 |
| Inbound IDocs | 000000 |

Selected IDocs

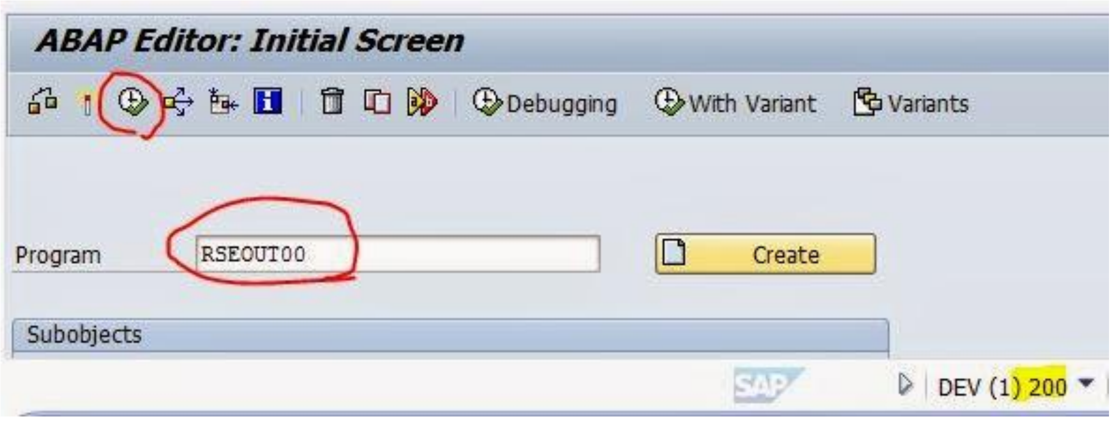
| IDoc Number | Segm... | Sta... | Sta... | Partner | Basic type | Date crea |
|---|---|---|---|---|---|---|
| 0000000000203039 | 1 | 30 | ○△○ | LS/ /LS_300 | ZCNTRY_T... | 01.11.20 |
| 0000000000203040 | 1 | 30 | ○△○ | LS/ /LS_300 | ZCNTRY_T... | 01.11.20 |

Status Message for Selected IDoc

Status Text:

T100 Text:

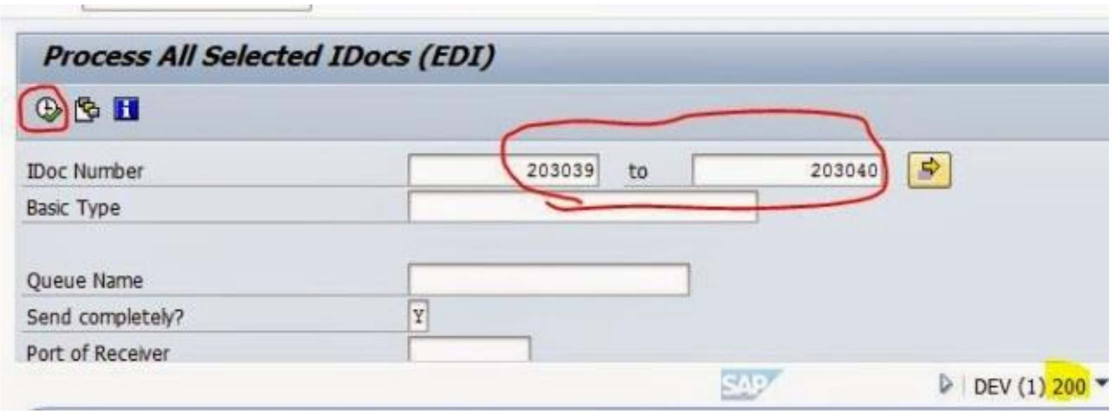
**Step124. Go to SE38 and execute the report RSEOUT00 .**



ABAP Editor: Initial Screen

Program RSEOUT00

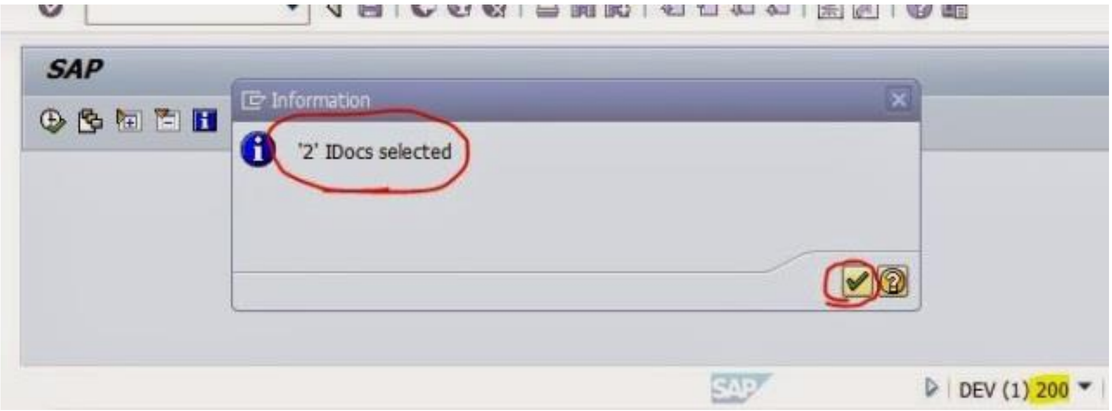
**Step125. Provide the Idoc numbers and execute it.**



