## SAP Logistics Business Network, Global Track and Trace Option

# Track Purchase Orders - Deep Dive with SAP ERP Integration

Logistics Business Network
December 2020

PUBLIC

THE BEST RUN

# Objectives

## After completing this learning module, you will be able to:

- □ Learn what prerequisite is necessary for Global Track and Trace Option
- □ Learn how to maintain IDOC configurations in ERP for integration
- □ Learn how to maintain extractors in ERP for integration
- □ Learn how to download and implement sample ABAP codes from Github
- □ Learn how to customize own logic based on sample codes

# Agenda

**A Prerequisites**

**B Configuration and Implementation - Basic**

**B1 IDOC Configuration**

**B2 Extractor Configuration**

**C Download ABAP Code from GitHub**

**D Configuration and Coding Guide - Advanced**

# A) Prerequisites

# STEP 1: Check the SAP Version

1-1: The SAP Product Version for GTT v2 shall be SAP EHP1 FOR SAP NETWEAVER 7.3 or higher.

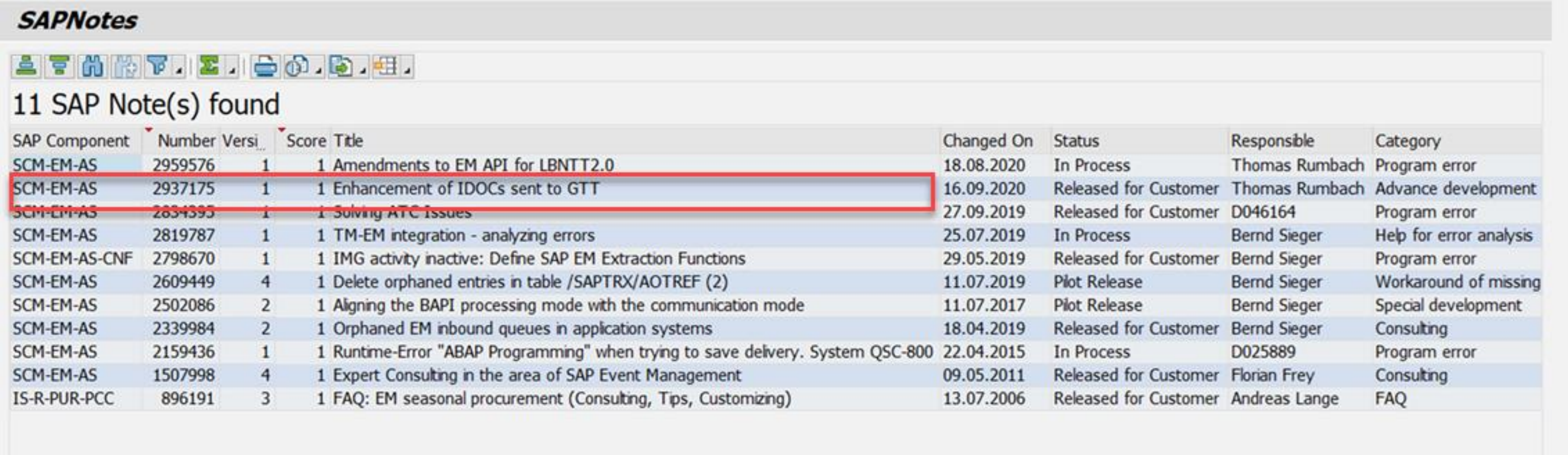
1-2: SAP NOTE 2937175 shall be implemented.

1-3: The ABAP codes to support sample applications for GTT v2 can be implemented in S4 HANA 2101 on premise, which is not validated in lower release.

TIPs:

1, SAP version reference: https://support.sap.com/en/my-support/software-downloads/support-package-stacks/product-versions.html#section

2, Note-assistant reference: https://support.sap.com/en/my-support/knowledge-base/note-assistant.html



SAPNotes

11 SAP Note(s) found

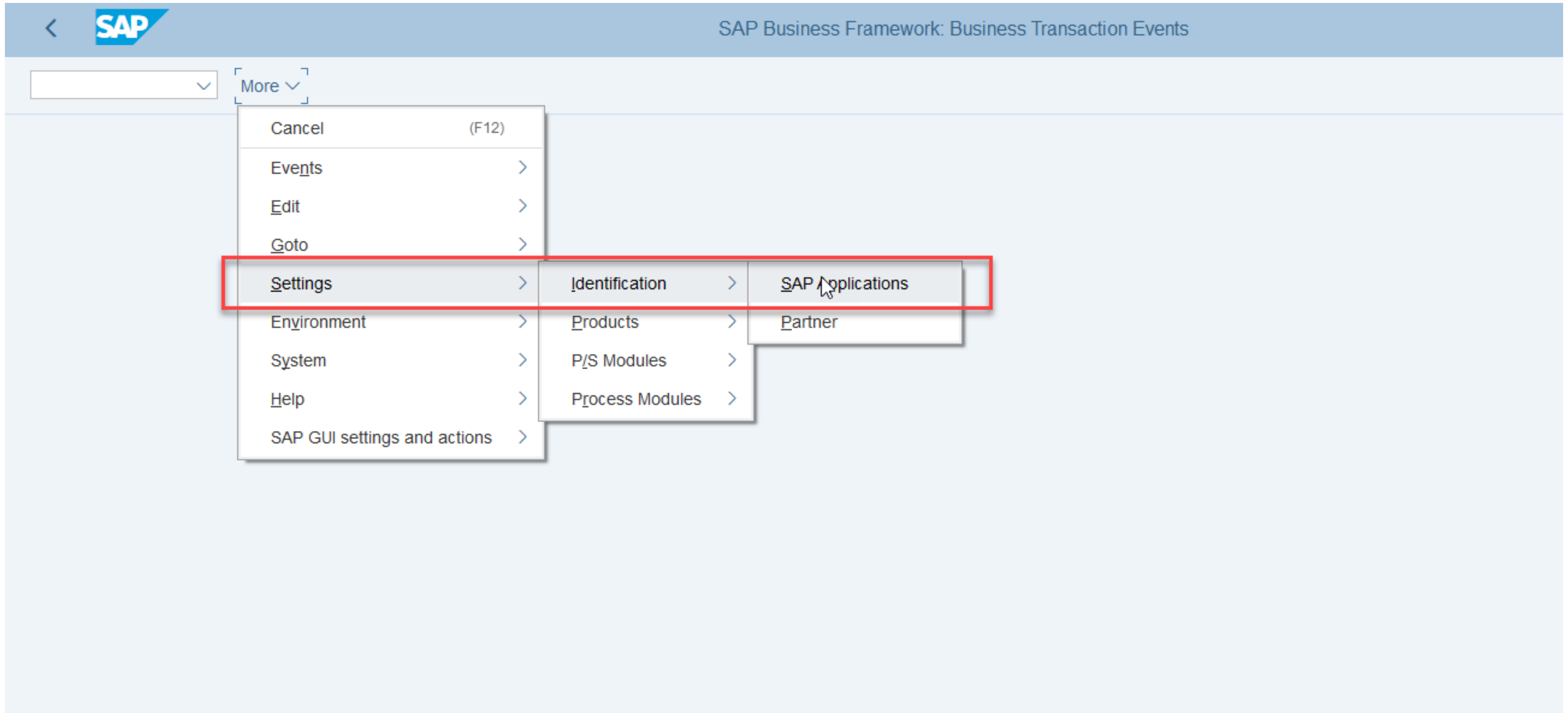
| SAP Component | Number | Versi.. | Score | Title | Changed On | Status | Responsible | Category |
|---|---|---|---|---|---|---|---|---|
| SCM-EM-AS | 2959576 | 1 | 1 | Amendments to EM API for LBNTT2.0 | 18.08.2020 | In Process | Thomas Rumbach | Program error |
| SCM-EM-AS | 2937175 | 1 | 1 | Enhancement of IDOCs sent to GTT | 16.09.2020 | Released for Customer | Thomas Rumbach | Advance development |
| SCM-EM-AS | 2834393 | 1 | 1 | Solving ATC Issues | 27.09.2019 | Released for Customer | D046164 | Program error |
| SCM-EM-AS | 2819787 | 1 | 1 | TM-EM integration - analyzing errors | 25.07.2019 | In Process | Bernd Sieger | Help for error analysis |
| SCM-EM-AS-CNF | 2798670 | 1 | 1 | IMG activity inactive: Define SAP EM Extraction Functions | 29.05.2019 | Released for Customer | Bernd Sieger | Program error |
| SCM-EM-AS | 2609449 | 4 | 1 | Delete orphaned entries in table /SAPTRX/AOTREF (2) | 11.07.2019 | Pilot Release | Bernd Sieger | Workaround of missing |
| SCM-EM-AS | 2502086 | 2 | 1 | Aligning the BAPI processing mode with the communication mode | 11.07.2017 | Pilot Release | Bernd Sieger | Special development |
| SCM-EM-AS | 2339984 | 2 | 1 | Orphaned EM inbound queues in application systems | 18.04.2019 | Released for Customer | Bernd Sieger | Consulting |
| SCM-EM-AS | 2159436 | 1 | 1 | Runtime-Error "ABAP Programming" when trying to save delivery. System QSC-800 | 22.04.2015 | In Process | D025889 | Program error |
| SCM-EM-AS | 1507998 | 4 | 1 | Expert Consulting in the area of SAP Event Management | 09.05.2011 | Released for Customer | Florian Frey | Consulting |
| IS-R-PUR-PCC | 896191 | 3 | 1 | FAQ: EM seasonal procurement (Consulting, Tips, Customizing) | 13.07.2006 | Released for Customer | Andreas Lange | FAQ |

# STEP 2: Log on the Development Client to Configure BTE

2-1: Ensure you have development access to the client for cross-client customizing and local development

2-2: Log on to the client and enter transaction code (T-code): **FIBF**

2-3: Click **More -> Settings -> Identification -> SAP Applications**

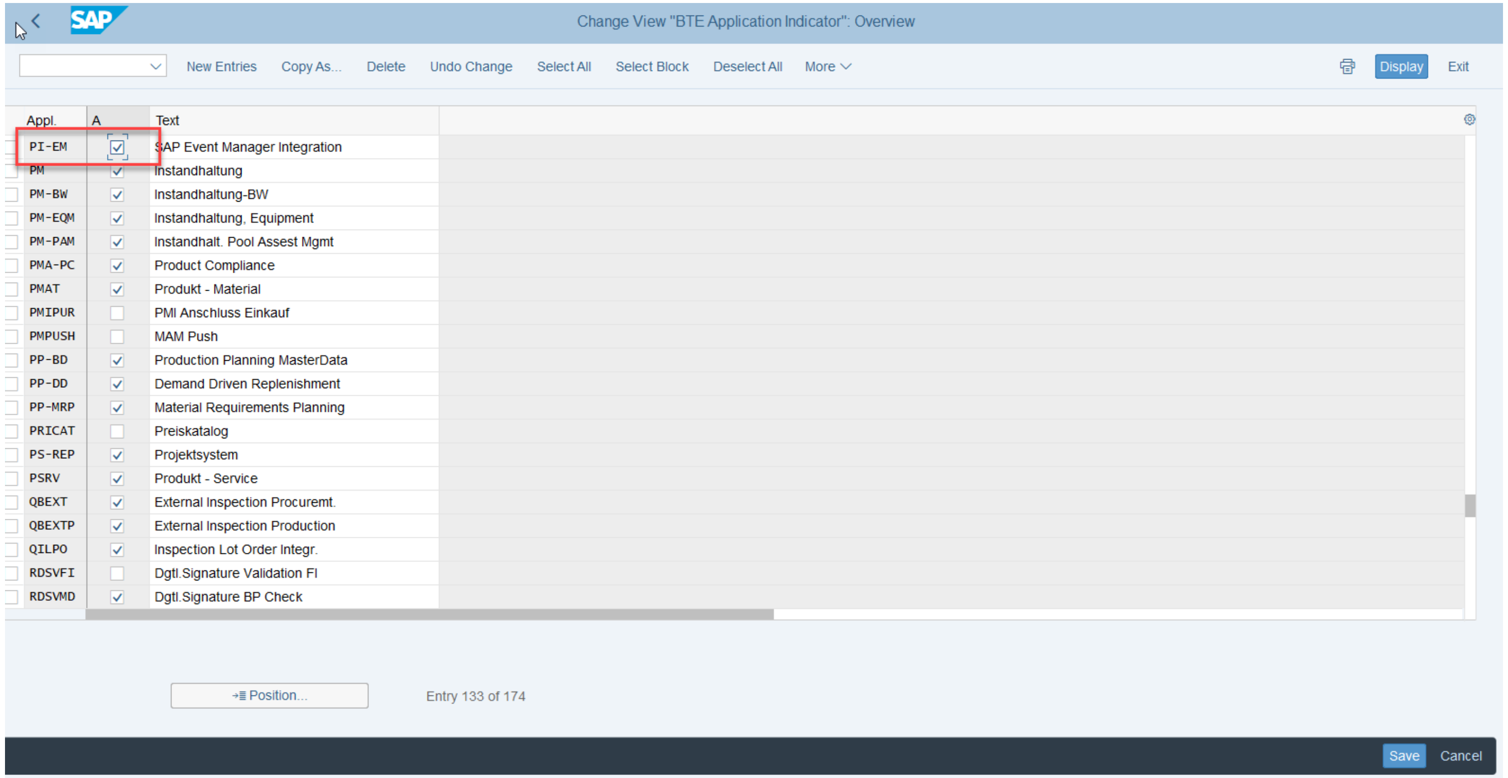


# STEP 2: Activate SAP Event Manager Integration

2-4: Position on the Application ID: **PI-EM**

2-5: Check the field **Application Active**

2-6: Click **Save**



Change View "BTE Application Indicator": Overview

| Appl. | A | Text |
|---|---|---|
| PI-EM | ☑ | SAP Event Manager Integration |
| PM | ☑ | Instandhaltung |
| PM-BW | ☑ | Instandhaltung-BW |
| PM-EQM | ☑ | Instandhaltung, Equipment |
| PM-PAM | ☑ | Instandhalt. Pool Assest Mgmt |
| PMA-PC | ☑ | Product Compliance |
| PMAT | ☑ | Produkt - Material |
| PMIPUR | □ | PMI Anschluss Einkauf |
| PMPUSH | □ | MAM Push |
| PP-BD | ☑ | Production Planning MasterData |
| PP-DD | ☑ | Demand Driven Replenishment |
| PP-MRP | ☑ | Material Requirements Planning |
| PRICAT | □ | Preiskatalog |
| PS-REP | ☑ | Projektsystem |
| PSRV | ☑ | Produkt - Service |
| QBEXT | ☑ | External Inspection Procuremt. |
| QBEXTP | ☑ | External Inspection Production |
| QILPO | ☑ | Inspection Lot Order Integr. |
| RDSVFI | □ | Dgtl.Signature Validation FI |
| RDSVMD | ☑ | Dgtl.Signature BP Check |

Entry 133 of 174

# B) Configuration and Implementation - Basic

# B1. IDOC Configuration

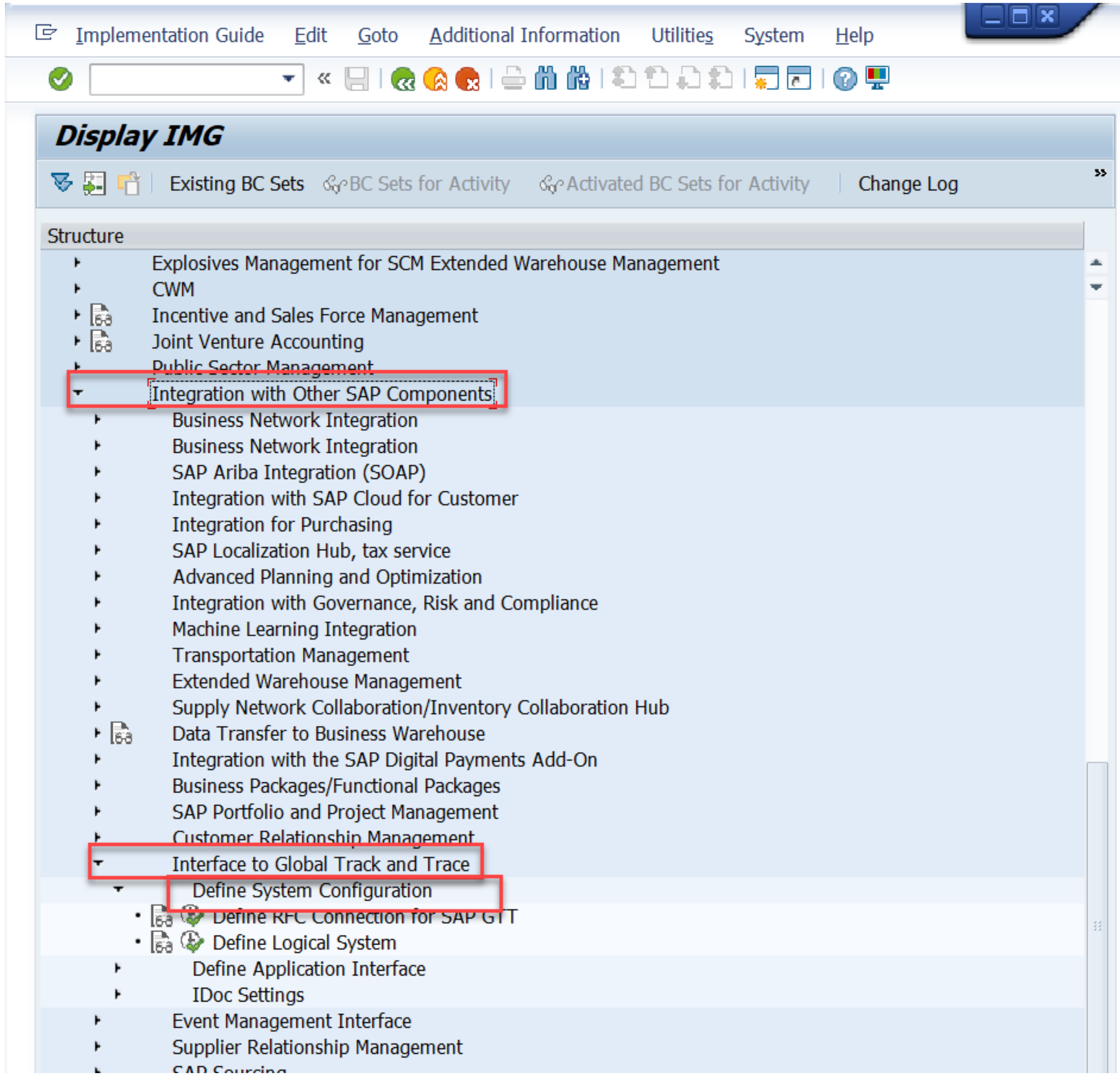
# STEP 1: Define RFC Connection for GTT

1-1: Log on to the business client

1-2: Enter T-code *SPRO* and then click **SAP Reference IMG** to open **Display IMG page**

1-3: Click **Integration with Other SAP Components** -> **Interface to Global Track and Trace** -> **Define System Configuration**

1-4: Choose activity:
**Define RFC Connection for SAP GTT**

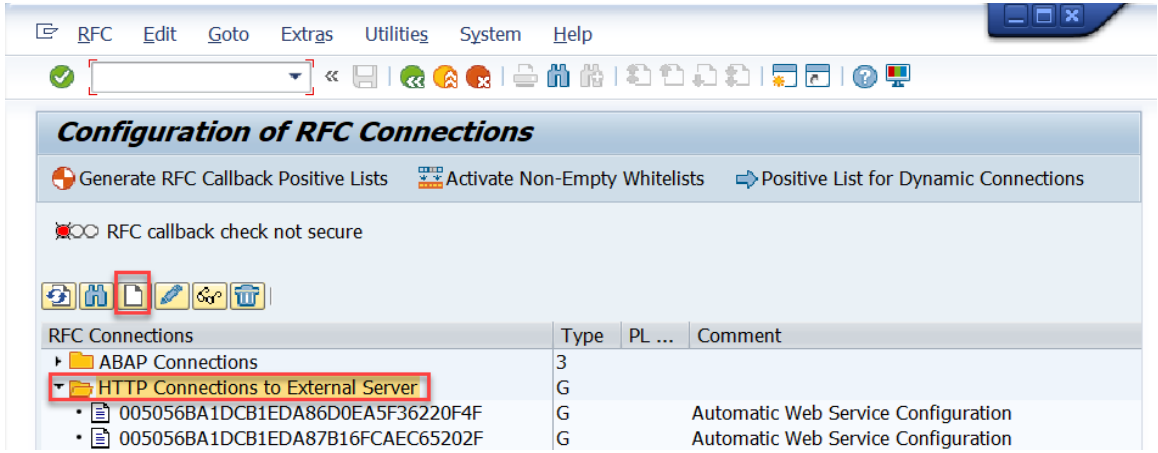


Structure
- Explosives Management for SCM Extended Warehouse Management
- CWM
- Incentive and Sales Force Management
- Joint Venture Accounting
- Public Sector Management
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
      - Define RFC Connection for SAP GTT
      - Define Logical System
    - Define Application Interface
    - IDoc Settings
  - Event Management Interface
  - Supplier Relationship Management
  - SAP Sourcing

# STEP 1: Define RFC Connection for GTT

1-5: Choose **HTTP Connections to External Server**, click **Create** and create a new RFC connection.

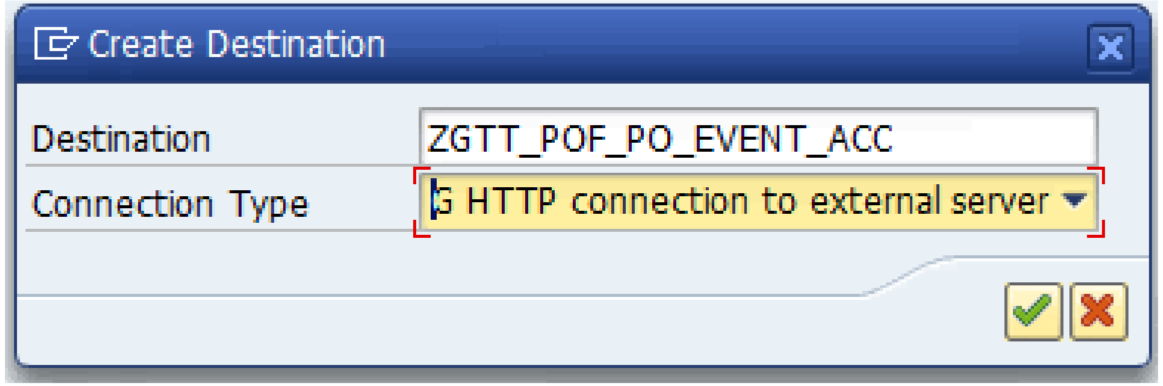
1-6: Fill in the **Destination** and choose the **Connection Type:** **'G-HTTP connection to external server'**.