Process All Selected IDocs (EDI)

| | | | |
|---|---|---|---|
| IDoc Number | 203039 | to | 203040 |
| Basic Type | | | |
| Queue Name | | | |
| Send completely? | Y | | |
| Port of Receiver | | | |

**Step126. clckc on the continue button.**

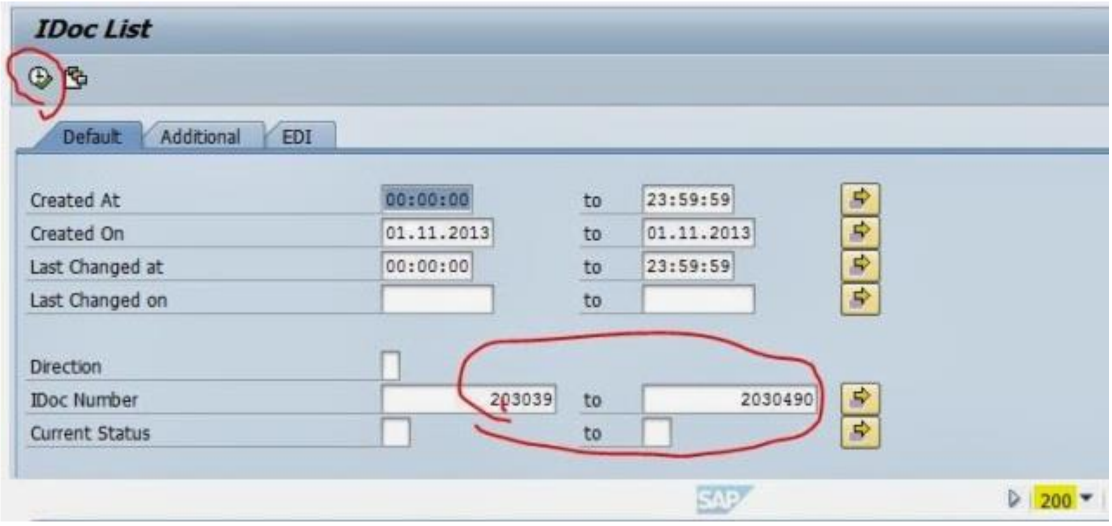


SAP

Information

'2' IDocs selected

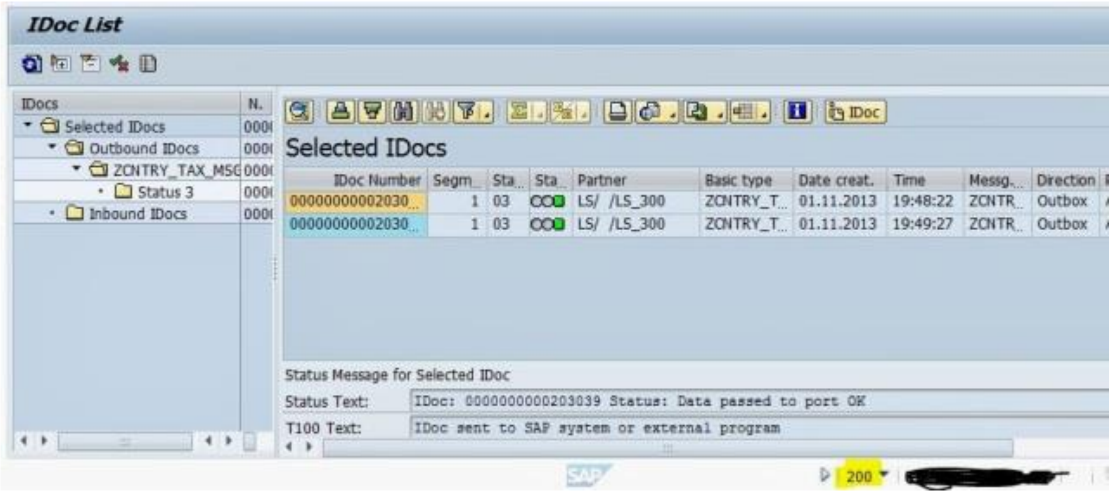
**Step127. Now again go to tcode- WE02 and provide the Idoc numbers and execute it.**



IDoc List

| | | | |
|---|---|---|---|
| Created At | 00:00:00 | to | 23:59:59 |
| Created On | 01.11.2013 | to | 01.11.2013 |
| Last Changed at | 00:00:00 | to | 23:59:59 |
| Last Changed on | | to | |
| Direction | | | |
| IDoc Number | 203039 | to | 2030490 |
| Current Status | | to | |

**Step128. Now the idoc status is 03 with green status. Double click on each idoc to see the details of it.**



IDoc List

| IDocs | N. |
|---|---|
| Selected IDocs | 000( |
| Outbound IDocs | 000( |
| ZCNTRY_TAX_MSG | 000( |
| Status 3 | 000( |
| Inbound IDocs | 000( |

Selected IDocs

| IDoc Number | Segm.. | Sta.. | Sta.. | Partner | Basic type | Date creat. | Time | Messg.. | Direction |
|---|---|---|---|---|---|---|---|---|---|
| 00000000002030.. | 1 | 03 | ○○● | LS/ /LS_300 | ZCNTRY_T.. | 01.11.2013 | 19:48:22 | ZCNTR.. | Outbox |
| 00000000002030.. | 1 | 03 | ○○● | LS/ /LS_300 | ZCNTRY_T.. | 01.11.2013 | 19:49:27 | ZCNTR.. | Outbox |

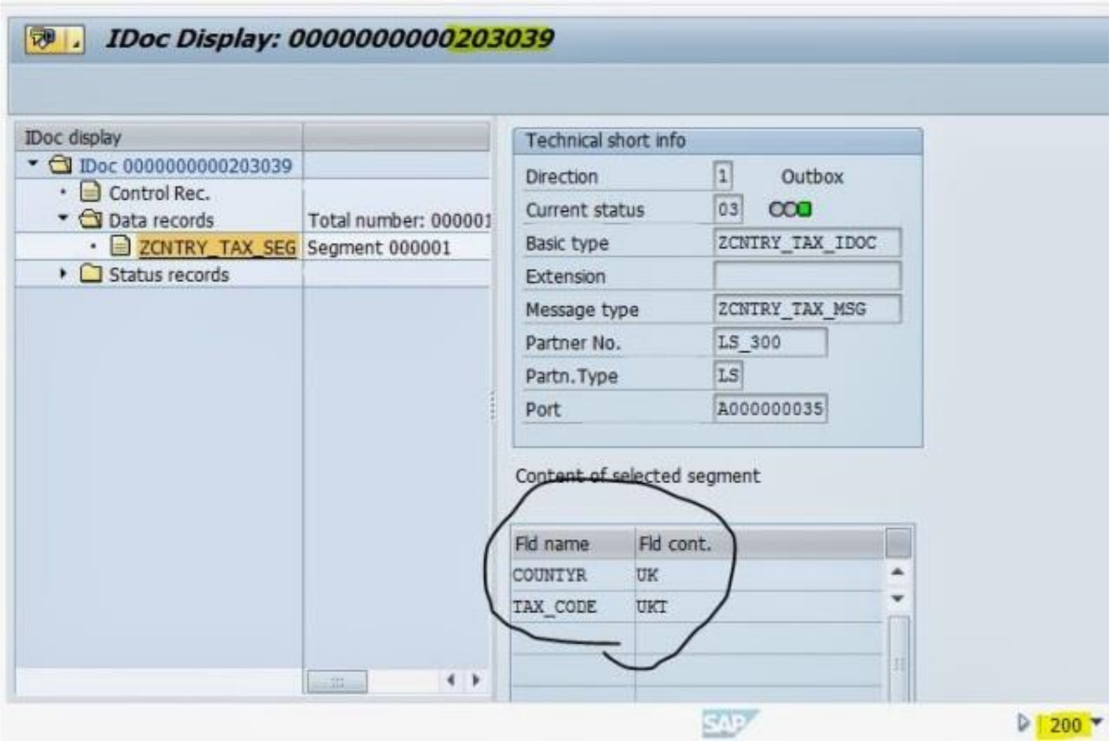
Status Message for Selected IDoc

Status Text: IDoc: 0000000000203039 Status: Data passed to port OK

T100 Text: IDoc sent to SAP system or external program

200 [redacted]

**Step129. The IDOC- 203039 contains the details as shown below.**



IDoc Display: 0000000000203039

IDoc display

- IDoc 0000000000203039
  - Control Rec.
  - Data records — Total number: 000001
    - ZCNTRY_TAX_SEG — Segment 000001
  - Status records

Technical short info

Direction: 1 Outbox

Current status: 03 ○○●

Basic type: ZCNTRY_TAX_IDOC

Extension:

Message type: ZCNTRY_TAX_MSG

Partner No.: LS_300

Partn.Type: LS

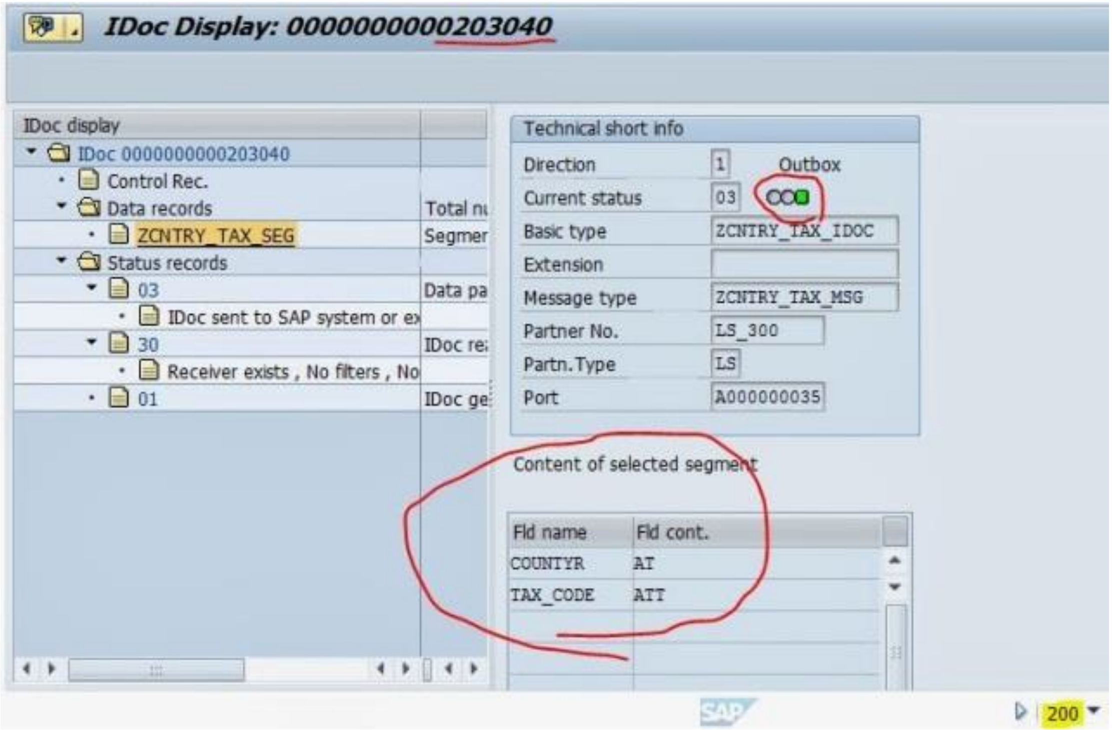
Port: A000000035

Content of selected segment

| Fld name | Fld cont. |
|---|---|
| COUNTYR | UK |
| TAX_CODE | UKT |

200

**Step130. The IDOC- 203040 contains the details as shown below.**



IDoc Display: 0000000000203040

IDoc display
- IDoc 0000000000203040
  - Control Rec.
  - Data records — Total nu
    - ZCNTRY_TAX_SEG — Segmer
  - Status records
    - 03 — Data pa
      - IDoc sent to SAP system or ex
    - 30 — IDoc re:
      - Receiver exists , No filters , No
    - 01 — IDoc ge

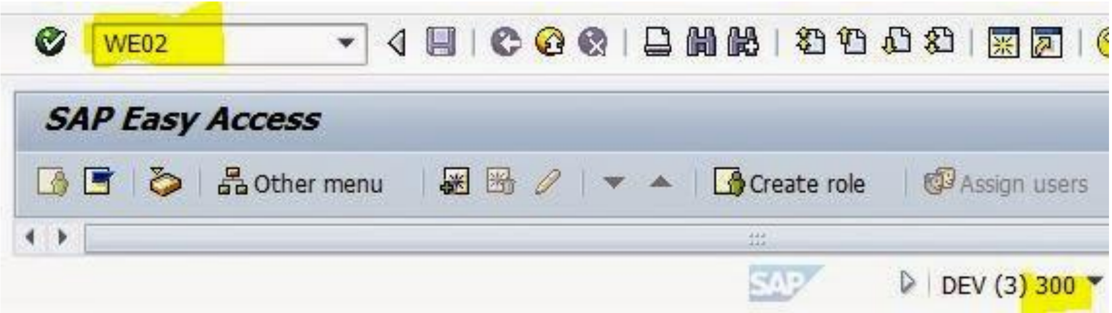
Technical short info

| | | |
|---|---|---|
| Direction | 1 | Outbox |
| Current status | 03 | ●○○ |
| Basic type | ZCNTRY_TAX_IDOC | |
| Extension | | |
| Message type | ZCNTRY_TAX_MSG | |
| Partner No. | LS_300 | |
| Partn.Type | LS | |
| Port | A000000035 | |