RFC
Edit
Goto
Extras
Utilities
System
Help
Generate RFC Callback Positive Lists
Activate Non-Empty Whitelists
Positive List for Dynamic Connections

Configuration of RFC Connections

RFC callback check not secure

| RFC Connections | Type | PL ... | Comment |
|---|---|---|---|
| ABAP Connections | 3 | | |
| HTTP Connections to External Server | G | | |
| 005056BA1DCB1EDA86D0EA5F36220F4F | G | | Automatic Web Service Configuration |
| 005056BA1DCB1EDA87B16FCAEC65202F | G | | Automatic Web Service Configuration |



Create Destination

Destination: ZGTT_POF_PO_EVENT_ACC

Connection Type: G HTTP connection to external server

# STEP 1: Define RFC Connection for GTT

1-7: Enter a description

1-8: In the **Technical Settings** tab, fill in the **Host**, **Port** and **Path Prefix**

For example, the url of solution owners is as below:

https://sat-so-01.gtt-flp-lbnplatform-pre-live.cfapps.eu10.hana.ondemand.com/

**Host**: *sat-so-01.gtt-flp-lbnplatform-pre-live.cfapps.eu10.hana.ondemand.com*

**Port**: *443*

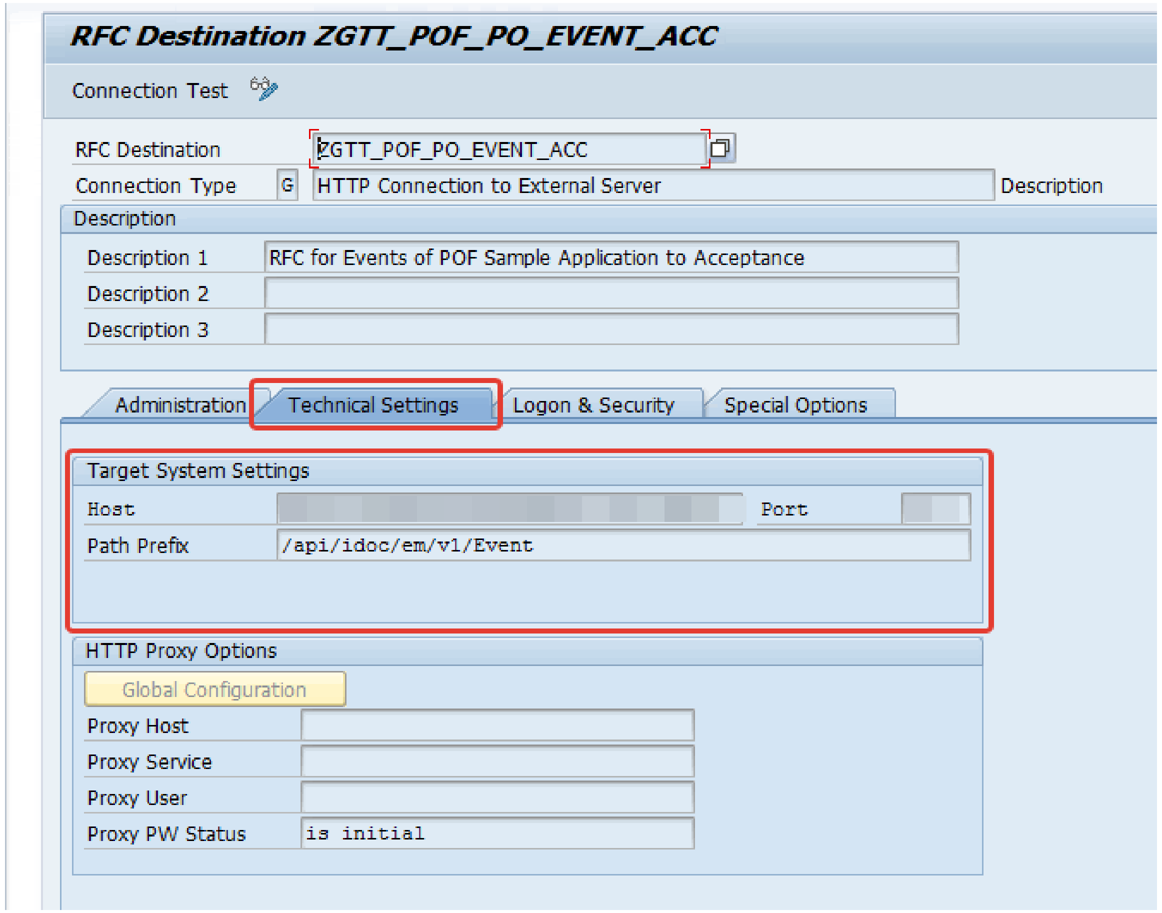
You need to configure two RFC connections separately for event and tracked process. They have different **Path Prefix**es.

For the event:

**Path Prefix**: */api/idoc/em/v1/Event*

For the tracked Process:

**Path Prefix**: */api/idoc/em/v1/TrackedProcess*



**RFC Destination ZGTT_POF_PO_EVENT_ACC**

| | | |
|---|---|---|
| RFC Destination | ZGTT_POF_PO_EVENT_ACC | |
| Connection Type | G HTTP Connection to External Server | Description |

Description

| | |
|---|---|
| Description 1 | RFC for Events of POF Sample Application to Acceptance |
| Description 2 | |
| Description 3 | |

Target System Settings

| | | | |
|---|---|---|---|
| Host | [redacted] | Port | [redacted] |
| Path Prefix | /api/idoc/em/v1/Event | | |

HTTP Proxy Options

| | |
|---|---|
| Proxy Host | |
| Proxy Service | |
| Proxy User | |
| Proxy PW Status | is initial |

# STEP 1: Define RFC Connection for GTT

1-9: In the **Logon & Security** tab, enter the Logon information.

For basic authentication, the GTT technical user / password is needed. You can get this from your GTT administrator.

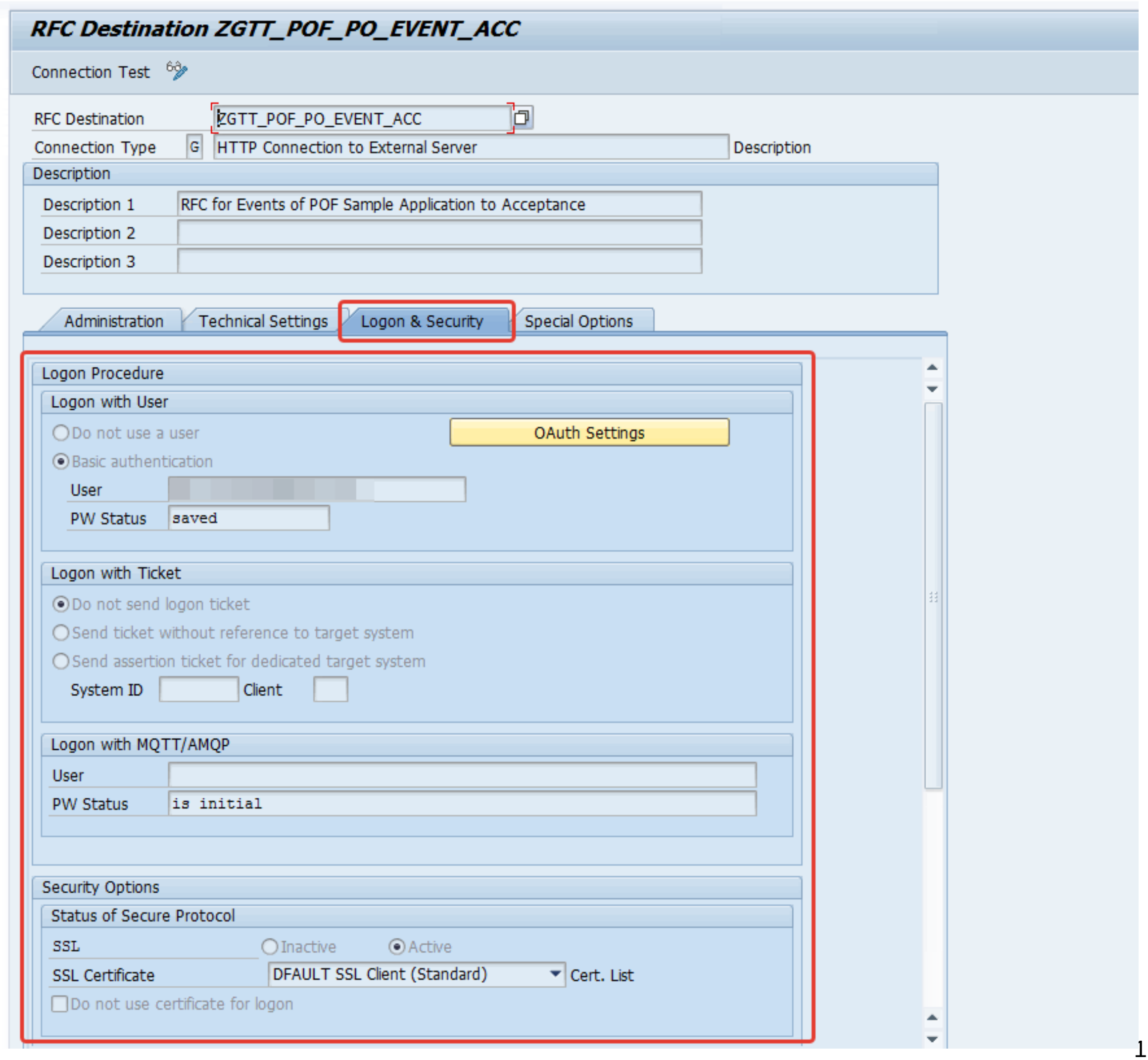
Also, SSL must be *Active.*
The recommended SSL Certificate is: *DFAULT SSL Client (Standard)*.

1-10: Save the configuration

1-11: Click **Connection Test.** A successful connection returns a status HTTP response of 200.

**Caution**: You need to configure two RFC Connections:

- one for event and
- the other for tracked process.



**RFC Destination ZGTT_POF_PO_EVENT_ACC**

Connection Test

RFC Destination: ZGTT_POF_PO_EVENT_ACC
Connection Type: G HTTP Connection to External Server Description

Description
Description 1: RFC for Events of POF Sample Application to Acceptance
Description 2:
Description 3:

Logon Procedure

Logon with User
○ Do not use a user
● Basic authentication
User: [redacted]
PW Status: saved

Logon with Ticket
● Do not send logon ticket
○ Send ticket without reference to target system
○ Send assertion ticket for dedicated target system
System ID: Client:

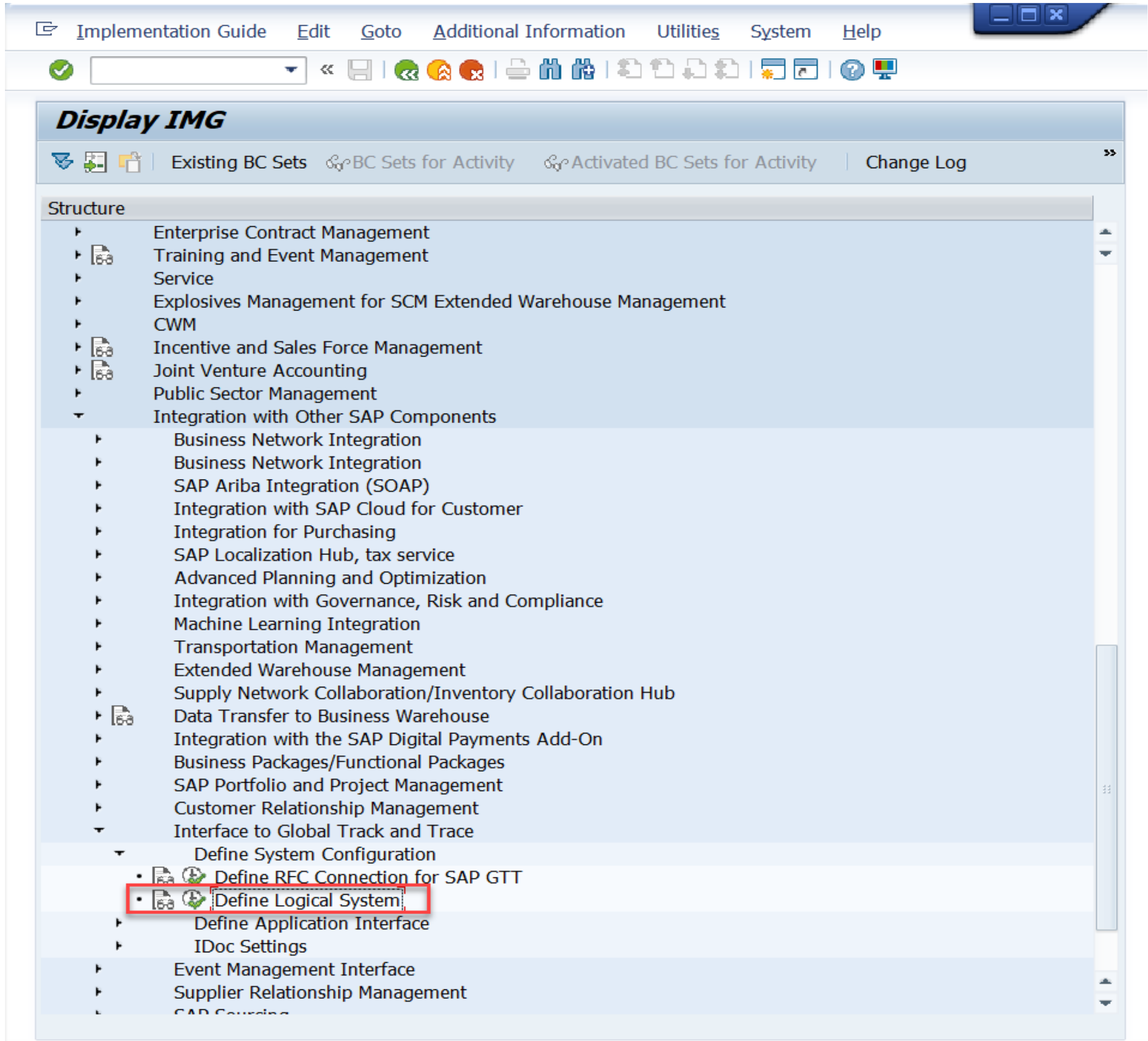
Logon with MQTT/AMQP
User:
PW Status: is initial

Security Options
Status of Secure Protocol
SSL ○ Inactive ● Active
SSL Certificate: DFAULT SSL Client (Standard) Cert. List
□ Do not use certificate for logon

# STEP 2: Define Logical System

2-1: In **Display IMG** page, click **Integration with Other SAP Components -> Interface to Global Track and Trace -> Define System Configuration.**

2-2: Choose activity **Define Logical System**.



Structure

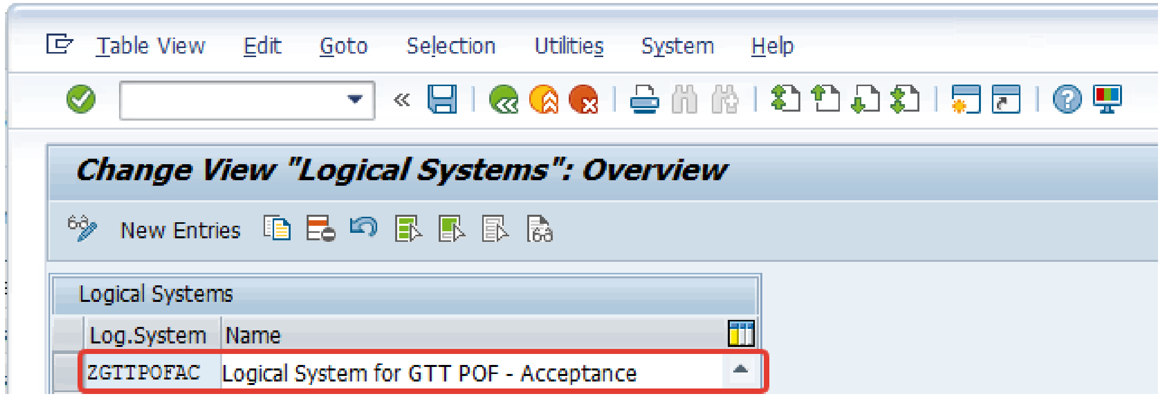
- Enterprise Contract Management
- Training and Event Management
- Service
- Explosives Management for SCM Extended Warehouse Management
- CWM
- Incentive and Sales Force Management
- Joint Venture Accounting
- Public Sector Management
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
      - Define RFC Connection for SAP GTT
      - Define Logical System
    - Define Application Interface
    - IDoc Settings
  - Event Management Interface
  - Supplier Relationship Management
  - SAP Sourcing

# STEP 2: Define Logical System

2-3: Create **New Entries** to create a new Logical System, fill in the:

- Logical system code and
- Name of the new logical system

2-4: Save the configuration



Change View "Logical Systems": Overview

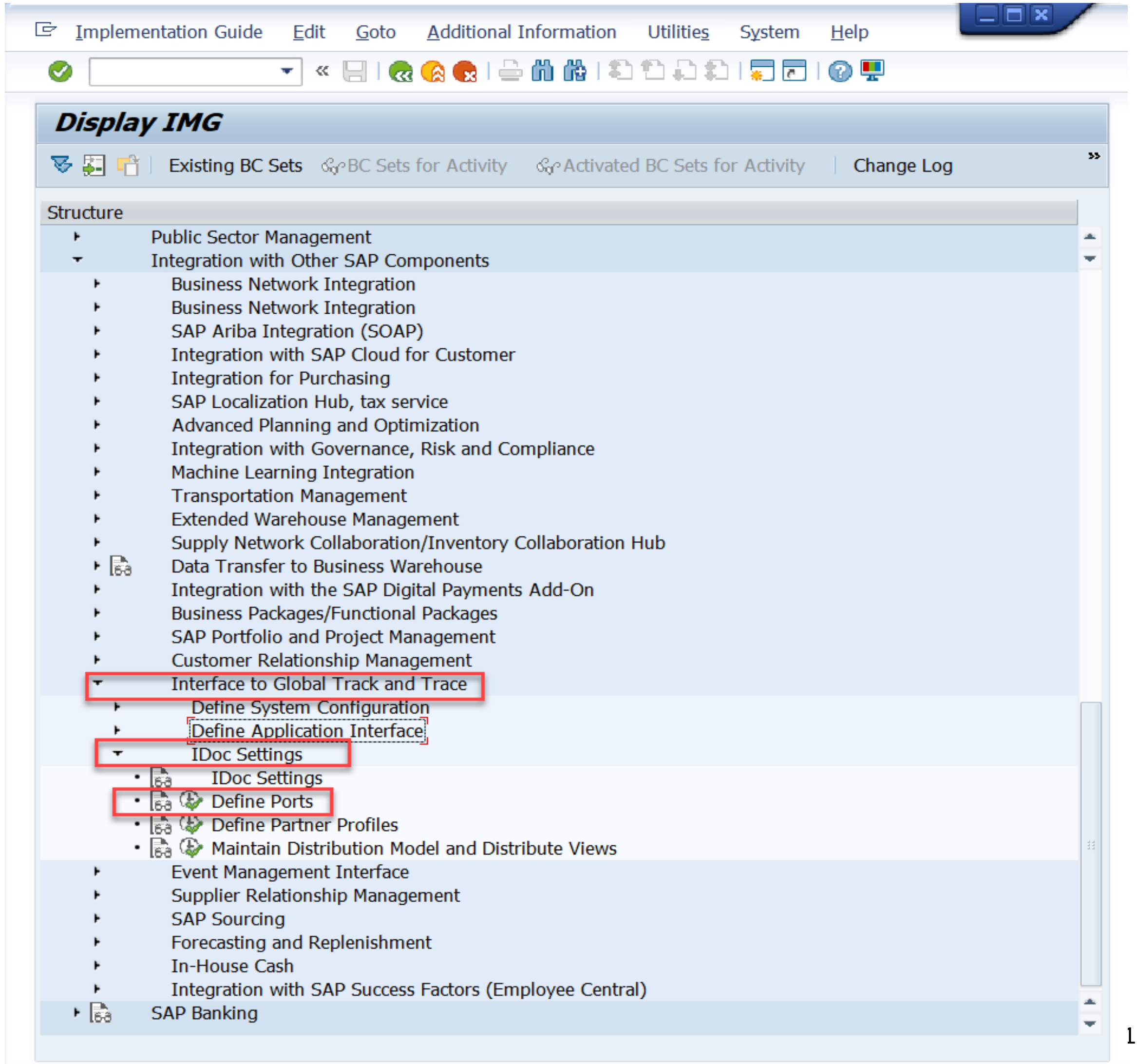
Logical Systems

| Log.System | Name |
|---|---|
| ZGTTPOFAC | Logical System for GTT POF - Acceptance |

# STEP 3: Define Ports

3-1: In **Display IMG** page, click **Integration with Other SAP Components -> Interface to Global Track and Trace -> IDoc Settings**

3-2: Choose activity **Define Ports**



Display IMG

Structure

- Public Sector Management
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
    - Define Application Interface
    - IDoc Settings
      - IDoc Settings
      - Define Ports
      - Define Partner Profiles
      - Maintain Distribution Model and Distribute Views
  - Event Management Interface
  - Supplier Relationship Management
  - SAP Sourcing
  - Forecasting and Replenishment
  - In-House Cash
  - Integration with SAP Success Factors (Employee Central)
- SAP Banking

# STEP 3: Define Ports

3-3: Choose **XML HTTP** folder, and click **Create** to create a new port

3-4: Fill in the **RFC Destination**, it is the RFC connection you created in STEP 1