Content of selected segment

| Fld name | Fld cont. |
|---|---|
| COUNTYR | AT |
| TAX_CODE | ATT |

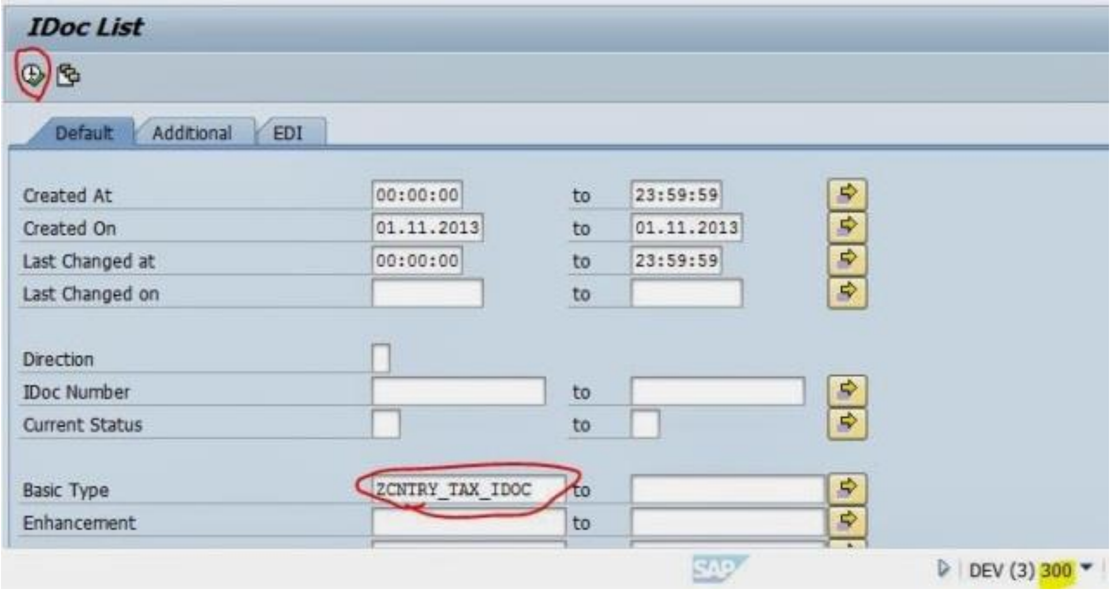
**Step131. Now go to code- WE02 in the client 300.**



WE02

SAP Easy Access

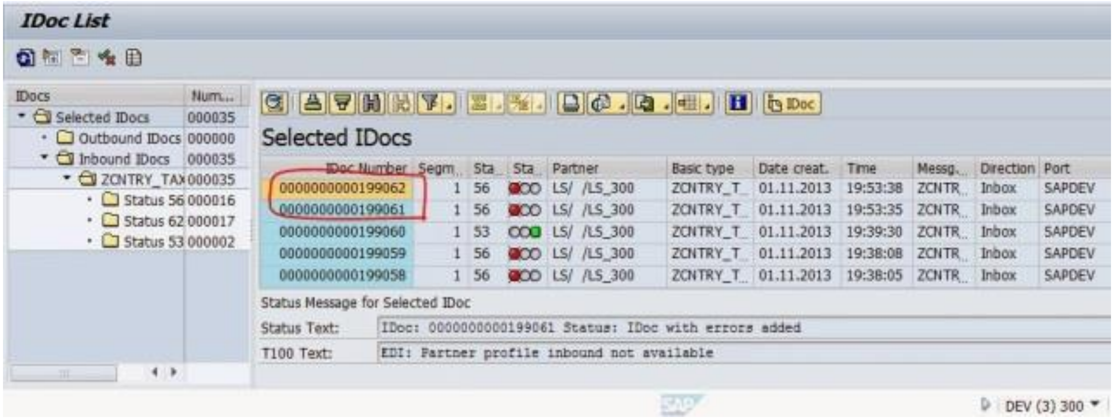
**Step132. Provide the Idoc type name and execute the report.**



IDoc List

| | | | |
|---|---|---|---|
| Created At | 00:00:00 | to | 23:59:59 |
| Created On | 01.11.2013 | to | 01.11.2013 |
| Last Changed at | 00:00:00 | to | 23:59:59 |
| Last Changed on | | to | |
| Direction | | | |
| IDoc Number | | to | |
| Current Status | | to | |
| Basic Type | ZCNTRY_TAX_IDOC | to | |
| Enhancement | | to | |

**Step133. Now the idocs are with status 56 with red status.**



| IDocs | Num... |
|---|---|
| Selected IDocs | 000035 |
| Outbound IDocs | 000000 |
| Inbound IDocs | 000035 |
| ZCNTRY_TAX | 000035 |
| Status 56 | 000016 |
| Status 62 | 000017 |
| Status 53 | 000002 |

Selected IDocs

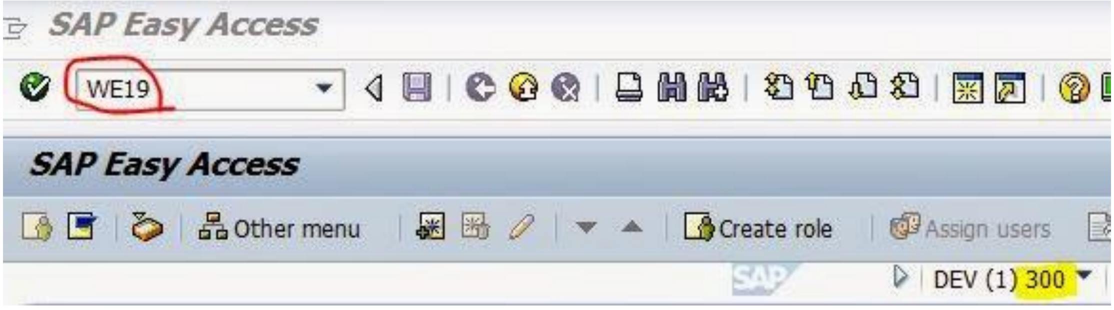
| IDoc Number | Segm... | Sta... | Sta... | Partner | Basic type | Date creat. | Time | Messg... | Direction | Port |
|---|---|---|---|---|---|---|---|---|---|---|
| 0000000000199062 | 1 | 56 | ●○○ | LS/ /LS_300 | ZCNTRY_T... | 01.11.2013 | 19:53:38 | ZCNTR... | Inbox | SAPDEV |
| 0000000000199061 | 1 | 56 | ●○○ | LS/ /LS_300 | ZCNTRY_T... | 01.11.2013 | 19:53:35 | ZCNTR... | Inbox | SAPDEV |
| 0000000000199060 | 1 | 53 | ○○● | LS/ /LS_300 | ZCNTRY_T... | 01.11.2013 | 19:39:30 | ZCNTR... | Inbox | SAPDEV |
| 0000000000199059 | 1 | 56 | ●○○ | LS/ /LS_300 | ZCNTRY_T... | 01.11.2013 | 19:38:08 | ZCNTR... | Inbox | SAPDEV |
| 0000000000199058 | 1 | 56 | ●○○ | LS/ /LS_300 | ZCNTRY_T... | 01.11.2013 | 19:38:05 | ZCNTR... | Inbox | SAPDEV |

Status Message for Selected IDoc

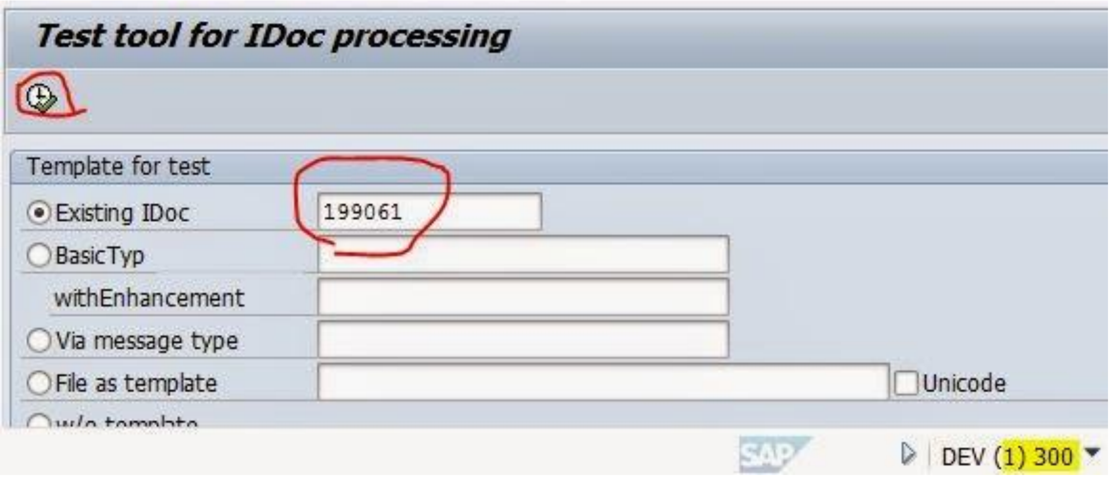
Status Text: IDoc: 0000000000199061 Status: IDoc with errors added