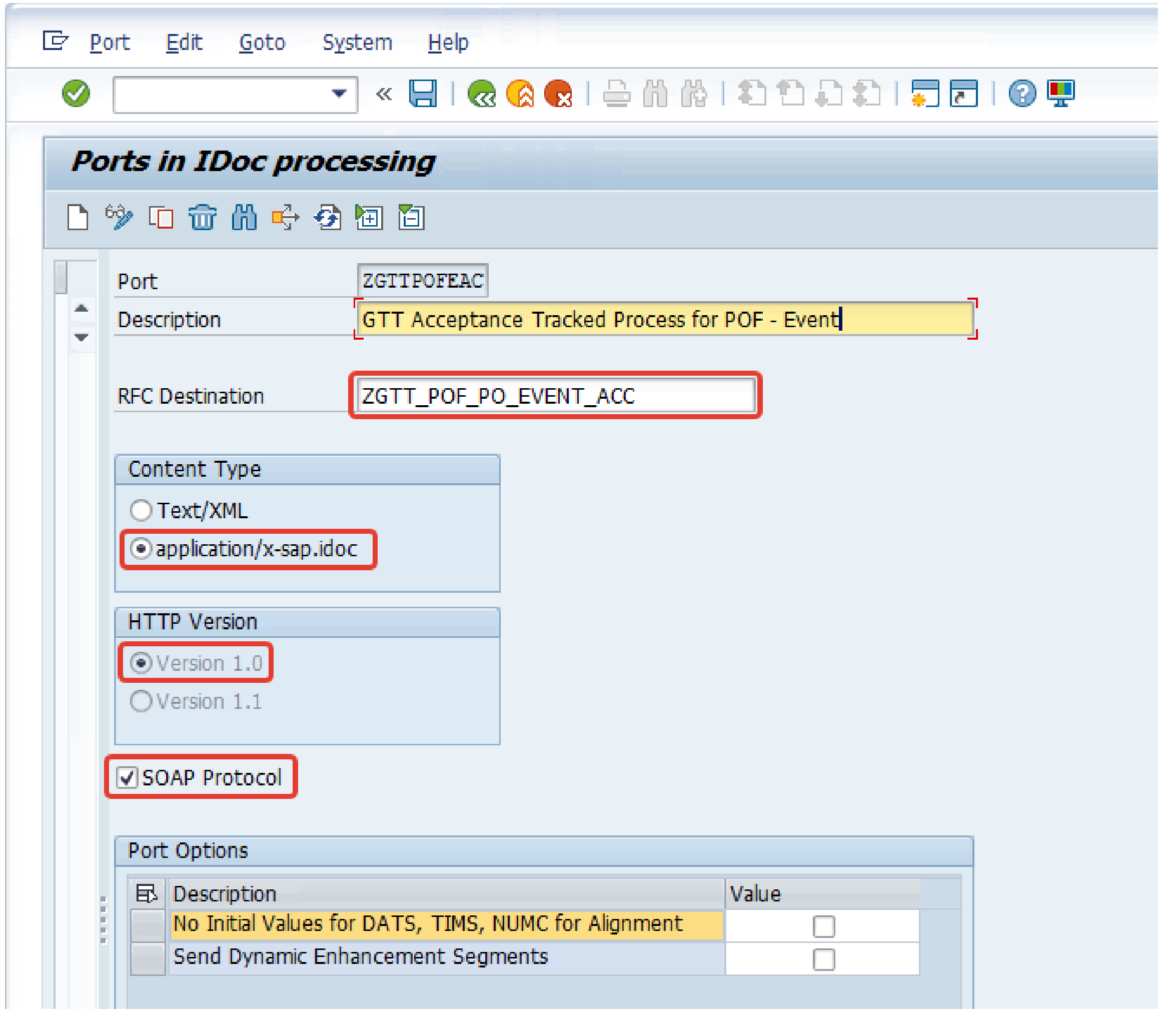
3-5: Choose **Content Type** as *application/x-sap.idoc*

3-6: Choose **HTTP Version** as *Version 1.0*

3-7: Mark it as SOAP Protocol

3-8: Save the configuration

**Caution**: You need to define two ports, one for event and the other for tracked process.



Ports in IDoc processing

| | |
|---|---|
| Port | ZGTTPOFEAC |
| Description | GTT Acceptance Tracked Process for POF - Event |
| RFC Destination | ZGTT_POF_PO_EVENT_ACC |

Content Type

○ Text/XML

● application/x-sap.idoc

HTTP Version

● Version 1.0

○ Version 1.1

☑ SOAP Protocol

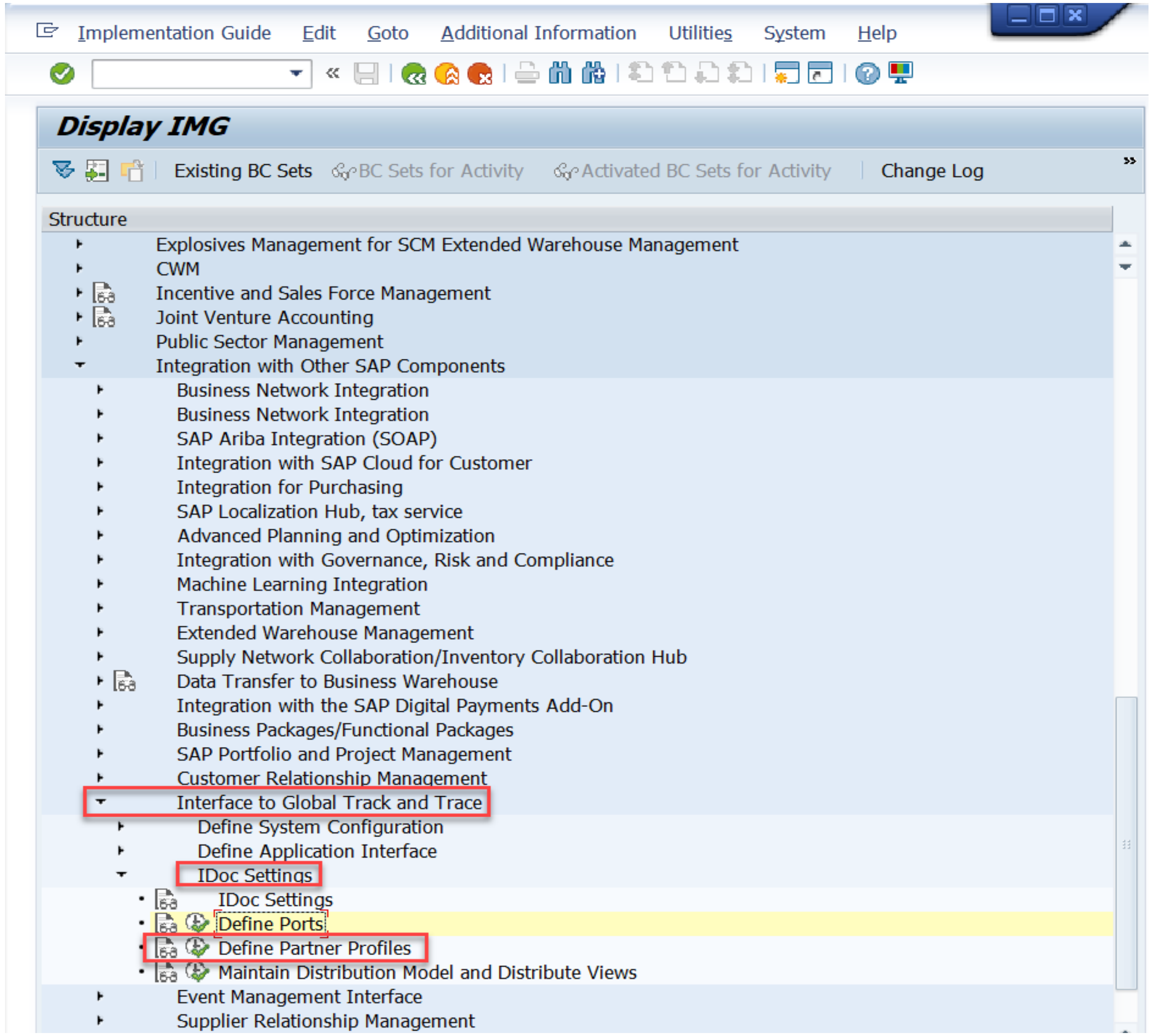
Port Options

| Description | Value |
|---|---|
| No Initial Values for DATS, TIMS, NUMC for Alignment | □ |
| Send Dynamic Enhancement Segments | □ |

# STEP 4: Define Partner Profiles

4-1: In **Display IMG** page, unfold **Integration with Other SAP Components -> Interface to Global Track and Trace -> IDoc Settings**

4-2: Choose activity **Define Partner Profiles**

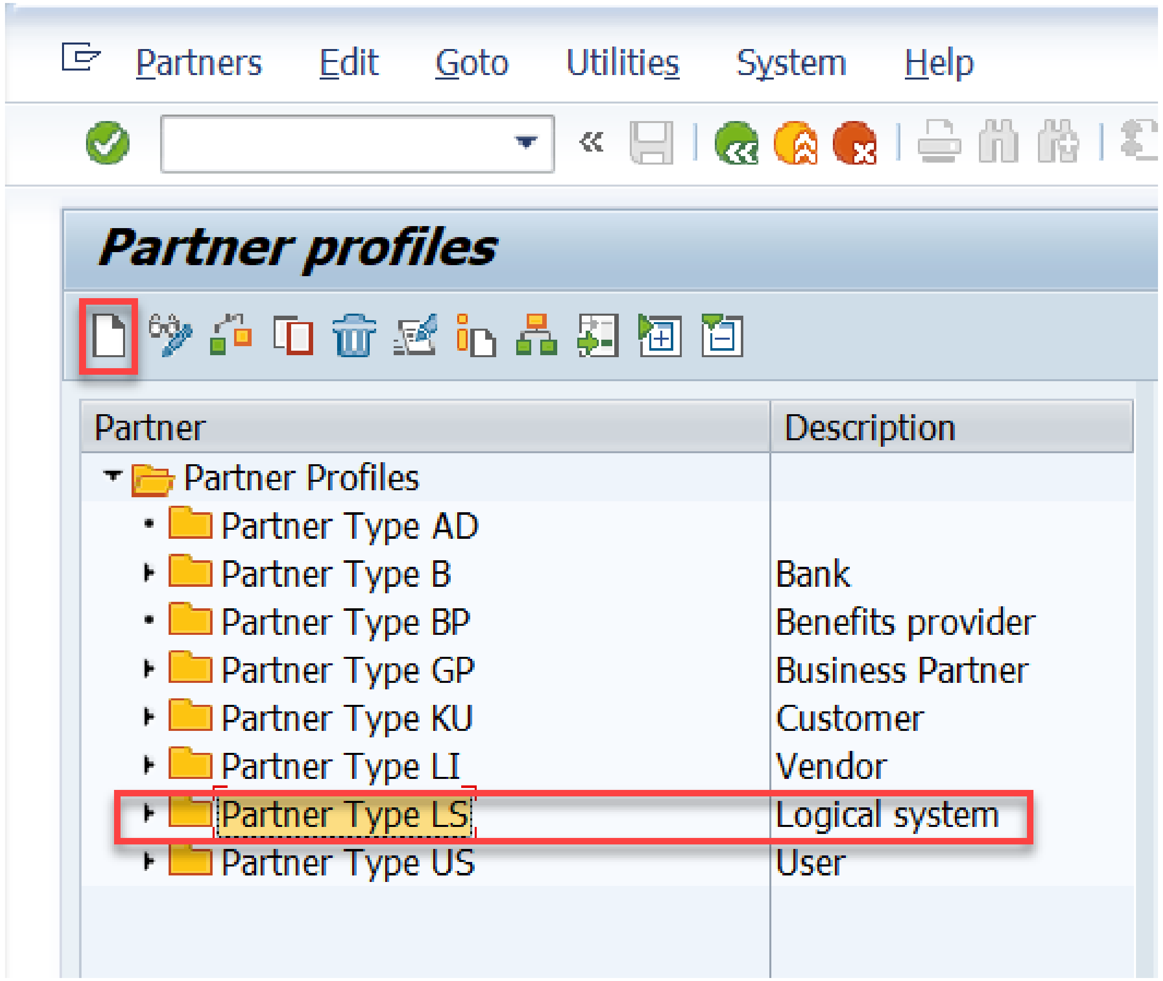


Structure

- Explosives Management for SCM Extended Warehouse Management
- CWM
- Incentive and Sales Force Management
- Joint Venture Accounting
- Public Sector Management
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
    - Define Application Interface
    - IDoc Settings
      - IDoc Settings
      - Define Ports
      - Define Partner Profiles
      - Maintain Distribution Model and Distribute Views
  - Event Management Interface
  - Supplier Relationship Management

# STEP 4: Define Partner Profiles

4-3: Choose **Partner Type LS** folder, and click **Create** to create a new partner profile



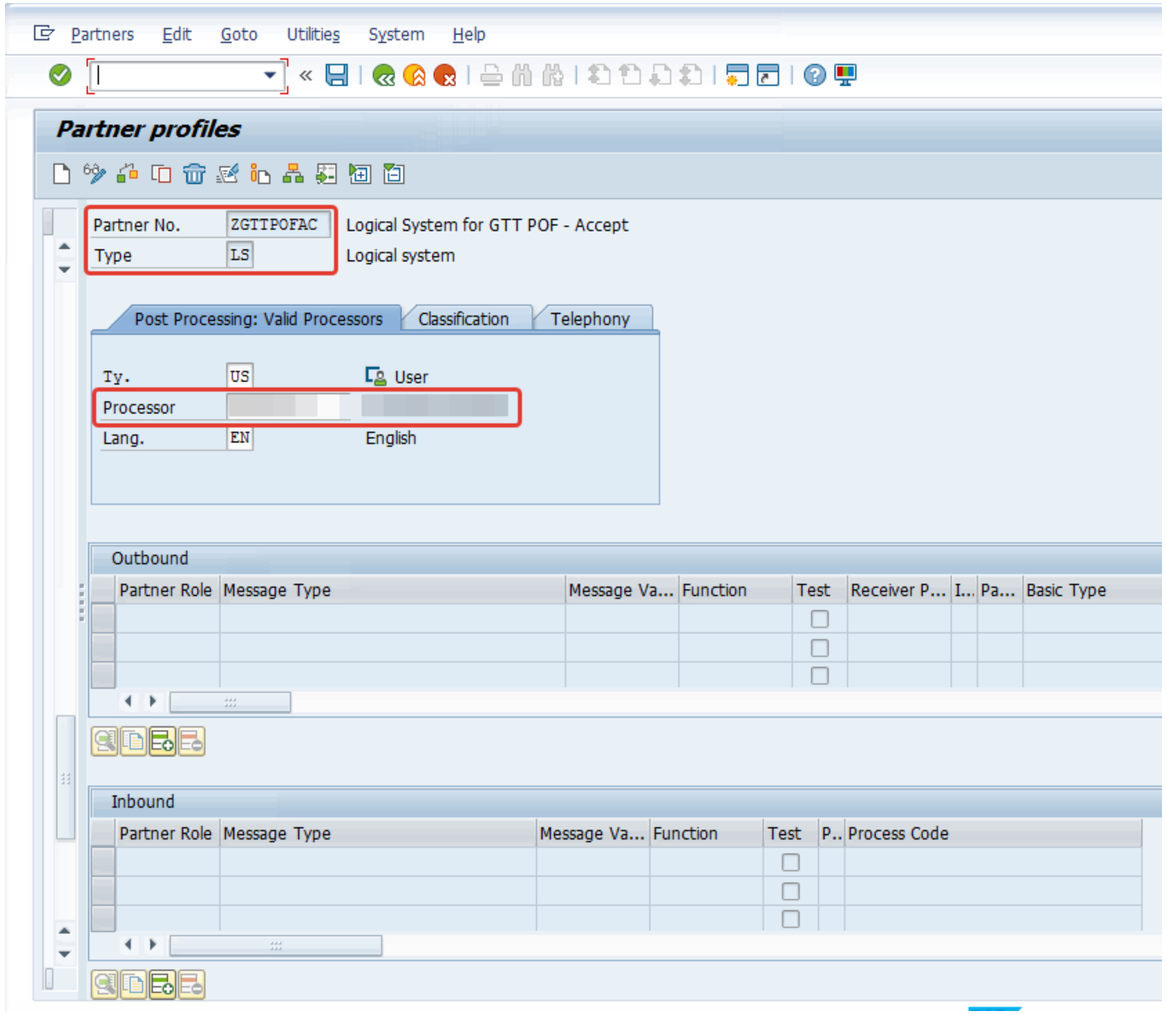
Partner profiles

| Partner | Description |
|---|---|
| Partner Profiles | |
| Partner Type AD | |
| 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 |
| Partner Type US | User |

# STEP 4: Define Partner Profiles

4-4: Fill in the **Partner No.** that you created in STEP 2

4-5: Fill in the **Processor** information



Partner profiles

| | | |
|---|---|---|
| Partner No. | ZGTTPOFAC | Logical System for GTT POF - Accept |
| Type | LS | Logical system |

| | | |
|---|---|---|
| Ty. | US | User |
| Processor | [redacted] | |
| Lang. | EN | English |

Outbound

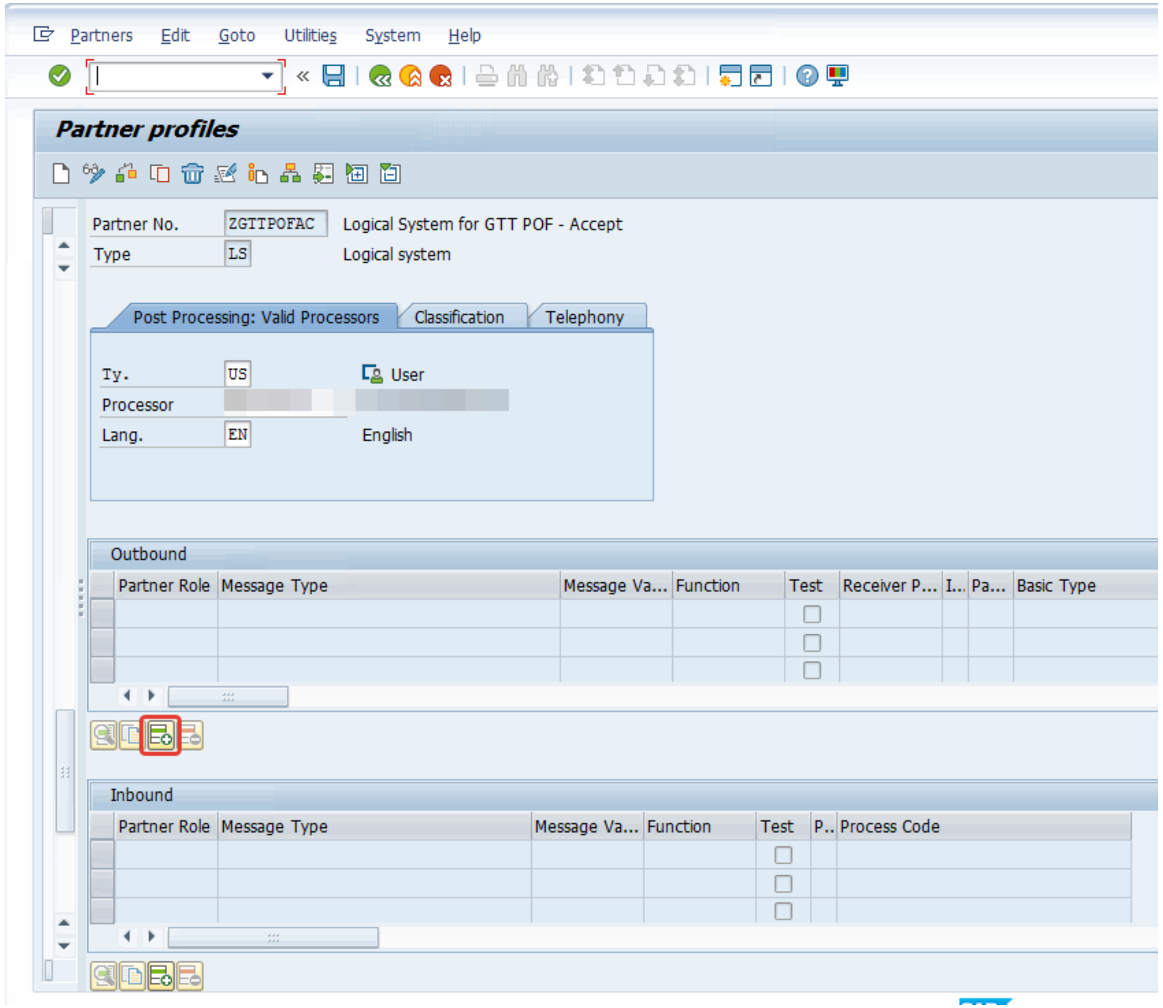
| Partner Role | Message Type | Message Va... | Function | Test | Receiver P... | I... | Pa... | Basic Type |
|---|---|---|---|---|---|---|---|---|
| | | | | □ | | | | |
| | | | | □ | | | | |
| | | | | □ | | | | |

Inbound

| Partner Role | Message Type | Message Va... | Function | Test | P.. | Process Code |
|---|---|---|---|---|---|---|
| | | | | □ | | |
| | | | | □ | | |
| | | | | □ | | |

# STEP 4: Define Partner Profiles

4-6: Click **Add** under **Outbound** box to create a new outbound parameter



**Partner profiles**

Partner No. ZGTTPOFAC Logical System for GTT POF - Accept

Type LS Logical system

Ty. US User

Processor [redacted]

Lang. EN English

Outbound

| Partner Role | Message Type | Message Va... | Function | Test | Receiver P... | I... | Pa... | Basic Type |
|---|---|---|---|---|---|---|---|---|
| | | | | □ | | | | |
| | | | | □ | | | | |
| | | | | □ | | | | |

Inbound

| Partner Role | Message Type | Message Va... | Function | Test | P... | Process Code |
|---|---|---|---|---|---|---|
| | | | | □ | | |
| | | | | □ | | |
| | | | | □ | | |

# STEP 4: Define Partner Profiles

4-7: Fill in the Message Type.