T100 Text: EDI: Partner profile inbound not available

**Step134. Go to Tcode WE19 and provide the inbound Idoc number.**



**Step135. Provide the Inbound Idoc number and execute it.**



Template for test

● Existing IDoc 199061

○ BasicTyp

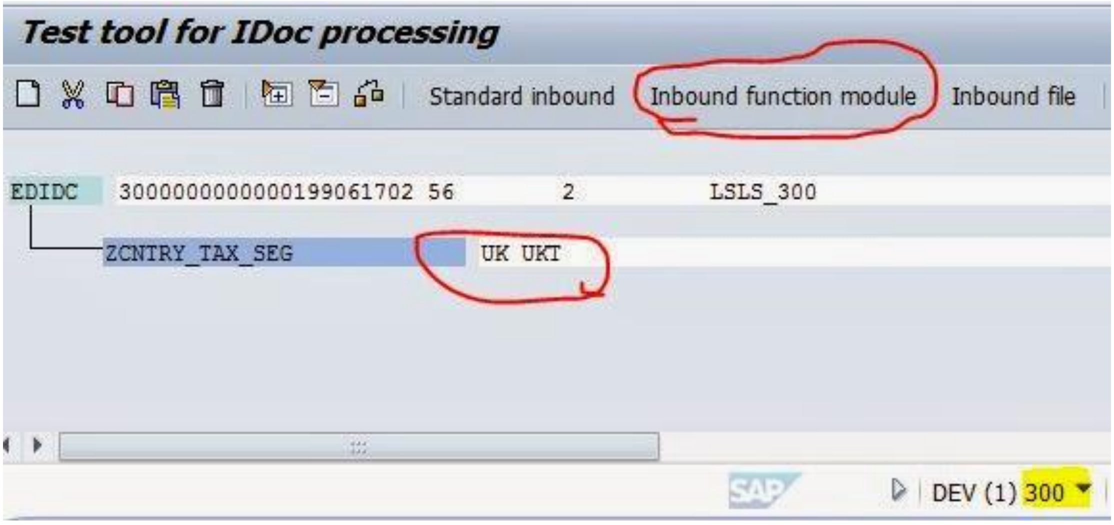
withEnhancement

○ Via message type

○ File as template □ Unicode

○ w/o template

**Step136. Now click on Inbound Function Module name .**

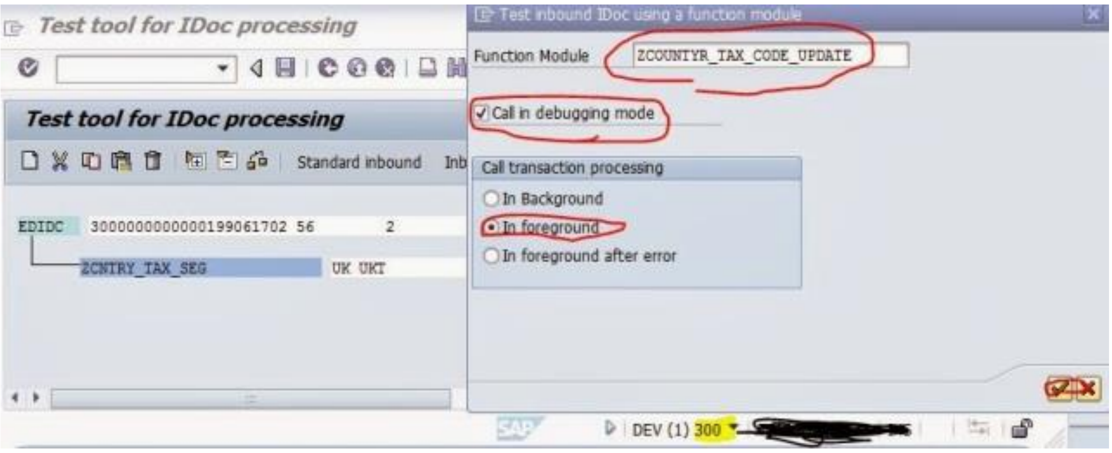


Test tool for IDoc processing

EDIDC 3000000000000199061702 56 2 LSLS_300

ZCNTRY_TAX_SEG UK UKT

**Step137. From the popup, select the Call in debugging mode check box and Select the radio button In foreground and then click on Continue button.**



Test inbound IDoc using a function module

Function Module ZCOUNTYR_TAX_CODE_UPDATE

☑ Call in debugging mode

Call transaction processing

○ In Background

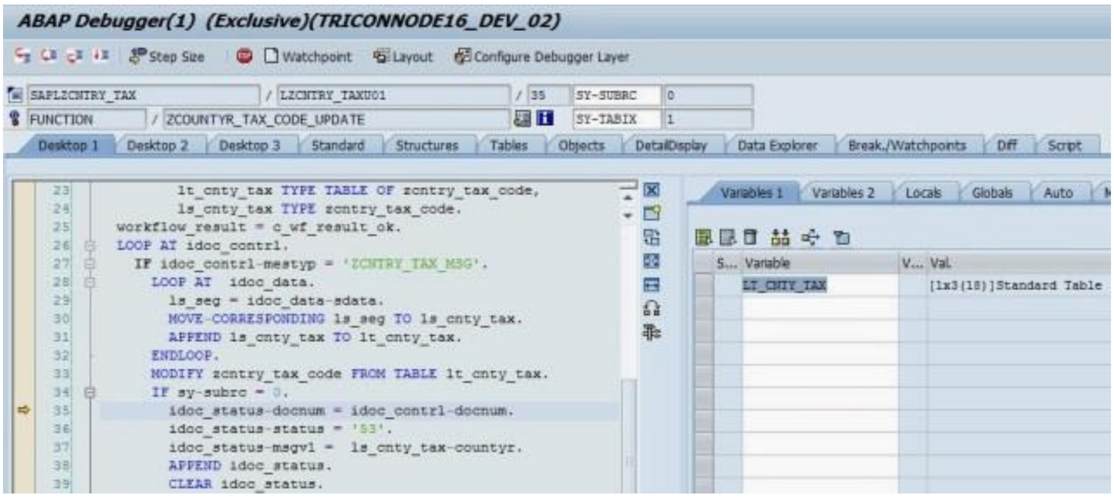
● In foreground

○ In foreground after error

Test tool for IDoc processing
Standard inbound
Inb
EDIDC 3000000000000199061702 56 2
ZCNTRY_TAX_SEG UK UKT
DEV (1) 300

[redacted]