For the event:

**Message Type**: *EVMSTA*

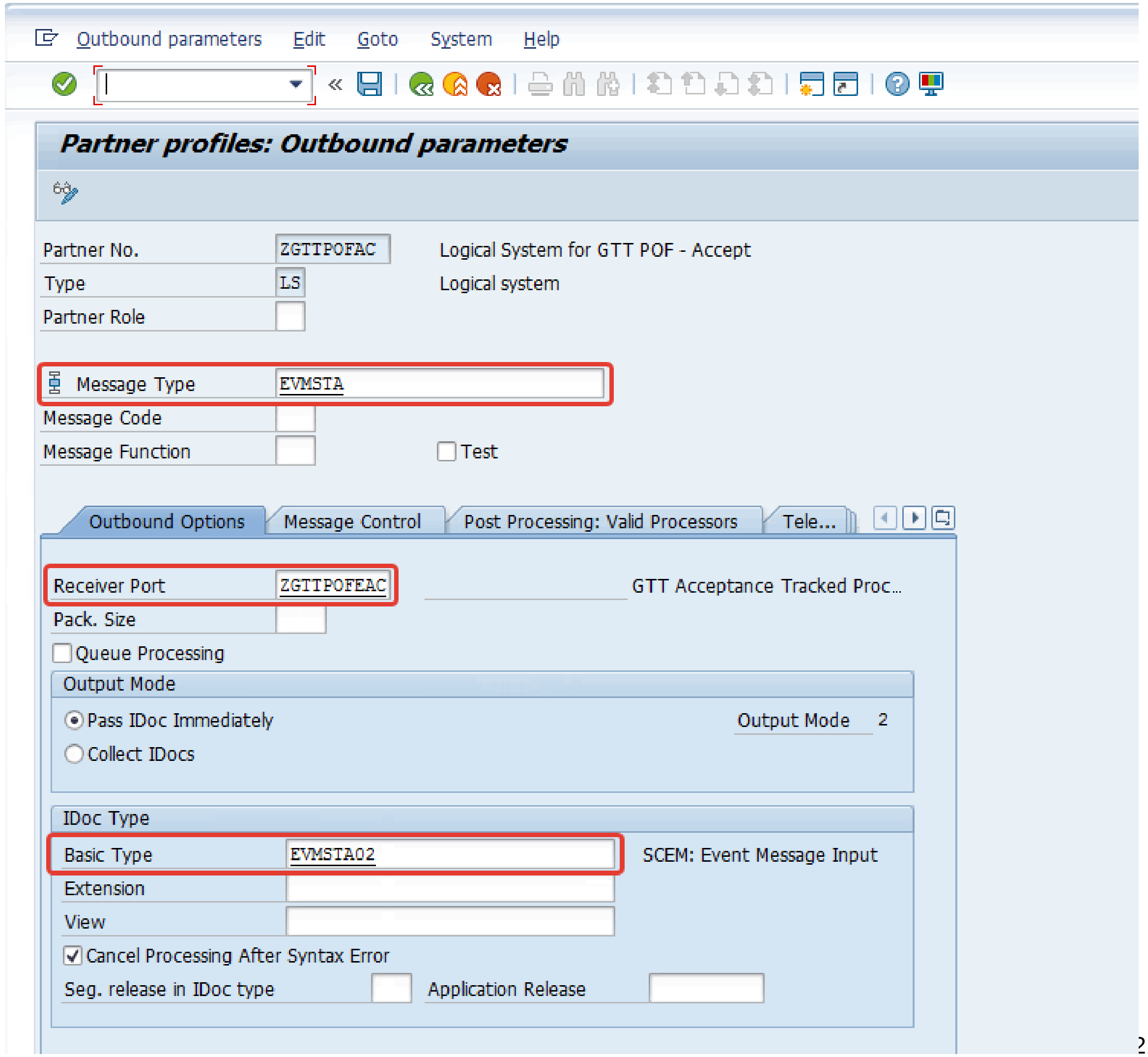
For the tracked Process:

**Message Type**: *AOPOST*

4-8: Fill in the Receiver Port, that you created in STEP 3

4-9: Save the configuration

**Caution**: In this step, you need to repeat steps 4-6 ~ 4-9 to add two outbound parameters, one for event and the other for tracked process.



**Partner profiles: Outbound parameters**

| | | |
|---|---|---|
| Partner No. | ZGTTPOFAC | Logical System for GTT POF - Accept |
| Type | LS | Logical system |
| Partner Role | | |
| Message Type | EVMSTA | |
| Message Code | | |
| Message Function | | □ Test |

Outbound Options | Message Control | Post Processing: Valid Processors | Tele...

| | | |
|---|---|---|
| Receiver Port | ZGTTPOFEAC | GTT Acceptance Tracked Proc... |
| Pack. Size | | |

□ Queue Processing

Output Mode

● Pass IDoc Immediately — Output Mode 2

○ Collect IDocs

IDoc Type

| | | |
|---|---|---|
| Basic Type | EVMSTA02 | SCEM: Event Message Input |
| Extension | | |
| View | | |

☑ Cancel Processing After Syntax Error

Seg. release in IDoc type — Application Release

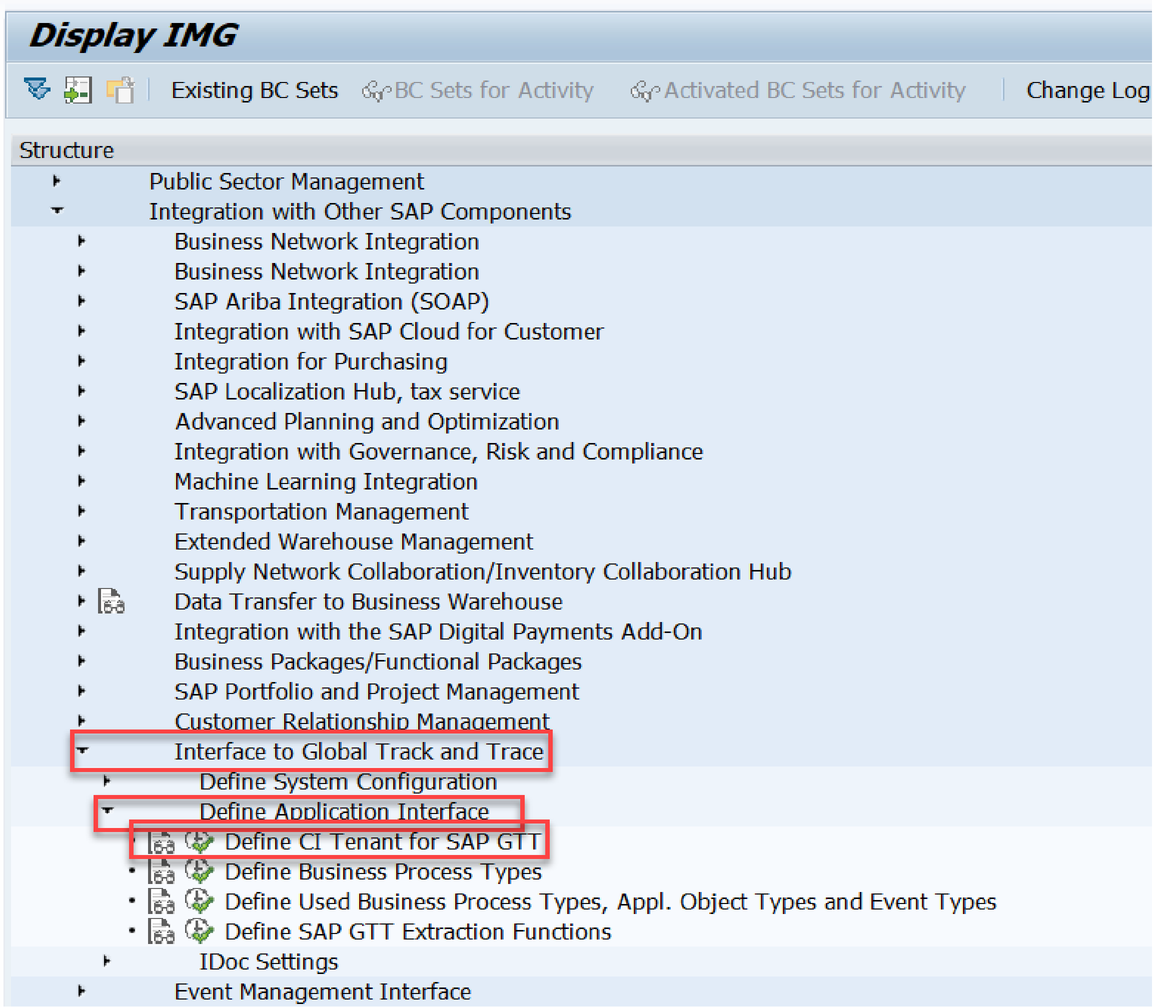
# B) Configuration and Implementation - Basic

## B2. Extractor Configuration

# STEP 5: Define CI Tenant for GTT

5-1: In **Display IMG** page, click **Integration with Other SAP Components** -> **Interface to Global Track and Trace** -> **Define Application Interface**

5-2: Choose activity **Define CI Tenant for SAP GTT**



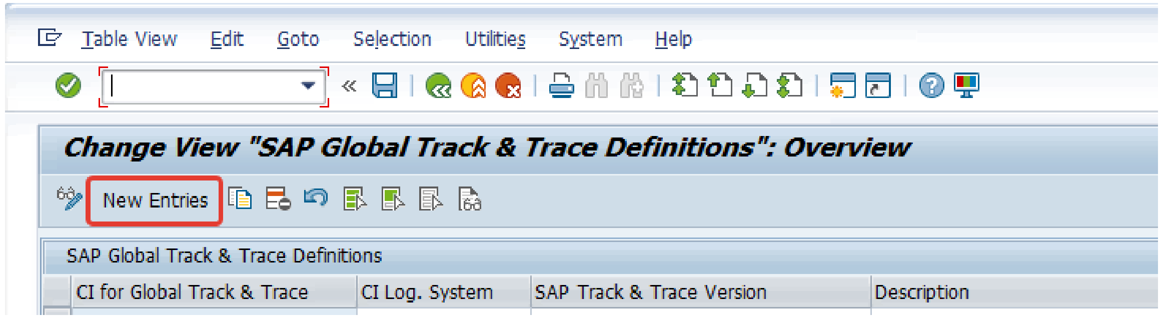
Structure

- Public Sector Management
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
    - Define Application Interface
      - Define CI Tenant for SAP GTT
      - Define Business Process Types
      - Define Used Business Process Types, Appl. Object Types and Event Types
      - Define SAP GTT Extraction Functions
    - IDoc Settings
  - Event Management Interface

# STEP 5: Define CI Tenant for GTT

5-3: Click **New Entries** to create a new CI tenant for GTT

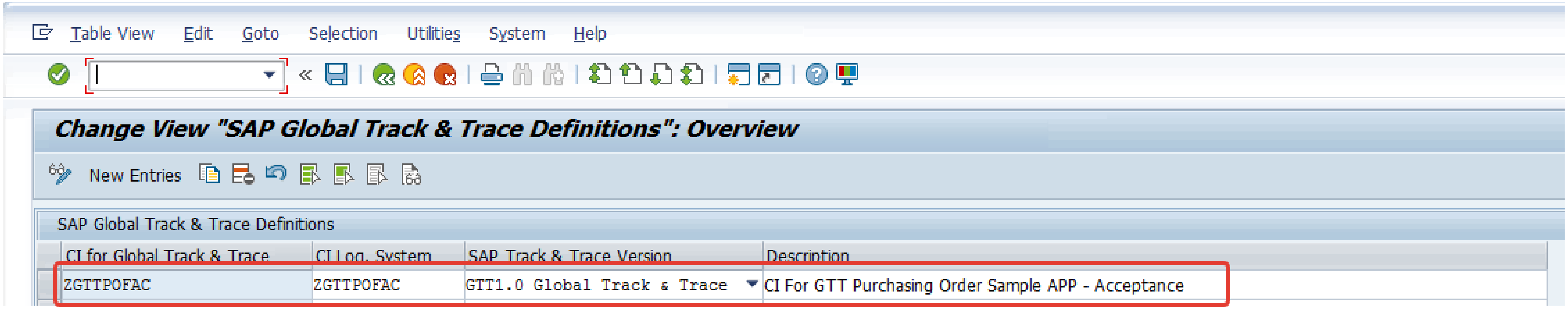
5-4: Fill in the information for the new CI tenant. The **CI Log. System** is the logical system you created in STEP 2.



Change View "SAP Global Track & Trace Definitions": Overview

SAP Global Track & Trace Definitions

| CI for Global Track & Trace | CI Log. System | SAP Track & Trace Version | Description |
|---|---|---|---|



Change View "SAP Global Track & Trace Definitions": Overview

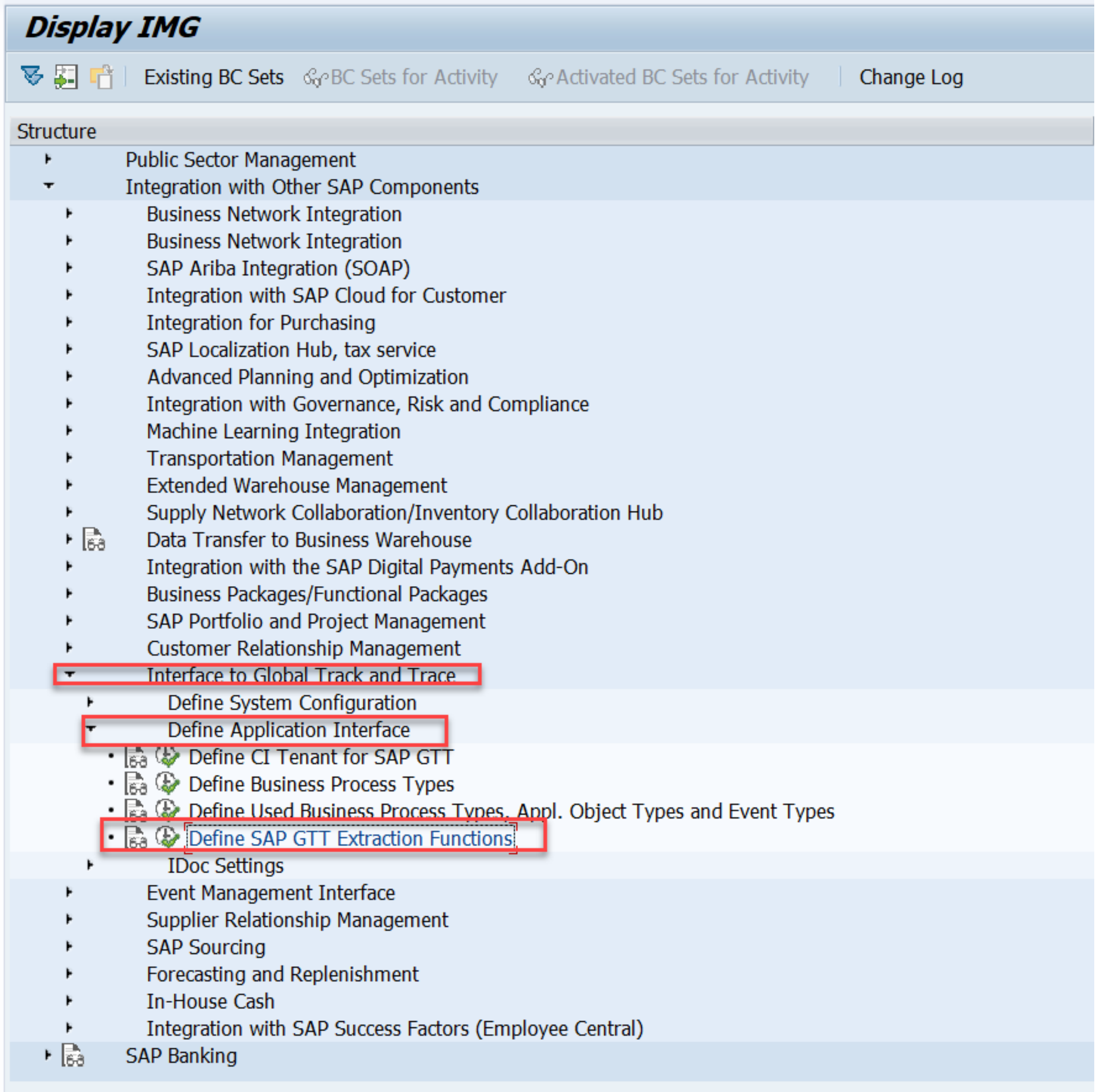
SAP Global Track & Trace Definitions

| CI for Global Track & Trace | CI Log. System | SAP Track & Trace Version | Description |
|---|---|---|---|
| ZGTTPOFAC | ZGTTPOFAC | GTT1.0 Global Track & Trace | CI For GTT Purchasing Order Sample APP - Acceptance |

# STEP 6: Define GTT Extraction Functions

6-1: In **Display IMG** page, click **Integration with Other SAP Components** -> **Interface to Global Track and Trace** -> **Define Application Interface**

6-2: Choose activity **Define SAP GTT Extraction Functions**



**Display IMG**

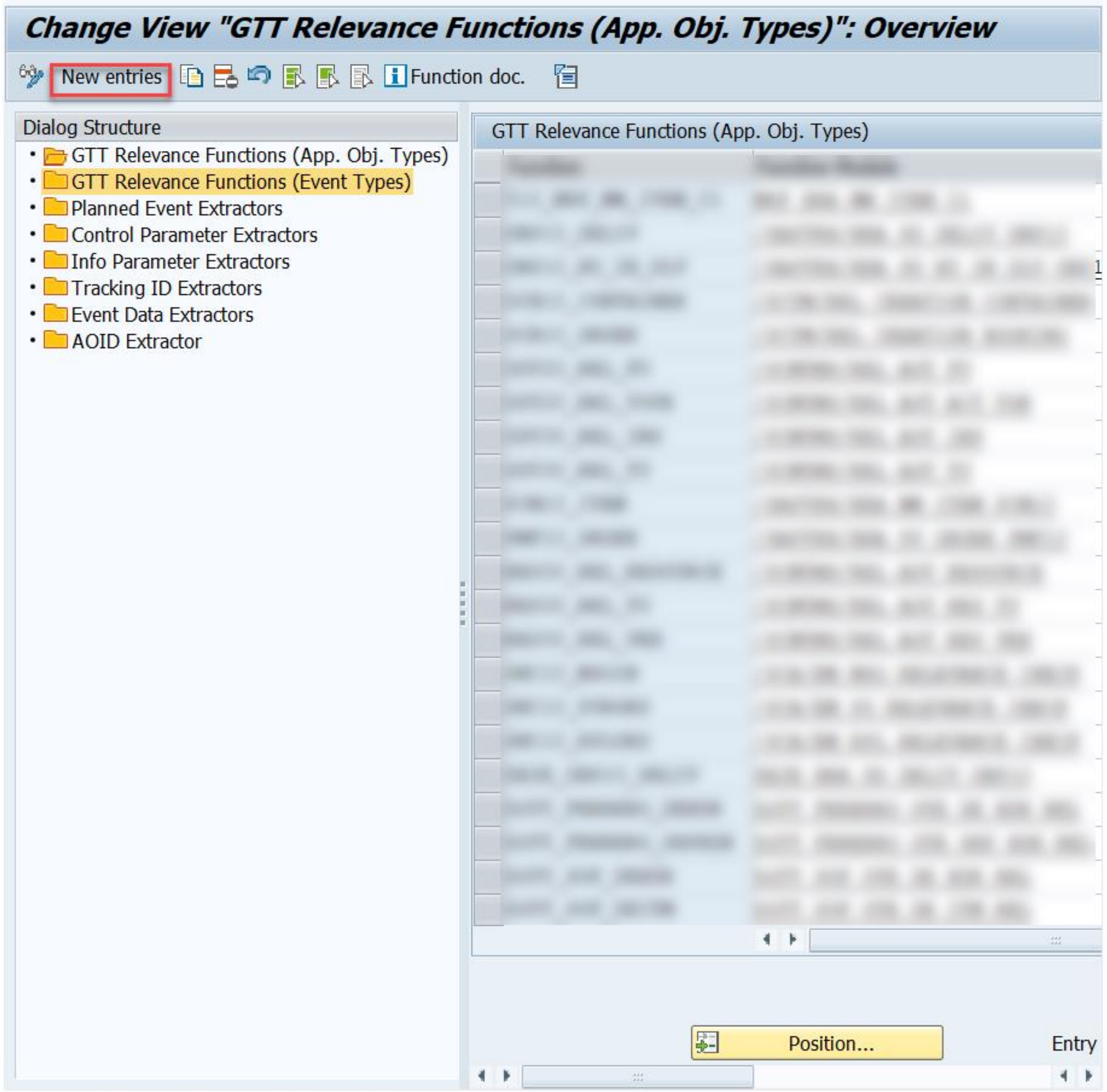
Existing BC Sets | BC Sets for Activity | Activated BC Sets for Activity | Change Log

Structure

- Public Sector Management
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
    - Define Application Interface
      - Define CI Tenant for SAP GTT
      - Define Business Process Types
      - Define Used Business Process Types, Appl. Object Types and Event Types
      - Define SAP GTT Extraction Functions
    - IDoc Settings
  - Event Management Interface
  - Supplier Relationship Management
  - SAP Sourcing
  - Forecasting and Replenishment
  - In-House Cash
  - Integration with SAP Success Factors (Employee Central)
- SAP Banking

# STEP 6: Define GTT Extraction Functions

6-3: Choose the type of Extraction Function you want to create from the **Dialog Structure,** and click **New entries**

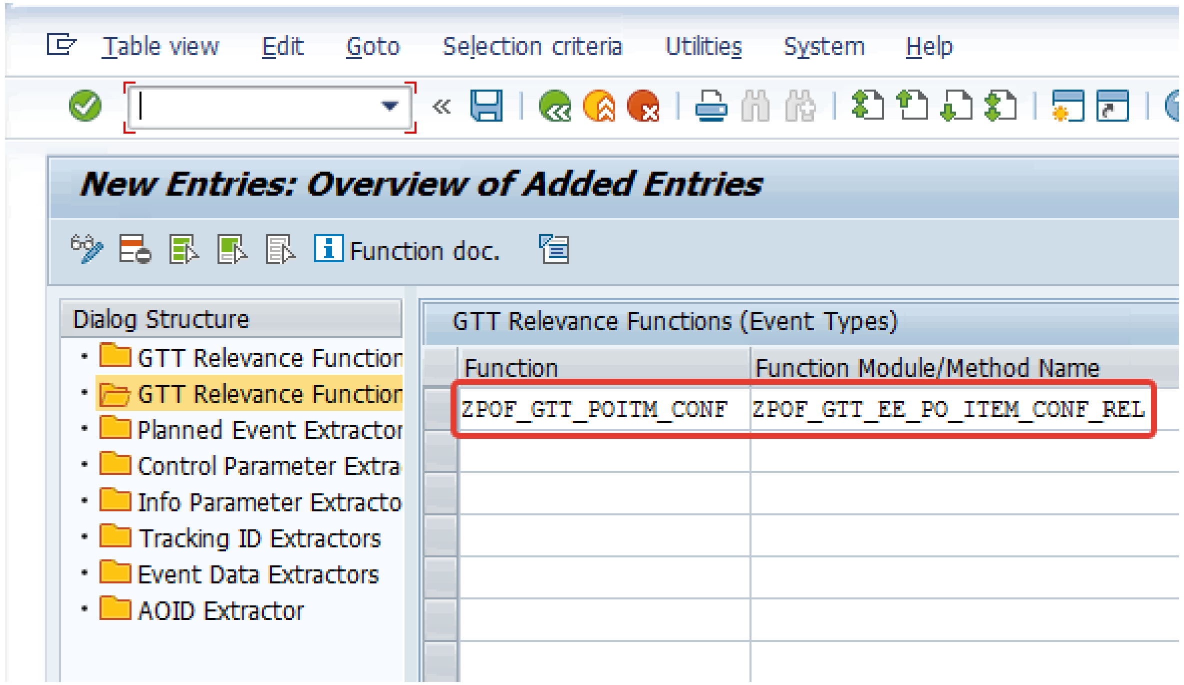


Change View "GTT Relevance Functions (App. Obj. Types)": Overview

New entries
Function doc.
Dialog Structure
GTT Relevance Functions (App. Obj. Types)
GTT Relevance Functions (Event Types)
Planned Event Extractors
Control Parameter Extractors
Info Parameter Extractors
Tracking ID Extractors
Event Data Extractors
AOID Extractor
GTT Relevance Functions (App. Obj. Types)
Position...
Entry

# STEP 6: Define GTT Extraction Functions

6-4: Input the **Function** name and **Function Module** for the newly created extraction function



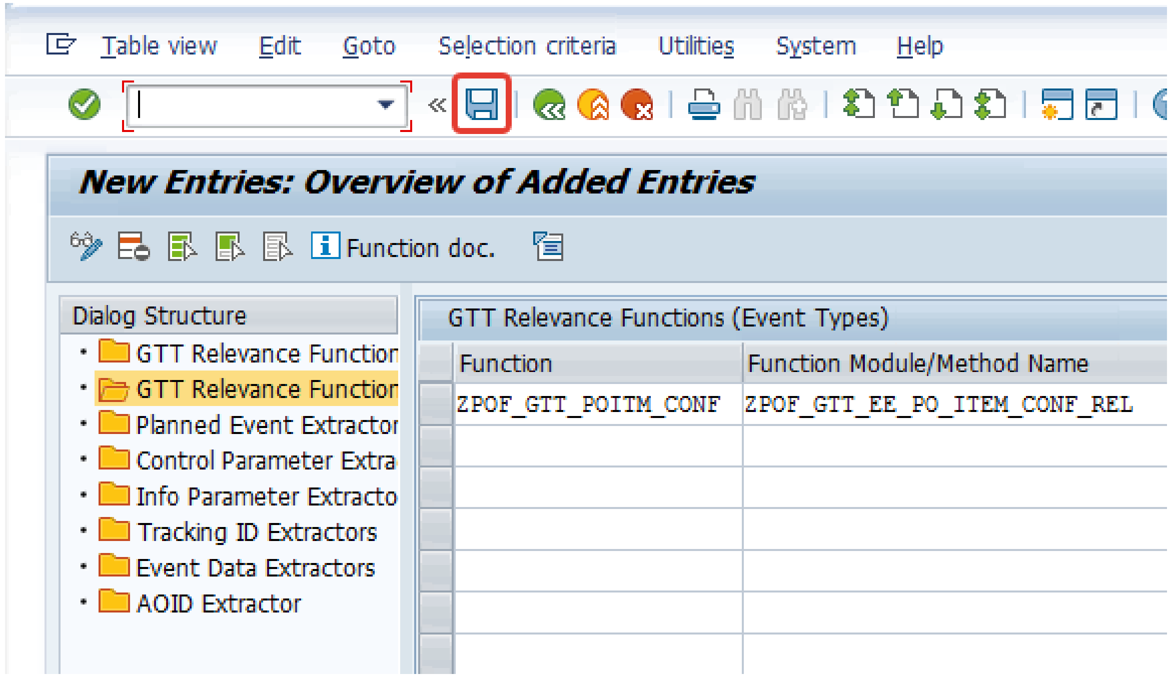
New Entries: Overview of Added Entries

GTT Relevance Functions (Event Types)

| Function | Function Module/Method Name |
|---|---|
| ZPOF_GTT_POITM_CONF | ZPOF_GTT_EE_PO_ITEM_CONF_REL |

# STEP 6: Define GTT Extraction Functions

6-5: Click **Save**



New Entries: Overview of Added Entries

Dialog Structure

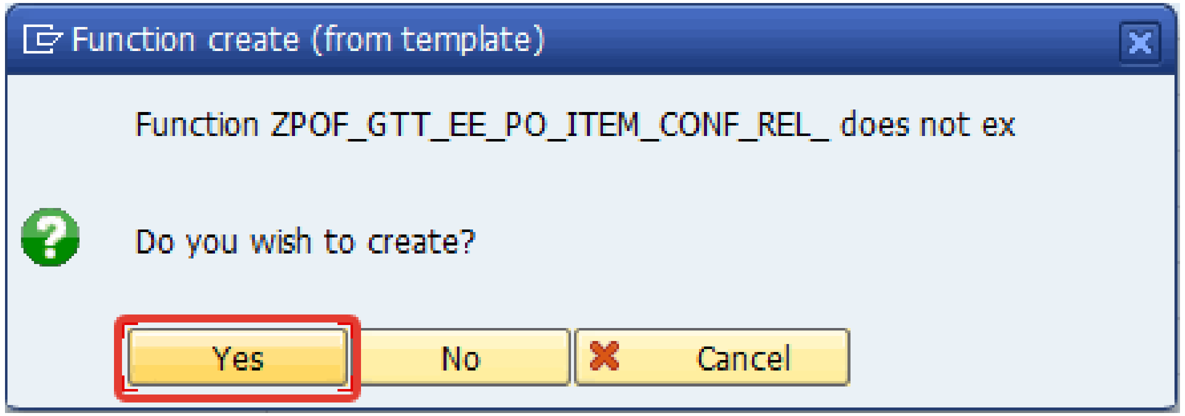
- GTT Relevance Function
- GTT Relevance Function
- Planned Event Extractor
- Control Parameter Extra
- Info Parameter Extracto
- Tracking ID Extractors
- Event Data Extractors
- AOID Extractor

GTT Relevance Functions (Event Types)

| Function | Function Module/Method Name |
|---|---|
| ZPOF_GTT_POITM_CONF | ZPOF_GTT_EE_PO_ITEM_CONF_REL |

# STEP 6: Define GTT Extraction Functions

6-6: If the function module you use to create the extraction function has not been created yet, then a dialog reminds you to create the function module. Click **Yes** in the dialog box.



Function create (from template)

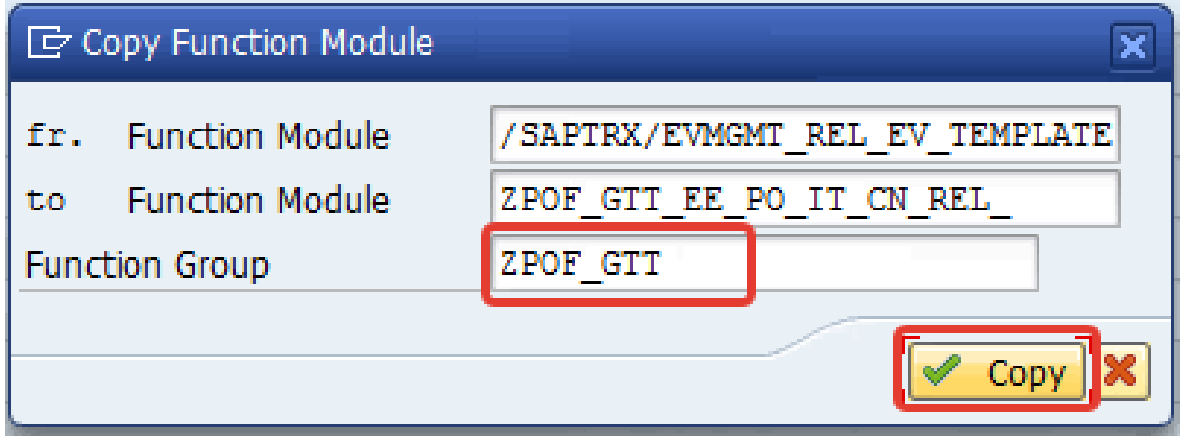
Function ZPOF_GTT_EE_PO_ITEM_CONF_REL_ does not ex

Do you wish to create?

# STEP 6: Define GTT Extraction Functions

6-7: Input the **Function Group** where the function module is to be created

6-8: Click **Copy**



Copy Function Module

fr. Function Module /SAPTRX/EVMGMT_REL_EV_TEMPLATE

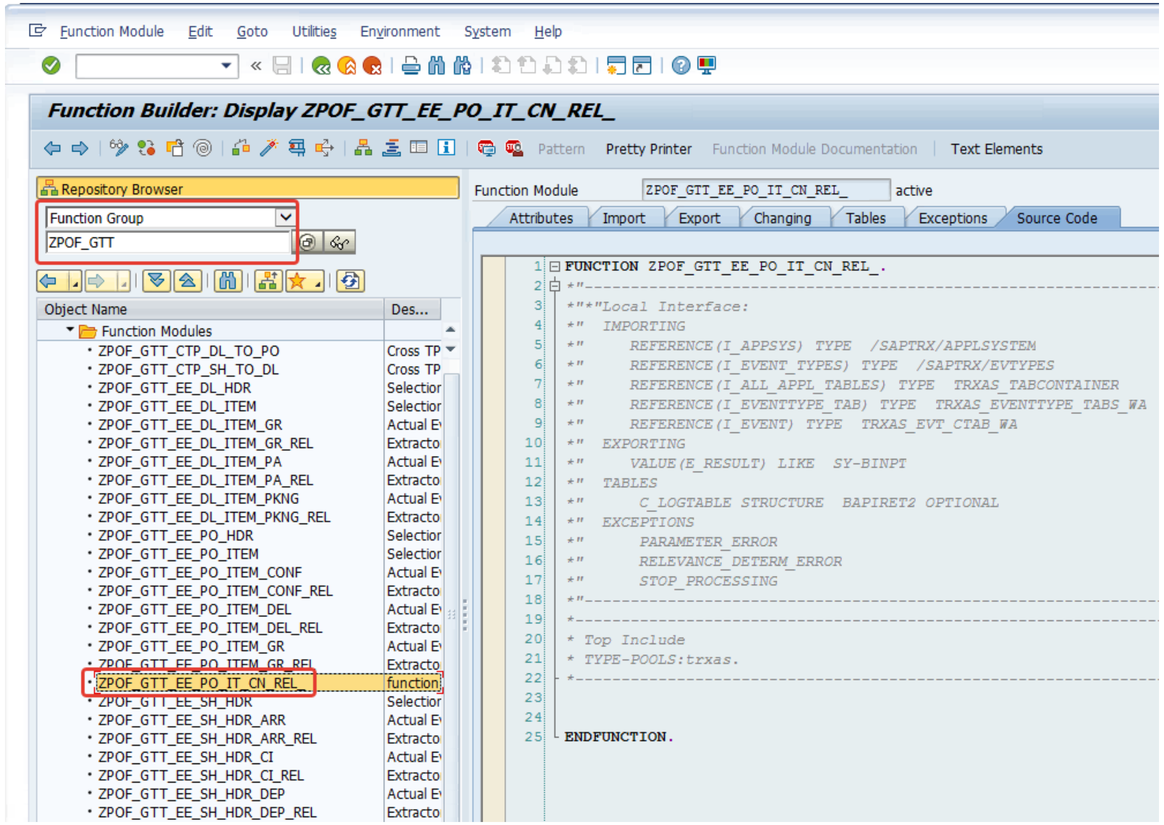
to Function Module ZPOF_GTT_EE_PO_IT_CN_REL_

Function Group ZPOF_GTT

# STEP 6: Define GTT Extraction Functions

6-9: Use T-Code SE80 to check the function module you just created

**Caution**: More information on how to implement extraction functions and the relevant sample code is introduced later.



Function Group

ZPOF_GTT

Function Module ZPOF_GTT_EE_PO_IT_CN_REL_ active

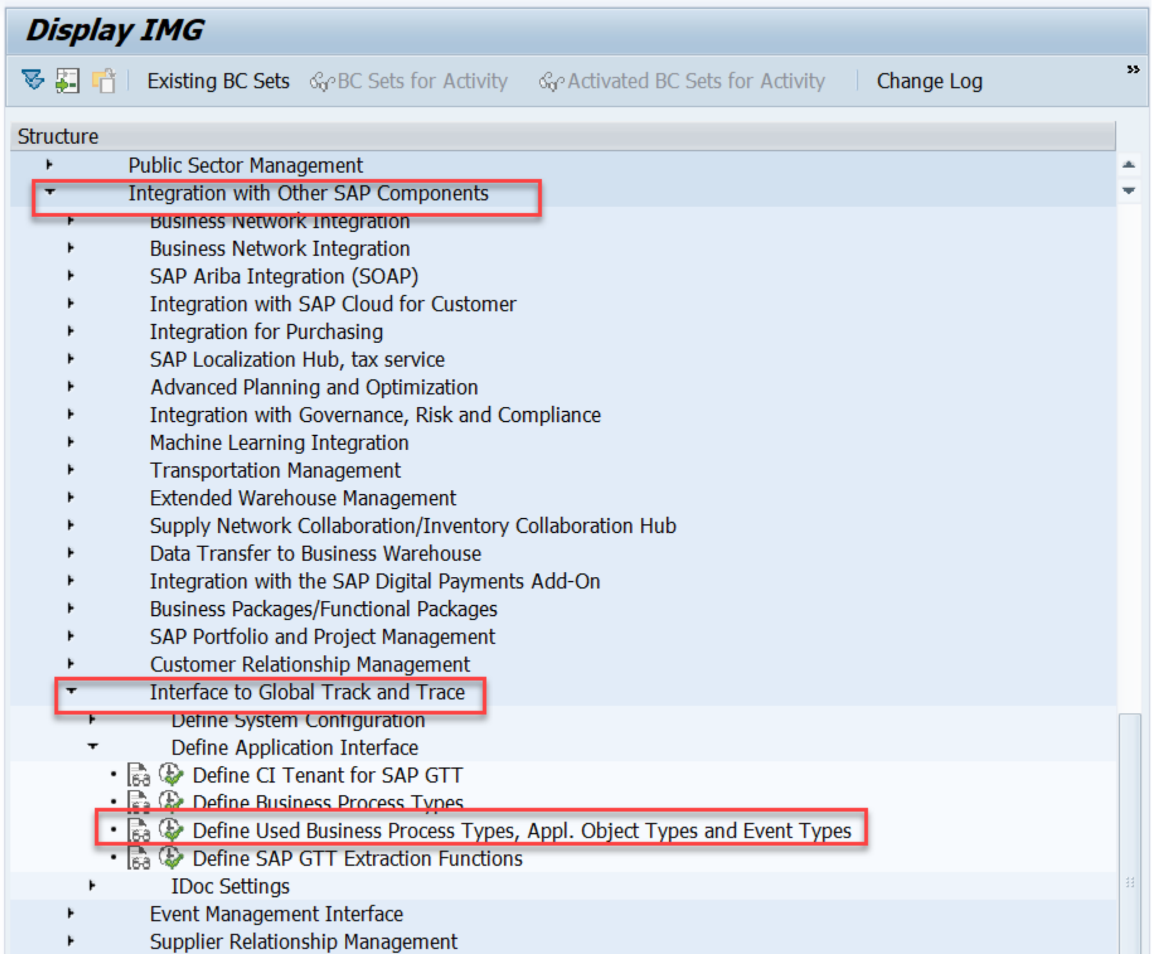
| Object Name | Des... |
|---|---|
| Function Modules | |
| ZPOF_GTT_CTP_DL_TO_PO | Cross TP |
| ZPOF_GTT_CTP_SH_TO_DL | Cross TP |
| ZPOF_GTT_EE_DL_HDR | Selectior |
| ZPOF_GTT_EE_DL_ITEM | Selectior |
| ZPOF_GTT_EE_DL_ITEM_GR | Actual E |
| ZPOF_GTT_EE_DL_ITEM_GR_REL | Extracto |
| ZPOF_GTT_EE_DL_ITEM_PA | Actual E |
| ZPOF_GTT_EE_DL_ITEM_PA_REL | Extracto |
| ZPOF_GTT_EE_DL_ITEM_PKNG | Actual E |
| ZPOF_GTT_EE_DL_ITEM_PKNG_REL | Extracto |
| ZPOF_GTT_EE_PO_HDR | Selectior |
| ZPOF_GTT_EE_PO_ITEM | Selectior |
| ZPOF_GTT_EE_PO_ITEM_CONF | Actual E |
| ZPOF_GTT_EE_PO_ITEM_CONF_REL | Extracto |
| ZPOF_GTT_EE_PO_ITEM_DEL | Actual E |
| ZPOF_GTT_EE_PO_ITEM_DEL_REL | Extracto |
| ZPOF_GTT_EE_PO_ITEM_GR | Actual E |
| ZPOF_GTT_EE_PO_ITEM_GR_REL | Extracto |
| ZPOF_GTT_EE_PO_IT_CN_REL_ | function |
| ZPOF_GTT_EE_SH_HDR | Selectior |
| ZPOF_GTT_EE_SH_HDR_ARR | Actual E |
| ZPOF_GTT_EE_SH_HDR_ARR_REL | Extracto |
| ZPOF_GTT_EE_SH_HDR_CI | Actual E |
| ZPOF_GTT_EE_SH_HDR_CI_REL | Extracto |
| ZPOF_GTT_EE_SH_HDR_DEP | Actual E |
| ZPOF_GTT_EE_SH_HDR_DEP_REL | Extracto |

```
FUNCTION ZPOF_GTT_EE_PO_IT_CN_REL_.
*"----------------------------------------------------------------------
*"*"Local Interface:
*"  IMPORTING
*"     REFERENCE(I_APPSYS) TYPE  /SAPTRX/APPLSYSTEM
*"     REFERENCE(I_EVENT_TYPES) TYPE  /SAPTRX/EVTYPES
*"     REFERENCE(I_ALL_APPL_TABLES) TYPE  TRXAS_TABCONTAINER
*"     REFERENCE(I_EVENTTYPE_TAB) TYPE  TRXAS_EVENTTYPE_TABS_WA
*"     REFERENCE(I_EVENT) TYPE  TRXAS_EVT_CTAB_WA
*"  EXPORTING
*"     VALUE(E_RESULT) LIKE  SY-BINPT
*"  TABLES
*"      C_LOGTABLE STRUCTURE  BAPIRET2 OPTIONAL
*"  EXCEPTIONS
*"      PARAMETER_ERROR
*"      RELEVANCE_DETERM_ERROR
*"      STOP_PROCESSING
*"----------------------------------------------------------------------
*-----------------------------------------------------------------------
* Top Include
* TYPE-POOLS:trxas.
*-----------------------------------------------------------------------


ENDFUNCTION.
```

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-1: In **Display IMG** page, click **Integration with Other SAP Components** -> **Interface to Global Track and Trace** -> **Define Application Interface**

7-2: Choose activity **Define Used Business Process Types, Appl. Object Types and Event Types**



Display IMG

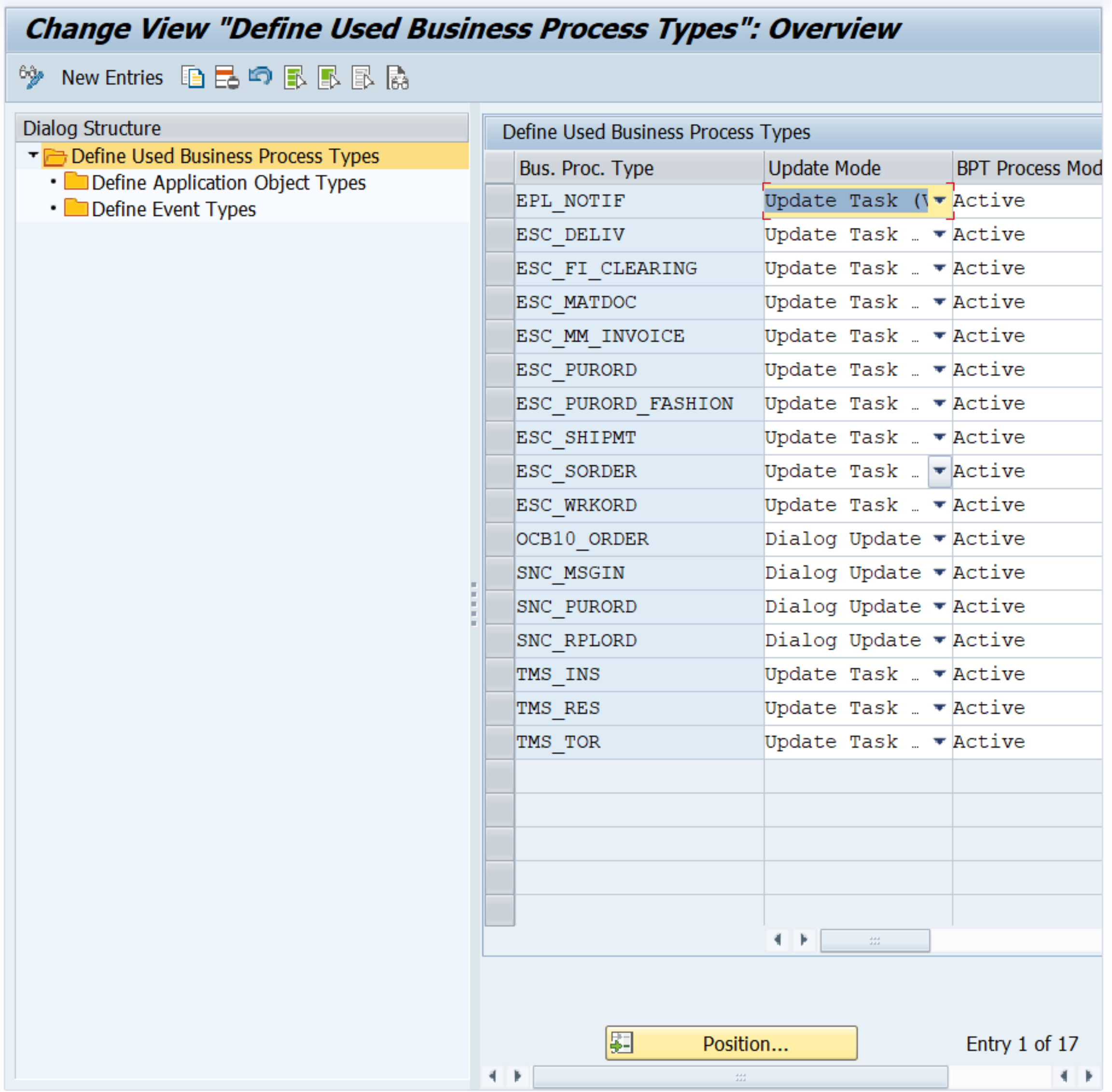
Structure
- Public Sector Management
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
    - Define Application Interface
      - Define CI Tenant for SAP GTT
      - Define Business Process Types
      - Define Used Business Process Types, Appl. Object Types and Event Types
      - Define SAP GTT Extraction Functions
    - IDoc Settings
  - Event Management Interface
  - Supplier Relationship Management

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

You can create event types and application object types for each business process type.