**Step138.**



ABAP Debugger(1) (Exclusive)(TRICONNODE16_DEV_02)

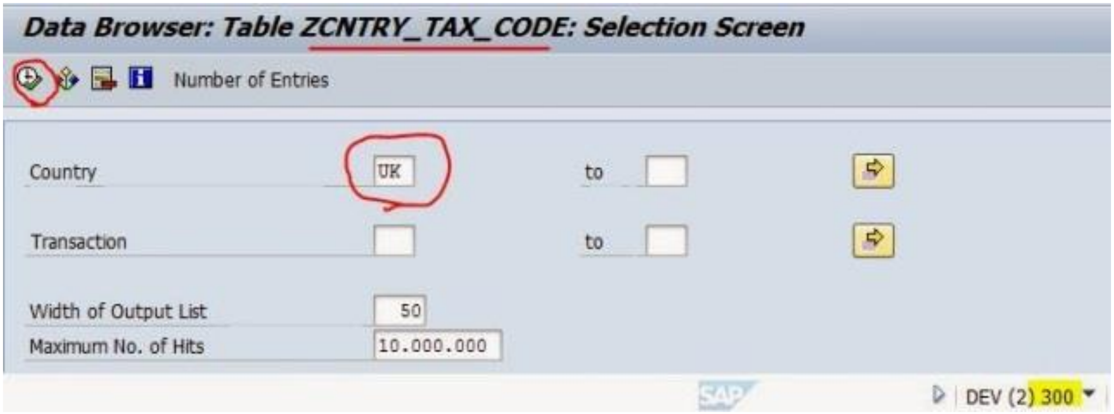
SAPLZCNTRY_TAX / LZCNTRY_TAXU01 / 35 SY-SUBRC 0

FUNCTION / ZCOUNTYR_TAX_CODE_UPDATE SY-TABIX 1

```
         lt_cnty_tax TYPE TABLE OF zcntry_tax_code,
         ls_cnty_tax TYPE zcntry_tax_code.
  workflow_result = c_wf_result_ok.
  LOOP AT idoc_contrl.
    IF idoc_contrl-mestyp = 'ZCNTRY_TAX_MSG'.
      LOOP AT  idoc_data.
        ls_seg = idoc_data-sdata.
        MOVE-CORRESPONDING ls_seg TO ls_cnty_tax.
        APPEND ls_cnty_tax TO lt_cnty_tax.
      ENDLOOP.
      MODIFY zcntry_tax_code FROM TABLE lt_cnty_tax.
      IF sy-subrc = 0.
        idoc_status-docnum = idoc_contrl-docnum.
        idoc_status-status = '53'.
        idoc_status-msgv1 =  ls_cnty_tax-countyr.
        APPEND idoc_status.
        CLEAR idoc_status.
```

| S... | Variable | V... | Val. |
|---|---|---|---|
| | LT_CNTY_TAX | | [1x3(18)]Standard Table |

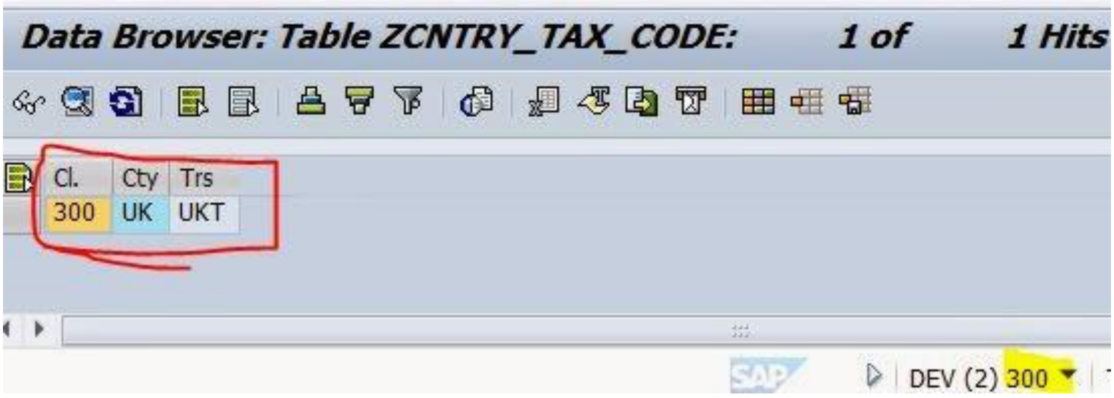
**Step139. Now go to SE11 and open the table and go to the contents button. Provide the country name UK and execute it.**



Data Browser: Table ZCNTRY_TAX_CODE: Selection Screen

| | | | |
|---|---|---|---|
| Country | UK | to | |
| Transaction | | to | |
| Width of Output List | 50 | | |
| Maximum No. of Hits | 10.000.000 | | |

**Step140. The Record is available in the client 300.**



Data Browser: Table ZCNTRY_TAX_CODE: 1 of 1 Hits

| Cl. | Cty | Trs |
|---|---|---|
| 300 | UK | UKT |