In the following:

- Steps 7-3 to 7-10 demonstrate how to create an *Event Type* for a given business process type
- Steps 7-11 to 7-21 demonstrate how to create an *Application Object Type* for a given business process type



Change View "Define Used Business Process Types": Overview

Dialog Structure
- Define Used Business Process Types
  - Define Application Object Types
  - Define Event Types

Define Used Business Process Types

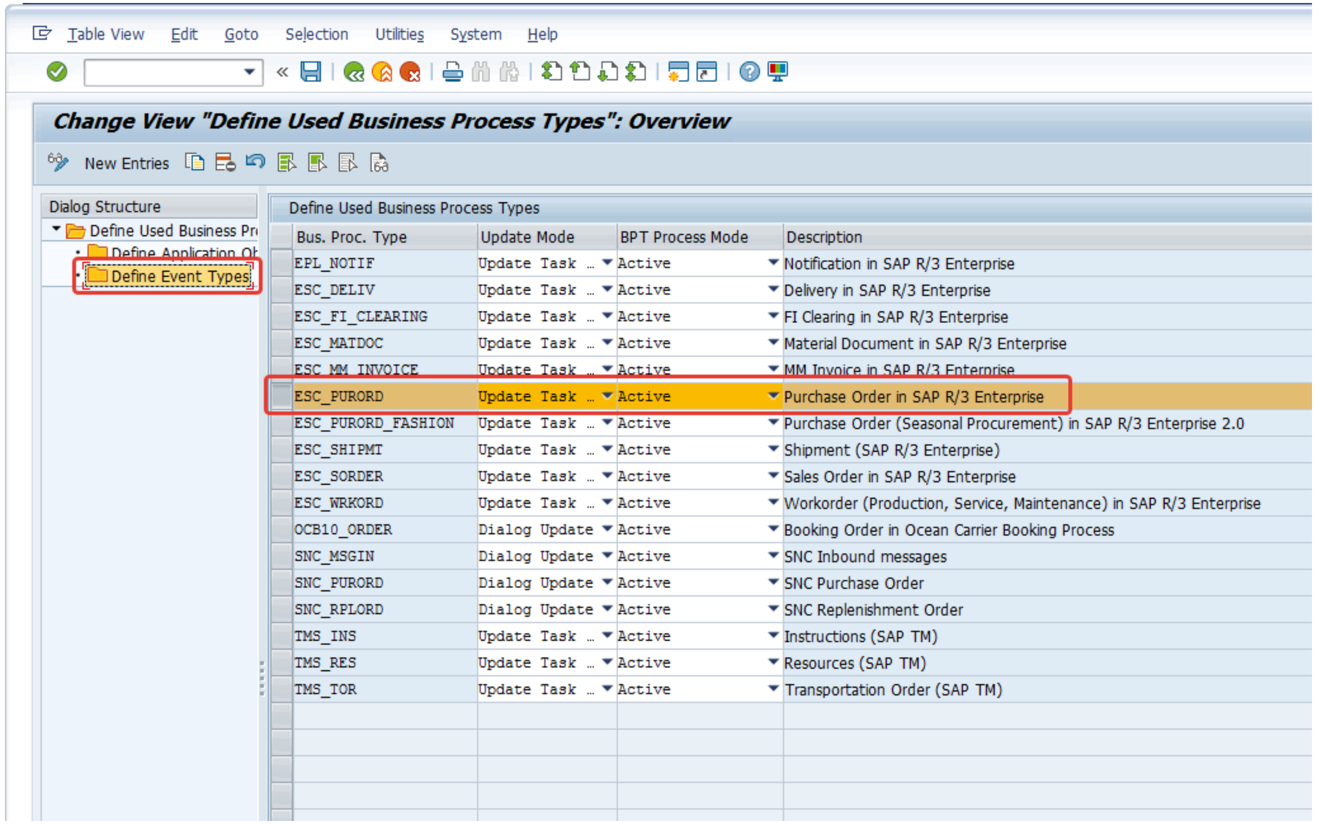
| Bus. Proc. Type | Update Mode | BPT Process Mod |
|---|---|---|
| EPL_NOTIF | Update Task (V | Active |
| ESC_DELIV | Update Task … | Active |
| ESC_FI_CLEARING | Update Task … | Active |
| ESC_MATDOC | Update Task … | Active |
| ESC_MM_INVOICE | Update Task … | Active |
| ESC_PURORD | Update Task … | Active |
| ESC_PURORD_FASHION | Update Task … | Active |
| ESC_SHIPMT | Update Task … | Active |
| ESC_SORDER | Update Task … | Active |
| ESC_WRKORD | Update Task … | Active |
| OCB10_ORDER | Dialog Update | Active |
| SNC_MSGIN | Dialog Update | Active |
| SNC_PURORD | Dialog Update | Active |
| SNC_RPLORD | Dialog Update | Active |
| TMS_INS | Update Task … | Active |
| TMS_RES | Update Task … | Active |
| TMS_TOR | Update Task … | Active |

Entry 1 of 17

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-3: Choose the business process type from the **Define Used Business Process Types** on the right side

7-4: Double click **Define Event Types**



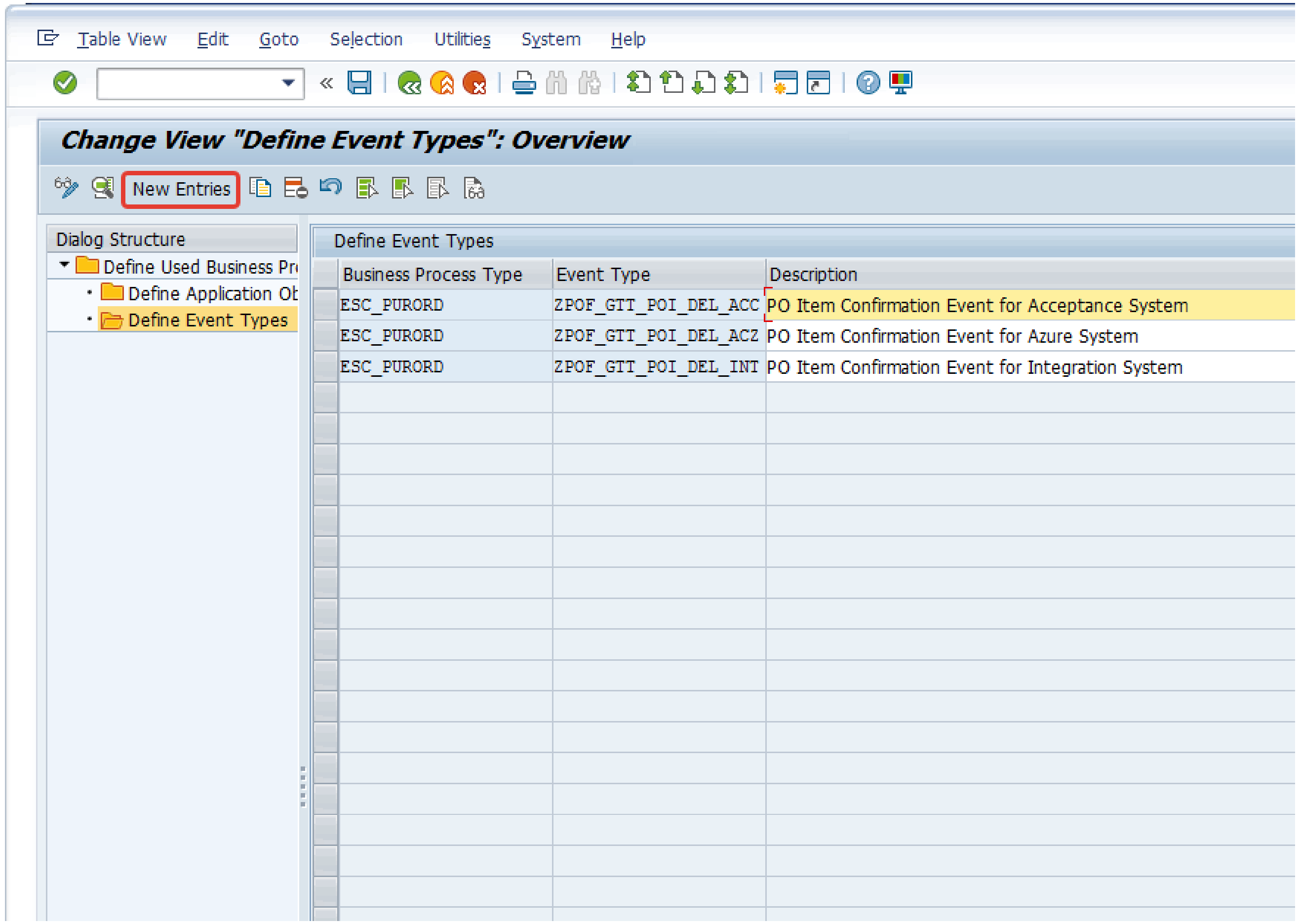
Change View "Define Used Business Process Types": Overview

Define Used Business Process Types

| Bus. Proc. Type | Update Mode | BPT Process Mode | Description |
|---|---|---|---|
| EPL_NOTIF | Update Task ... | Active | Notification in SAP R/3 Enterprise |
| ESC_DELIV | Update Task ... | Active | Delivery in SAP R/3 Enterprise |
| ESC_FI_CLEARING | Update Task ... | Active | FI Clearing in SAP R/3 Enterprise |
| ESC_MATDOC | Update Task ... | Active | Material Document in SAP R/3 Enterprise |
| ESC_MM_INVOICE | Update Task ... | Active | MM Invoice in SAP R/3 Enterprise |
| ESC_PURORD | Update Task ... | Active | Purchase Order in SAP R/3 Enterprise |
| ESC_PURORD_FASHION | Update Task ... | Active | Purchase Order (Seasonal Procurement) in SAP R/3 Enterprise 2.0 |
| ESC_SHIPMT | Update Task ... | Active | Shipment (SAP R/3 Enterprise) |
| ESC_SORDER | Update Task ... | Active | Sales Order in SAP R/3 Enterprise |
| ESC_WRKORD | Update Task ... | Active | Workorder (Production, Service, Maintenance) in SAP R/3 Enterprise |
| OCB10_ORDER | Dialog Update | Active | Booking Order in Ocean Carrier Booking Process |
| SNC_MSGIN | Dialog Update | Active | SNC Inbound messages |
| SNC_PURORD | Dialog Update | Active | SNC Purchase Order |
| SNC_RPLORD | Dialog Update | Active | SNC Replenishment Order |
| TMS_INS | Update Task ... | Active | Instructions (SAP TM) |
| TMS_RES | Update Task ... | Active | Resources (SAP TM) |
| TMS_TOR | Update Task ... | Active | Transportation Order (SAP TM) |

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-5: Click **New Entries** to create a new event type



Change View "Define Event Types": Overview

Define Event Types

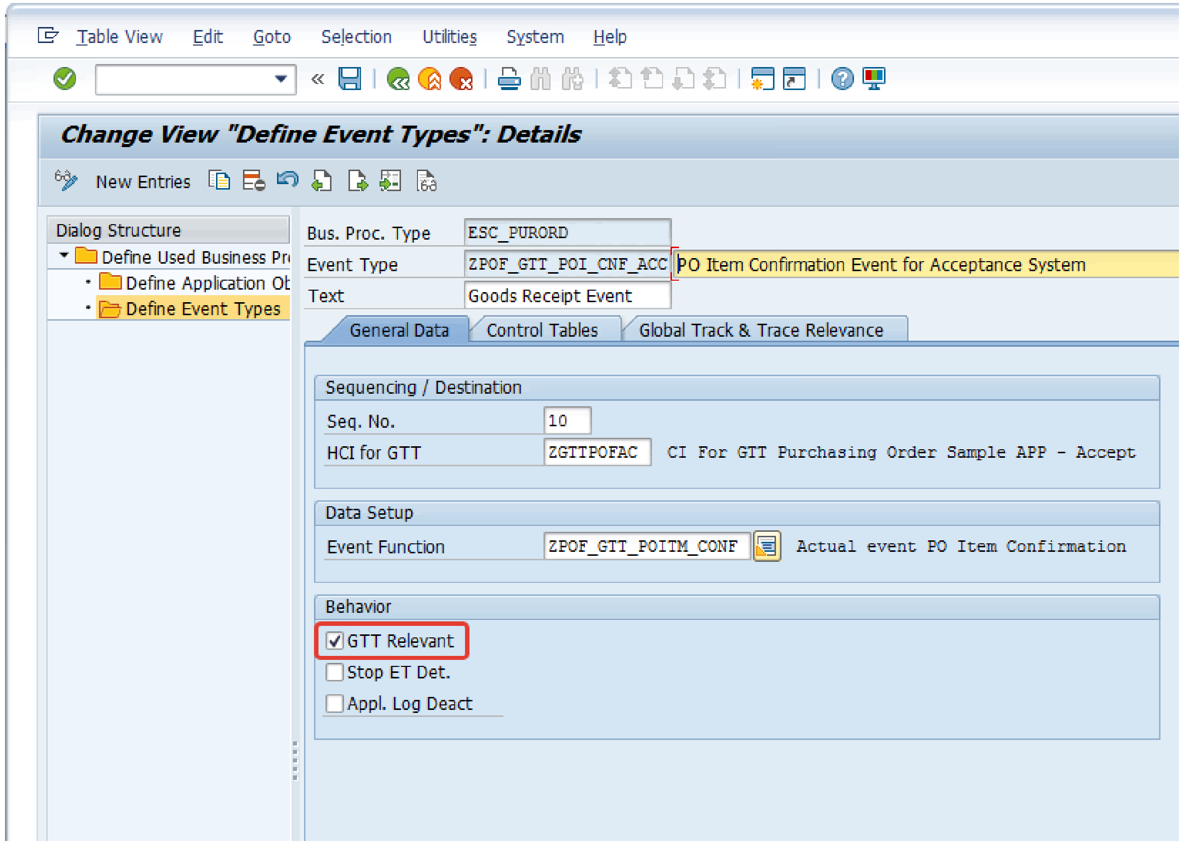
| Business Process Type | Event Type | Description |
|---|---|---|
| ESC_PURORD | ZPOF_GTT_POI_DEL_ACC | PO Item Confirmation Event for Acceptance System |
| ESC_PURORD | ZPOF_GTT_POI_DEL_ACZ | PO Item Confirmation Event for Azure System |
| ESC_PURORD | ZPOF_GTT_POI_DEL_INT | PO Item Confirmation Event for Integration System |

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-6: Fill in the **Event Type** and **Text** fields

7-7: Fill in the information required in the **General Data** tab. **HCI for GTT** is the CI Tenant you created in STEP 5. **Event Function** is the extractor function you created in STEP 6.

7-8: Check **GTT Relevant**



Change View "Define Event Types": Details

Bus. Proc. Type ESC_PURORD

Event Type ZPOF_GTT_POI_CNF_ACC PO Item Confirmation Event for Acceptance System

Text Goods Receipt Event

Sequencing / Destination

Seq. No. 10

HCI for GTT ZGTTPOFAC CI For GTT Purchasing Order Sample APP - Accept

Data Setup

Event Function ZPOF_GTT_POITM_CONF Actual event PO Item Confirmation

Behavior

☑ GTT Relevant

□ Stop ET Det.

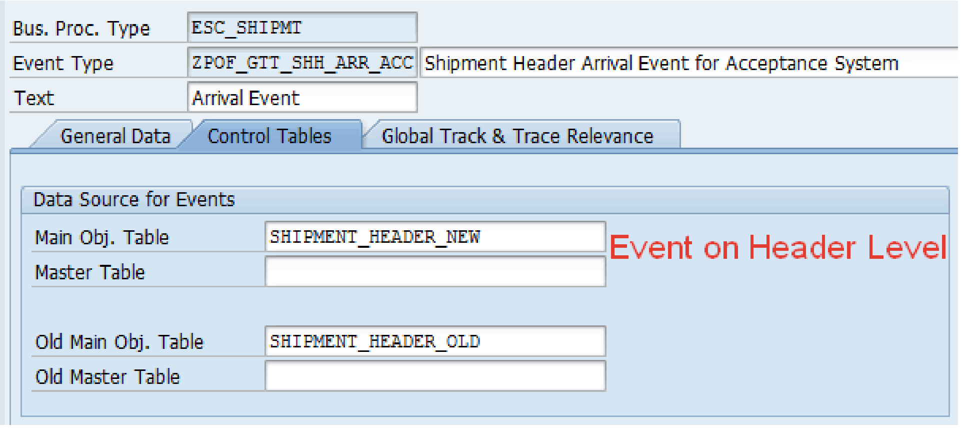
□ Appl. Log Deact

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-9: Fill in the **Main Object Table** and **Master Table**.

**Caution:**

If the event type or application object type is on header level, then you only need to assign the **Main Object Table**. Otherwise, if the event type or application object type is on item level, then you need to assign the **Main Object Table** and **Master Table**, and assign the reference between the **Main Object Table** and **Master Table**.

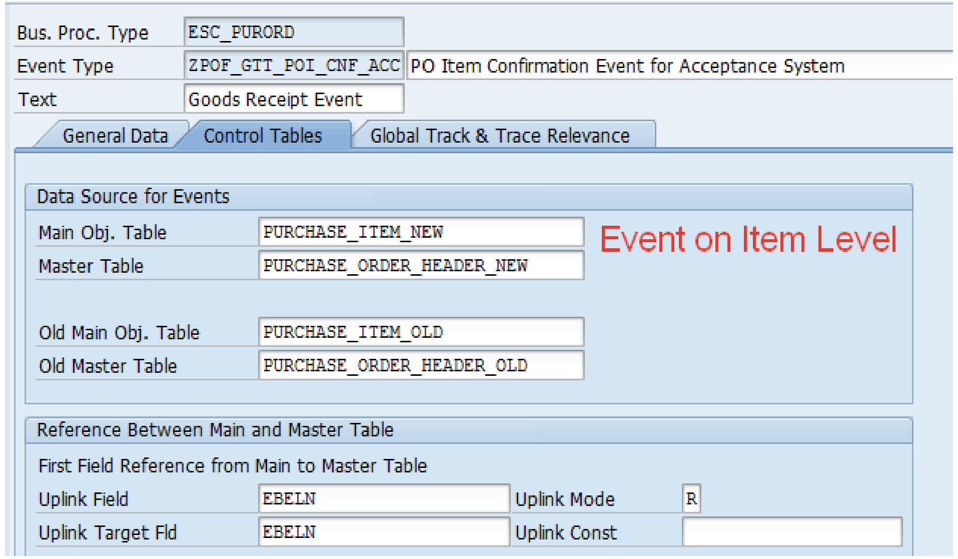


| | | |
|---|---|---|
| Bus. Proc. Type | ESC_SHIPMT | |
| Event Type | ZPOF_GTT_SHH_ARR_ACC | Shipment Header Arrival Event for Acceptance System |
| Text | Arrival Event | |

Data Source for Events

| | |
|---|---|
| Main Obj. Table | SHIPMENT_HEADER_NEW |
| Master Table | |
| Old Main Obj. Table | SHIPMENT_HEADER_OLD |
| Old Master Table | |

Event on Header Level



| | | |
|---|---|---|
| Bus. Proc. Type | ESC_PURORD | |
| Event Type | ZPOF_GTT_POI_CNF_ACC | PO Item Confirmation Event for Acceptance System |
| Text | Goods Receipt Event | |

Data Source for Events

| | |
|---|---|
| Main Obj. Table | PURCHASE_ITEM_NEW |
| Master Table | PURCHASE_ORDER_HEADER_NEW |
| Old Main Obj. Table | PURCHASE_ITEM_OLD |
| Old Master Table | PURCHASE_ORDER_HEADER_OLD |

Event on Item Level

Reference Between Main and Master Table

First Field Reference from Main to Master Table

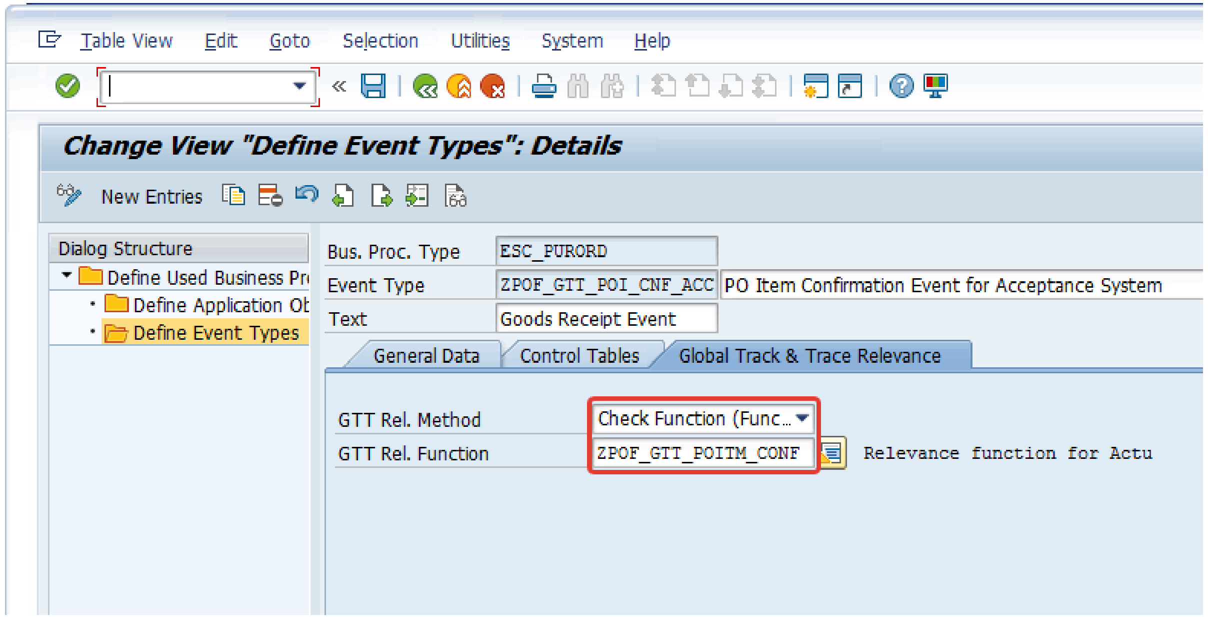
| | | | |
|---|---|---|---|
| Uplink Field | EBELN | Uplink Mode | R |
| Uplink Target Fld | EBELN | Uplink Const | |

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-10: In the **Global Track & Trace Relevance** tab, choose the **GTT Relevance Method** you need.

If you choose the **GTT Relevance Method** *Check Function*, then you need to define a relevance function according to STEP 6, and fill in the relevance function name here.

Click **Save**.



Bus. Proc. Type: ESC_PURORD

Event Type: ZPOF_GTT_POI_CNF_ACC PO Item Confirmation Event for Acceptance System

Text: Goods Receipt Event

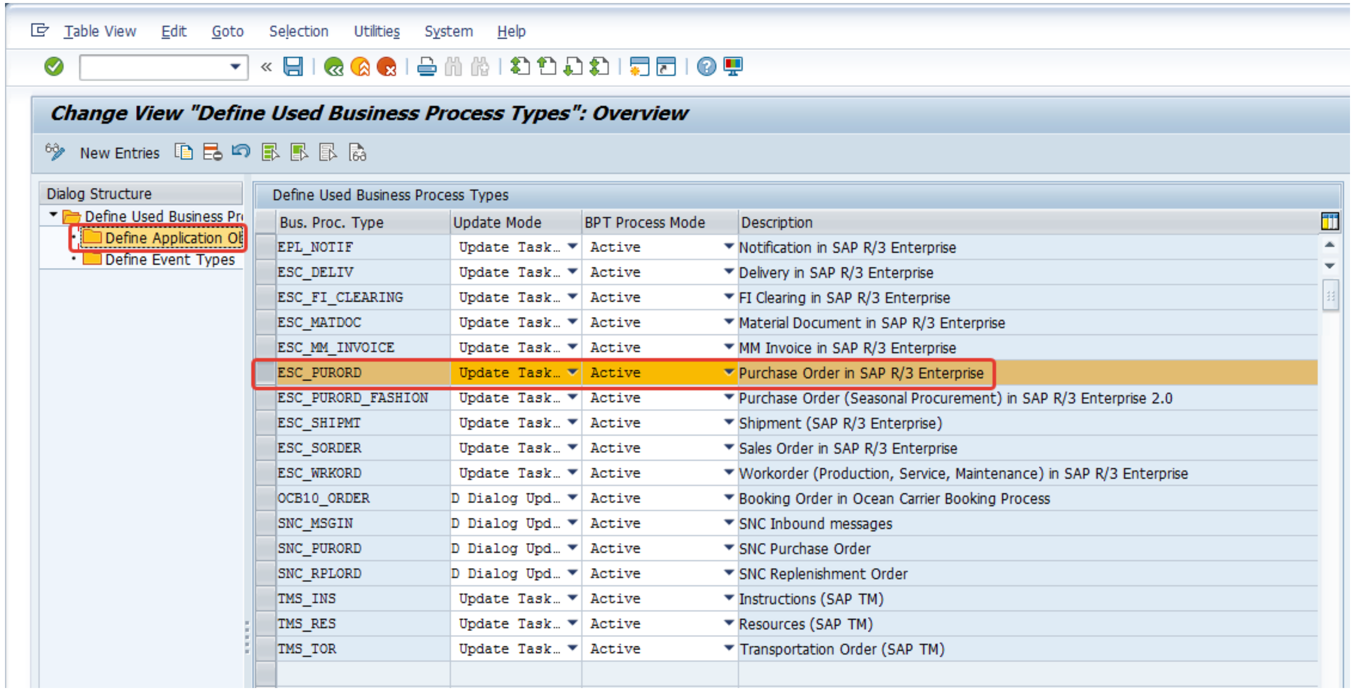
GTT Rel. Method: Check Function (Func...

GTT Rel. Function: ZPOF_GTT_POITM_CONF Relevance function for Actu

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-11: Choose the business process type from the **Define Used Business Process Types** on the right side

7-12: Double click **Define Application Object Types**



Change View "Define Used Business Process Types": Overview

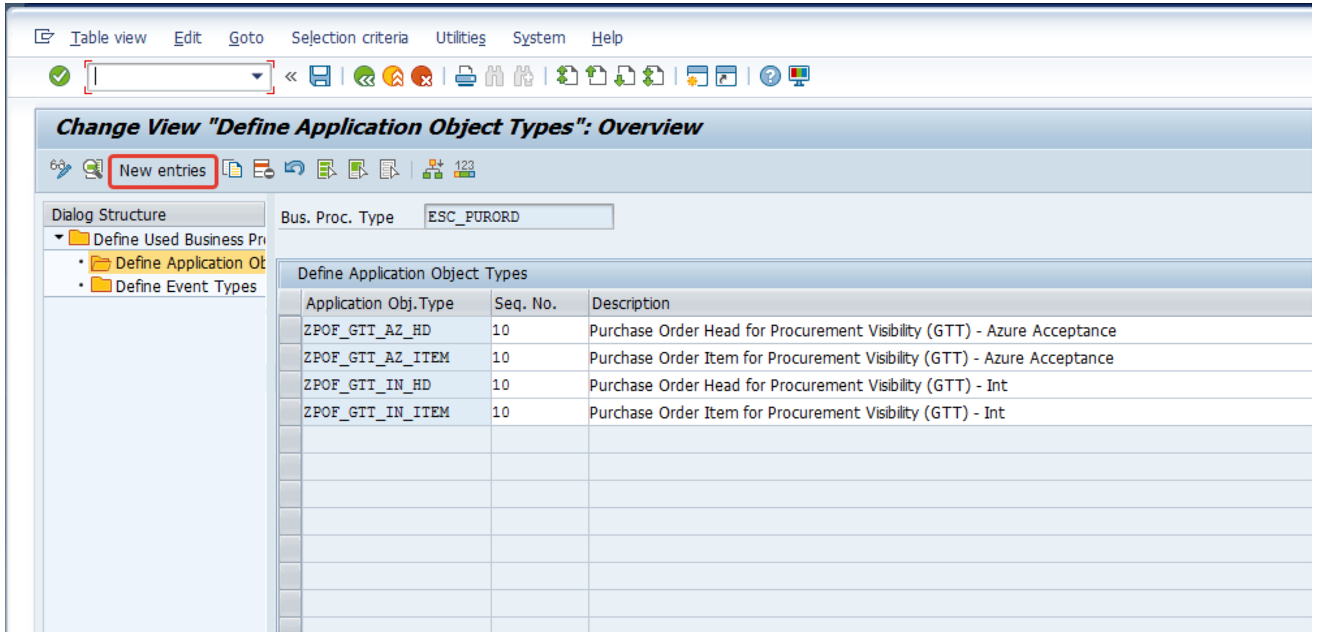
Dialog Structure
- Define Used Business Pr
  - Define Application Ol
  - Define Event Types

Define Used Business Process Types

| Bus. Proc. Type | Update Mode | BPT Process Mode | Description |
|---|---|---|---|
| EPL_NOTIF | Update Task… | Active | Notification in SAP R/3 Enterprise |
| ESC_DELIV | Update Task… | Active | Delivery in SAP R/3 Enterprise |
| ESC_FI_CLEARING | Update Task… | Active | FI Clearing in SAP R/3 Enterprise |
| ESC_MATDOC | Update Task… | Active | Material Document in SAP R/3 Enterprise |
| ESC_MM_INVOICE | Update Task… | Active | MM Invoice in SAP R/3 Enterprise |
| ESC_PURORD | Update Task… | Active | Purchase Order in SAP R/3 Enterprise |
| ESC_PURORD_FASHION | Update Task… | Active | Purchase Order (Seasonal Procurement) in SAP R/3 Enterprise 2.0 |
| ESC_SHIPMT | Update Task… | Active | Shipment (SAP R/3 Enterprise) |
| ESC_SORDER | Update Task… | Active | Sales Order in SAP R/3 Enterprise |
| ESC_WRKORD | Update Task… | Active | Workorder (Production, Service, Maintenance) in SAP R/3 Enterprise |
| OCB10_ORDER | D Dialog Upd… | Active | Booking Order in Ocean Carrier Booking Process |
| SNC_MSGIN | D Dialog Upd… | Active | SNC Inbound messages |
| SNC_PURORD | D Dialog Upd… | Active | SNC Purchase Order |
| SNC_RPLORD | D Dialog Upd… | Active | SNC Replenishment Order |
| TMS_INS | Update Task… | Active | Instructions (SAP TM) |
| TMS_RES | Update Task… | Active | Resources (SAP TM) |
| TMS_TOR | Update Task… | Active | Transportation Order (SAP TM) |

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-13: Click **New Entries** to create a new Application Object Type



Change View "Define Application Object Types": Overview

Bus. Proc. Type ESC_PURORD

Define Application Object Types

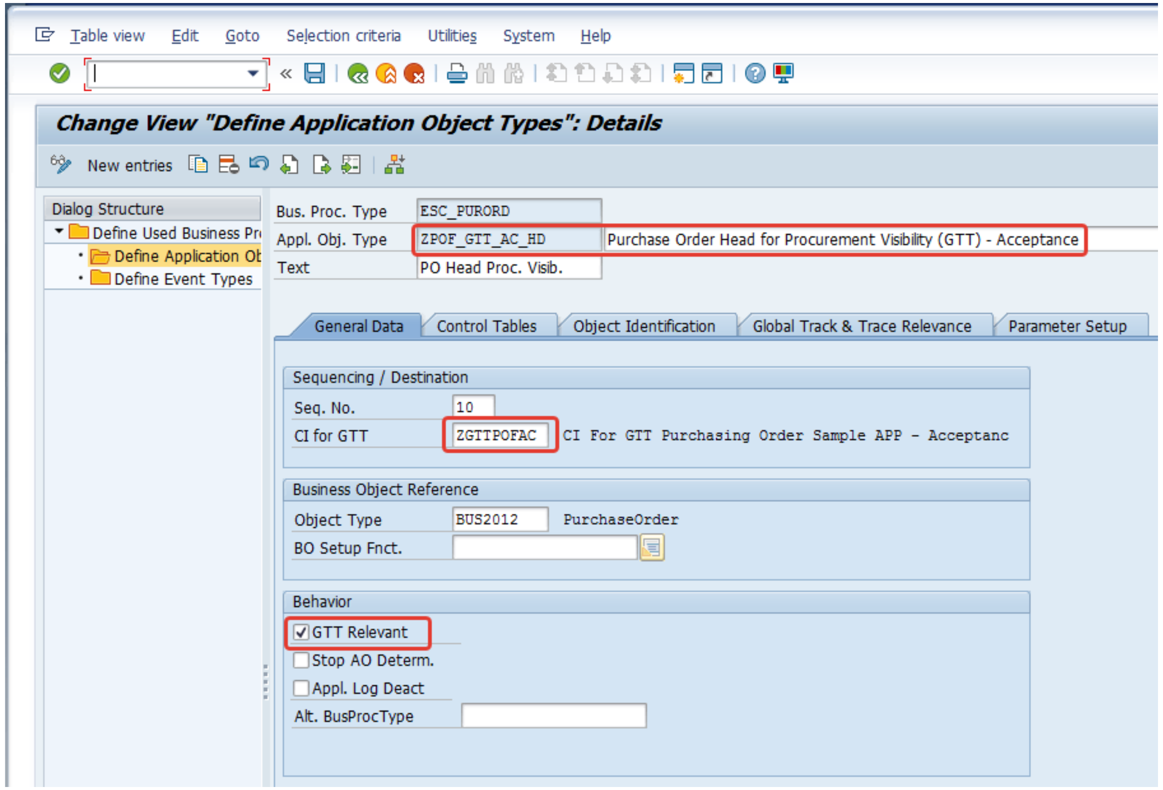
| Application Obj.Type | Seq. No. | Description |
|---|---|---|
| ZPOF_GTT_AZ_HD | 10 | Purchase Order Head for Procurement Visibility (GTT) - Azure Acceptance |
| ZPOF_GTT_AZ_ITEM | 10 | Purchase Order Item for Procurement Visibility (GTT) - Azure Acceptance |
| ZPOF_GTT_IN_HD | 10 | Purchase Order Head for Procurement Visibility (GTT) - Int |
| ZPOF_GTT_IN_ITEM | 10 | Purchase Order Item for Procurement Visibility (GTT) - Int |

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-14: Fill in the **Application Object Type** and **Text** fields

7-15: Fill in the information required in the **General Data** tab. **CI for GTT** is the CI Tenant you created in STEP 5.

7-16: Check **GTT Relevant**



Change View "Define Application Object Types": Details

Bus. Proc. Type: ESC_PURORD
Appl. Obj. Type: ZPOF_GTT_AC_HD Purchase Order Head for Procurement Visibility (GTT) - Acceptance
Text: PO Head Proc. Visib.

Sequencing / Destination
Seq. No.: 10
CI for GTT: ZGTTPOFAC CI For GTT Purchasing Order Sample APP - Acceptanc

Business Object Reference
Object Type: BUS2012 PurchaseOrder
BO Setup Fnct.:

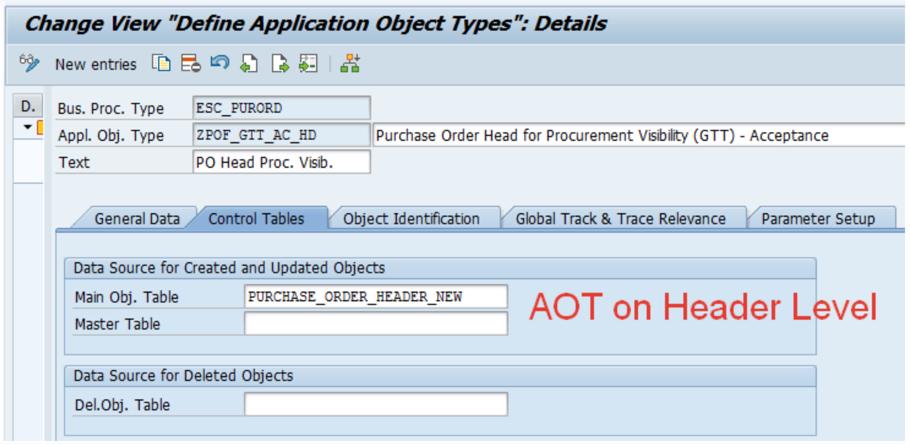
Behavior
☑ GTT Relevant
☐ Stop AO Determ.
☐ Appl. Log Deact
Alt. BusProcType:

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-17: Fill in the **Main Object table** and **Master Table**

**Caution:**

If the event type or application object type is on header level, then you only need to assign the **Main Object Table**. Otherwise, if the event type or application object type is on item level, then you need to assign the **Main Object Table** and **Master Table**, and assign the reference between the **Main Object Table** and **Master Table**.



*Change View "Define Application Object Types": Details*

| | | |
|---|---|---|
| Bus. Proc. Type | ESC_PURORD | |
| Appl. Obj. Type | ZPOF_GTT_AC_HD | Purchase Order Head for Procurement Visibility (GTT) - Acceptance |
| Text | PO Head Proc. Visib. | |

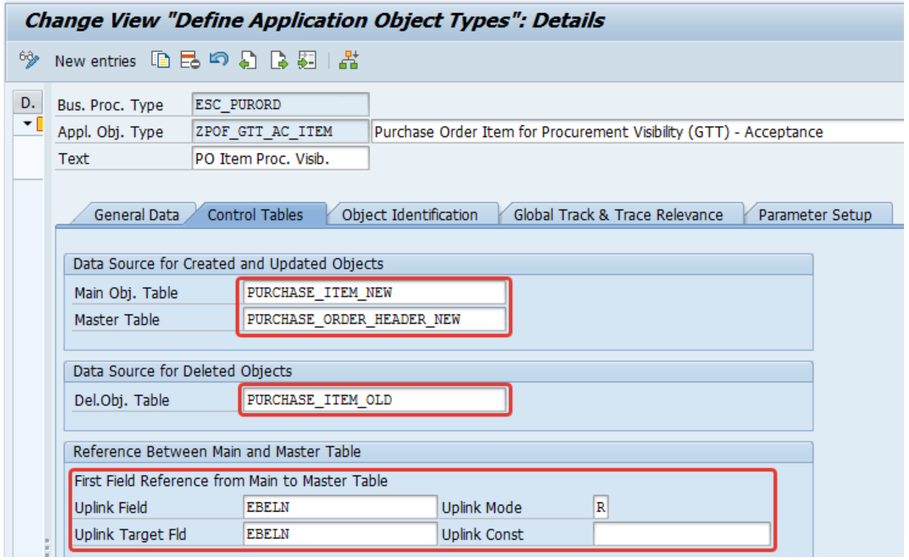
Data Source for Created and Updated Objects

| | |
|---|---|
| Main Obj. Table | PURCHASE_ORDER_HEADER_NEW |
| Master Table | |

Data Source for Deleted Objects

| | |
|---|---|
| Del.Obj. Table | |

AOT on Header Level



*Change View "Define Application Object Types": Details*

| | | |
|---|---|---|
| Bus. Proc. Type | ESC_PURORD | |
| Appl. Obj. Type | ZPOF_GTT_AC_ITEM | Purchase Order Item for Procurement Visibility (GTT) - Acceptance |
| Text | PO Item Proc. Visib. | |

Data Source for Created and Updated Objects

| | |
|---|---|
| Main Obj. Table | PURCHASE_ITEM_NEW |
| Master Table | PURCHASE_ORDER_HEADER_NEW |

Data Source for Deleted Objects

| | |
|---|---|
| Del.Obj. Table | PURCHASE_ITEM_OLD |

Reference Between Main and Master Table

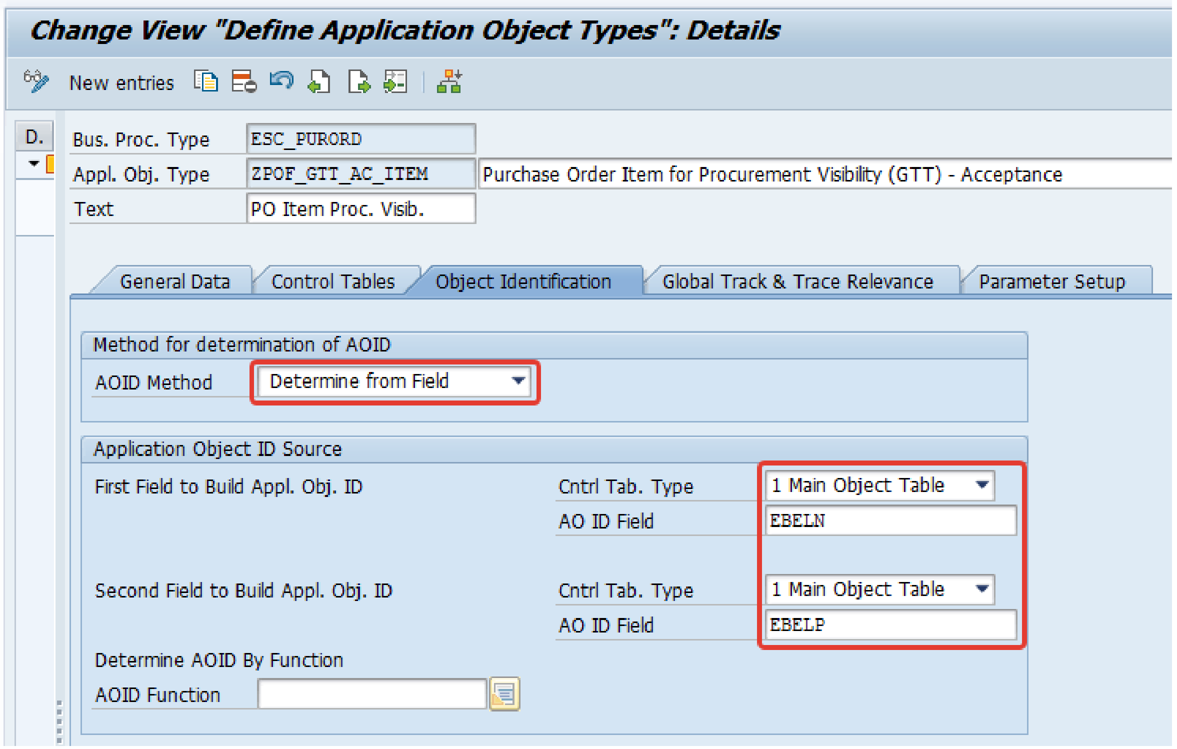
First Field Reference from Main to Master Table

| | | | |
|---|---|---|---|
| Uplink Field | EBELN | Uplink Mode | R |
| Uplink Target Fld | EBELN | Uplink Const | |

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-18: If there is no customized logic to determine the AOT ID, choose **Determine from Field**, use the key field to fill the AO ID fields

7-19: When choosing **Determine by Function**, you must enter the customized information in the AOID function field.



Change View "Define Application Object Types": Details

Bus. Proc. Type: ESC_PURORD

Appl. Obj. Type: ZPOF_GTT_AC_ITEM — Purchase Order Item for Procurement Visibility (GTT) - Acceptance

Text: PO Item Proc. Visib.

Method for determination of AOID

AOID Method: Determine from Field

Application Object ID Source

First Field to Build Appl. Obj. ID — Cntrl Tab. Type: 1 Main Object Table; AO ID Field: EBELN

Second Field to Build Appl. Obj. ID — Cntrl Tab. Type: 1 Main Object Table; AO ID Field: EBELP

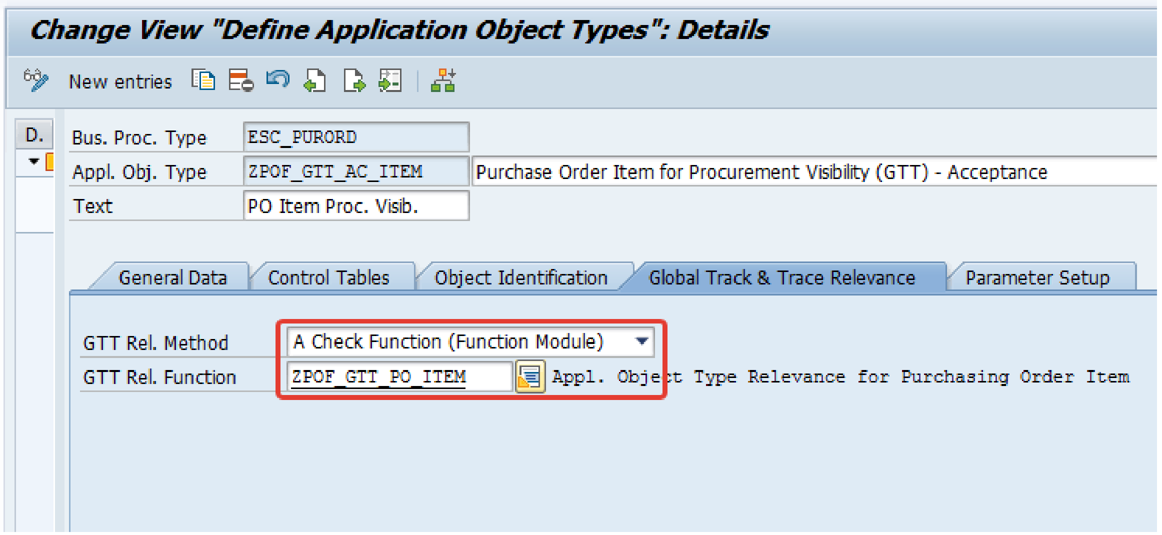
Determine AOID By Function

AOID Function:

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

7-20: In the **Global Track & Trace Relevance** tab, choose the **GTT Relevance Method** you need.

If you choose the **GTT Relevance Method** *Check Function*, then you need to define a relevance function according to STEP 6, and fill in the relevance function name here.



Change View "Define Application Object Types": Details

Bus. Proc. Type: ESC_PURORD
Appl. Obj. Type: ZPOF_GTT_AC_ITEM Purchase Order Item for Procurement Visibility (GTT) - Acceptance
Text: PO Item Proc. Visib.

GTT Rel. Method: A Check Function (Function Module)
GTT Rel. Function: ZPOF_GTT_PO_ITEM Appl. Object Type Relevance for Purchasing Order Item

# STEP 7: Define Used Business Process Types, Appl. Object Types and Event Types

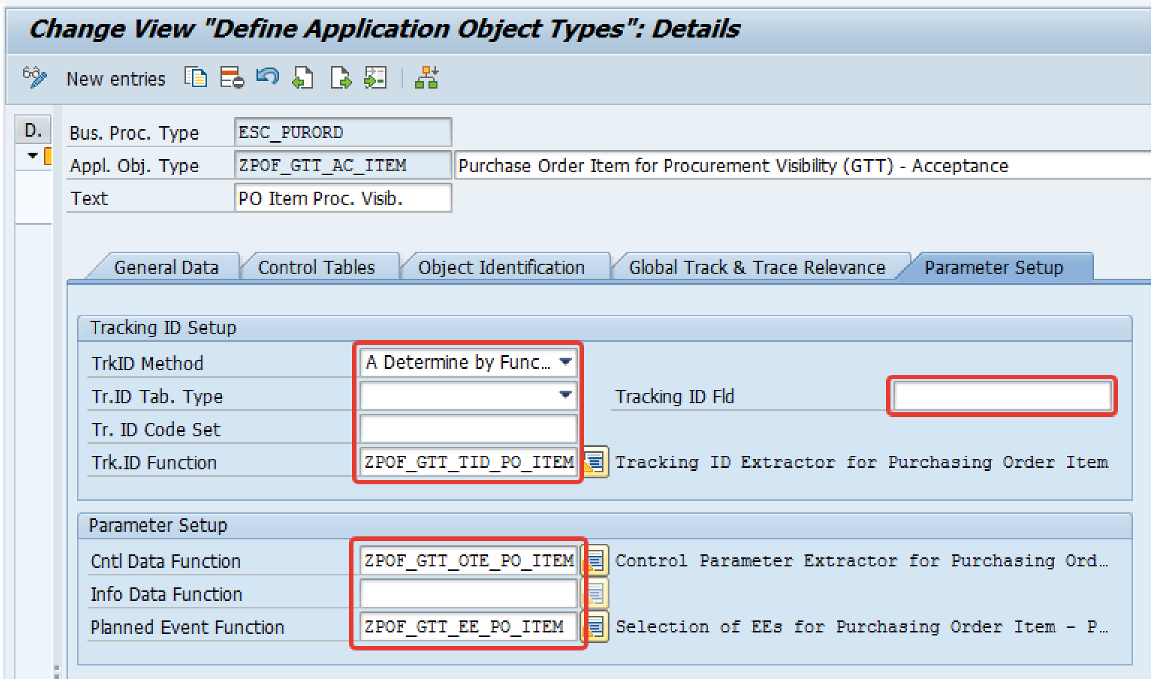
7-21: In the **Parameter Setup** tab, choose the **TrkID Method** as you need.

If you choose the **TrkID Method** as *Determine by Function*, then you need to define a tracking ID function according to STEP 6, and fill in the relevance function name here.

If no customized logic exists, for **TrkID Method** choose *Determine from Field,* then you need to fill the key field and name the Code Set for the AOT.

Fill in the extractor functions for **Control Data, Info Data(optional), Planned Event.**

Click **Save**.



***Change View "Define Application Object Types": Details***

|  |  |  |
|---|---|---|
| Bus. Proc. Type | ESC_PURORD |  |
| Appl. Obj. Type | ZPOF_GTT_AC_ITEM | Purchase Order Item for Procurement Visibility (GTT) - Acceptance |
| Text | PO Item Proc. Visib. |  |

Tracking ID Setup

|  |  |  |  |
|---|---|---|---|
| TrkID Method | A Determine by Func... |  |  |
| Tr.ID Tab. Type |  | Tracking ID Fld |  |
| Tr. ID Code Set |  |  |  |
| Trk.ID Function | ZPOF_GTT_TID_PO_ITEM | Tracking ID Extractor for Purchasing Order Item |  |

Parameter Setup

|  |  |  |
|---|---|---|
| Cntl Data Function | ZPOF_GTT_OTE_PO_ITEM | Control Parameter Extractor for Purchasing Ord... |
| Info Data Function |  |  |
| Planned Event Function | ZPOF_GTT_EE_PO_ITEM | Selection of EEs for Purchasing Order Item - P... |

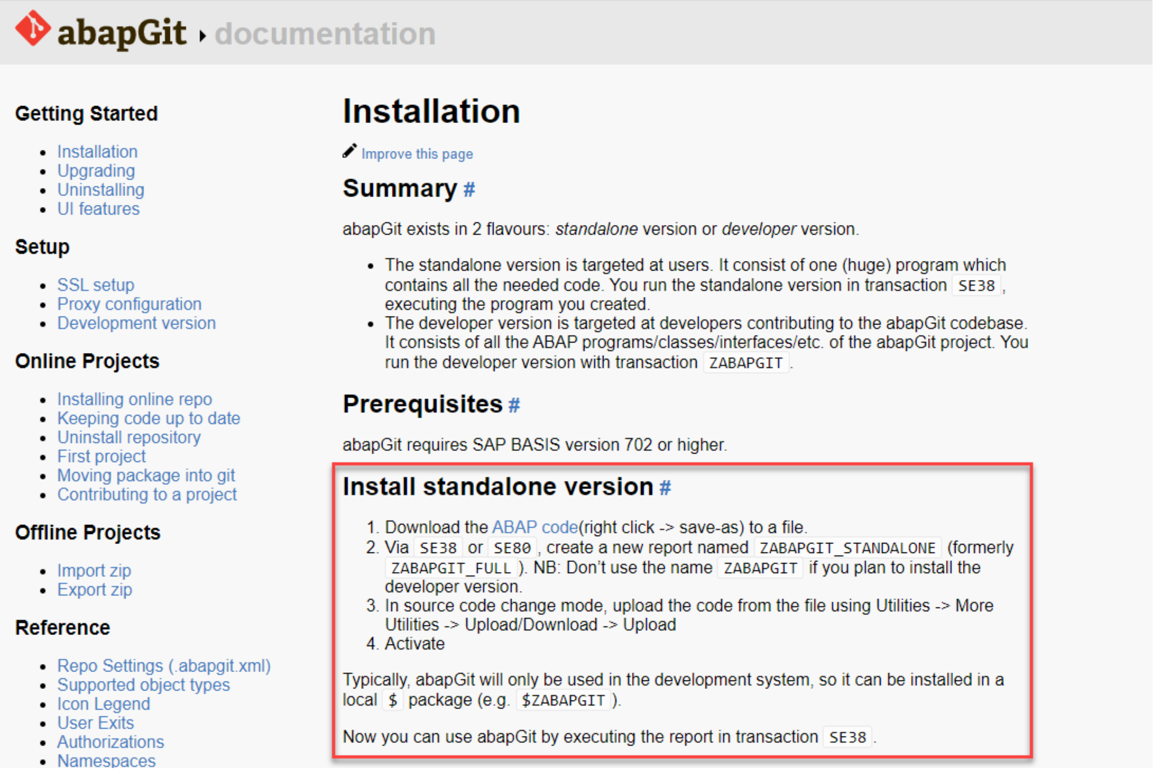
# C) Download ABAP Code from GitHub

# STEP 1: Install abapGit

You need to install abapGit before downloading codes from GitHub.

To install abapGit, follow the instructions on https://docs.abapgit.org/guide-install.html. Make sure you **Install the standalone version** in your dev system.

When installation is complete, a new report is created, *ZABAPGIT_STANDALONE*.



abapGit › documentation

**Getting Started**

- Installation
- Upgrading
- Uninstalling
- UI features

**Setup**

- SSL setup
- Proxy configuration
- Development version

**Online Projects**

- Installing online repo
- Keeping code up to date
- Uninstall repository
- First project
- Moving package into git
- Contributing to a project

**Offline Projects**

- Import zip
- Export zip

**Reference**

- Repo Settings (.abapgit.xml)
- Supported object types
- Icon Legend
- User Exits
- Authorizations
- Namespaces

## Installation

Improve this page

### Summary #

abapGit exists in 2 flavours: *standalone* version or *developer* version.

- The standalone version is targeted at users. It consist of one (huge) program which contains all the needed code. You run the standalone version in transaction `SE38`, executing the program you created.
- The developer version is targeted at developers contributing to the abapGit codebase. It consists of all the ABAP programs/classes/interfaces/etc. of the abapGit project. You run the developer version with transaction `ZABAPGIT`.

### Prerequisites #

abapGit requires SAP BASIS version 702 or higher.

### Install standalone version #

1. Download the ABAP code(right click -> save-as) to a file.
2. Via `SE38` or `SE80`, create a new report named `ZABAPGIT_STANDALONE` (formerly `ZABAPGIT_FULL`). NB: Don't use the name `ZABAPGIT` if you plan to install the developer version.
3. In source code change mode, upload the code from the file using Utilities -> More Utilities -> Upload/Download -> Upload
4. Activate

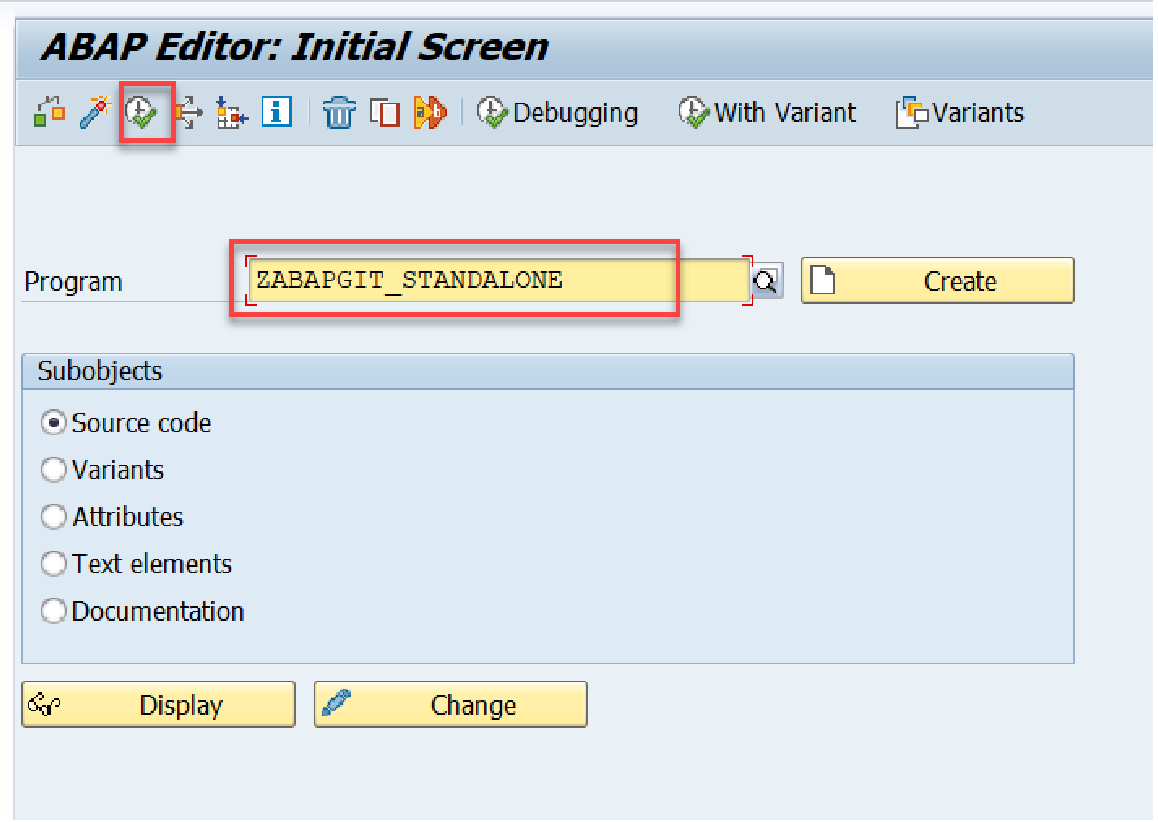
Typically, abapGit will only be used in the development system, so it can be installed in a local `$` package (e.g. `$ZABAPGIT`).

Now you can use abapGit by executing the report in transaction `SE38`.

# STEP 2: Download ABAP Code

2-1: Enter T-code *SE38* and fill in the report name from STEP 1, *ZABAPGIT_STANDALONE*

2-2: Click **Execute** to run the report



ABAP Editor: Initial Screen

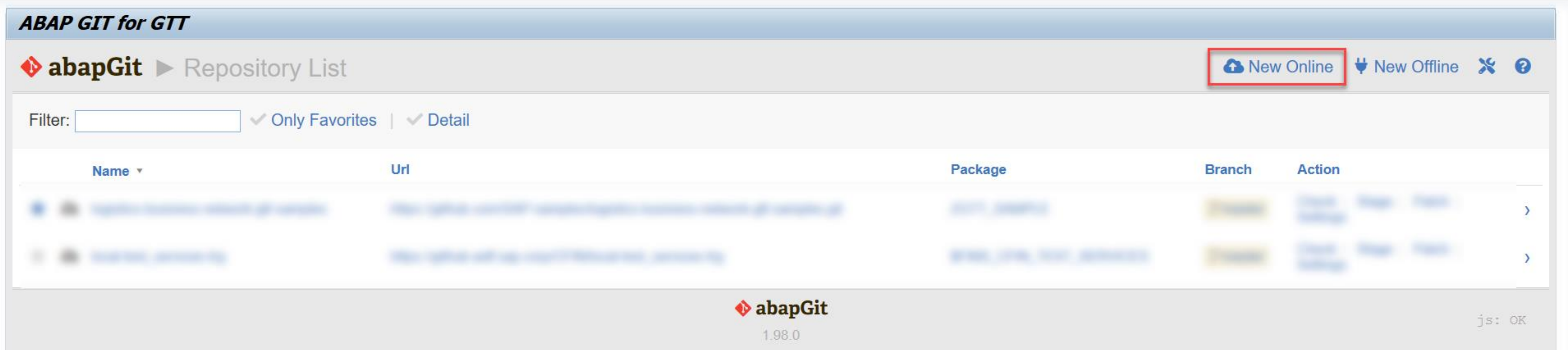
Program: ZABAPGIT_STANDALONE

Subobjects

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

# STEP 2: Download ABAP Code

2-3: Click **New Online** to download the code



| Name | Url | Package | Branch | Action |
|---|---|---|---|---|
| [redacted] | [redacted] | [redacted] | [redacted] | [redacted] |
| [redacted] | [redacted] | [redacted] | [redacted] | [redacted] |

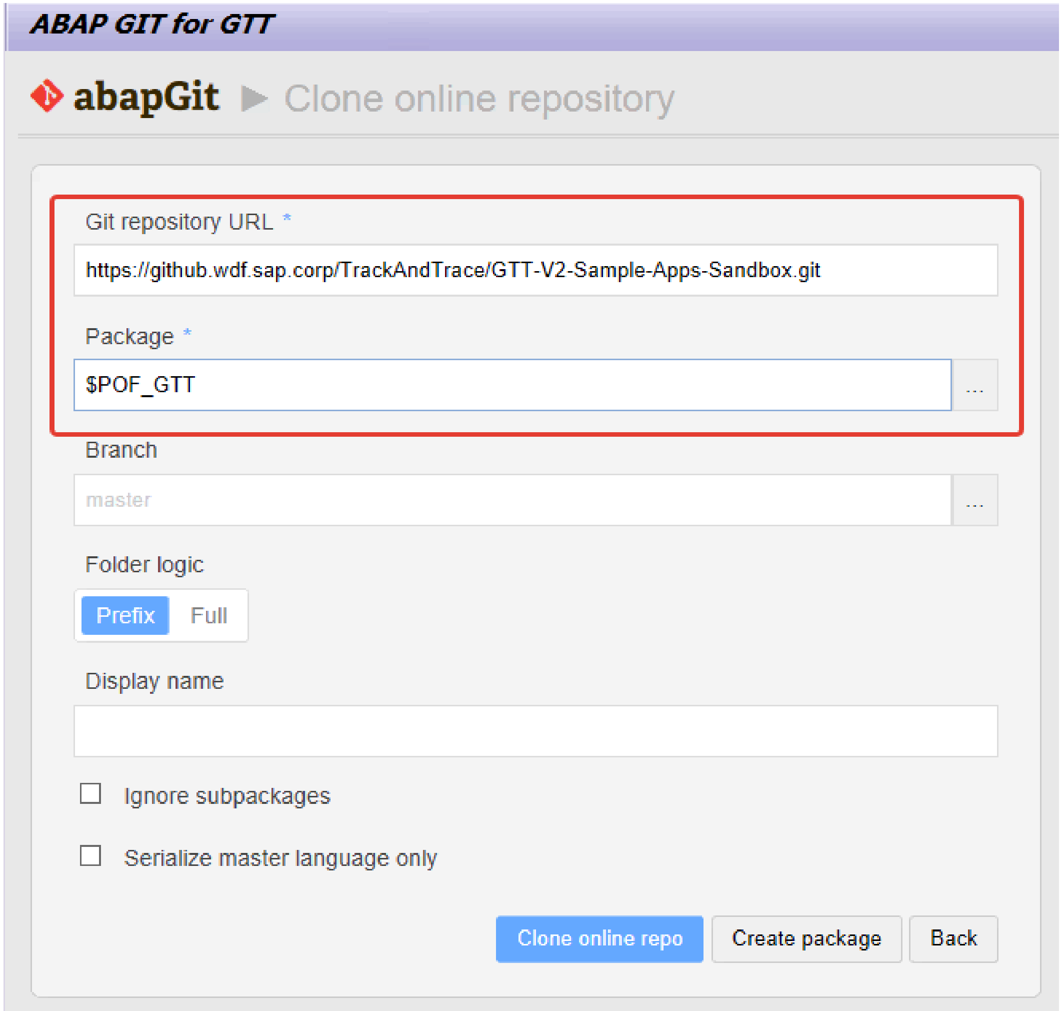
# STEP 2: Download ABAP Code

2-4: Fill in the **Git repository URL**:

https://github.com/SAP-samples/logistics-business-network-gtt-samples.git

2-5: Fill in the **Package** where you want to create the new ABAP code. If the package does not exist yet, click **Create package** to create it.

2-6: Click **Clone online repo** to download the code



ABAP GIT for GTT

abapGit ▶ Clone online repository

Git repository URL *

https://github.wdf.sap.corp/TrackAndTrace/GTT-V2-Sample-Apps-Sandbox.git

Package *

$POF_GTT

Branch

master

Folder logic

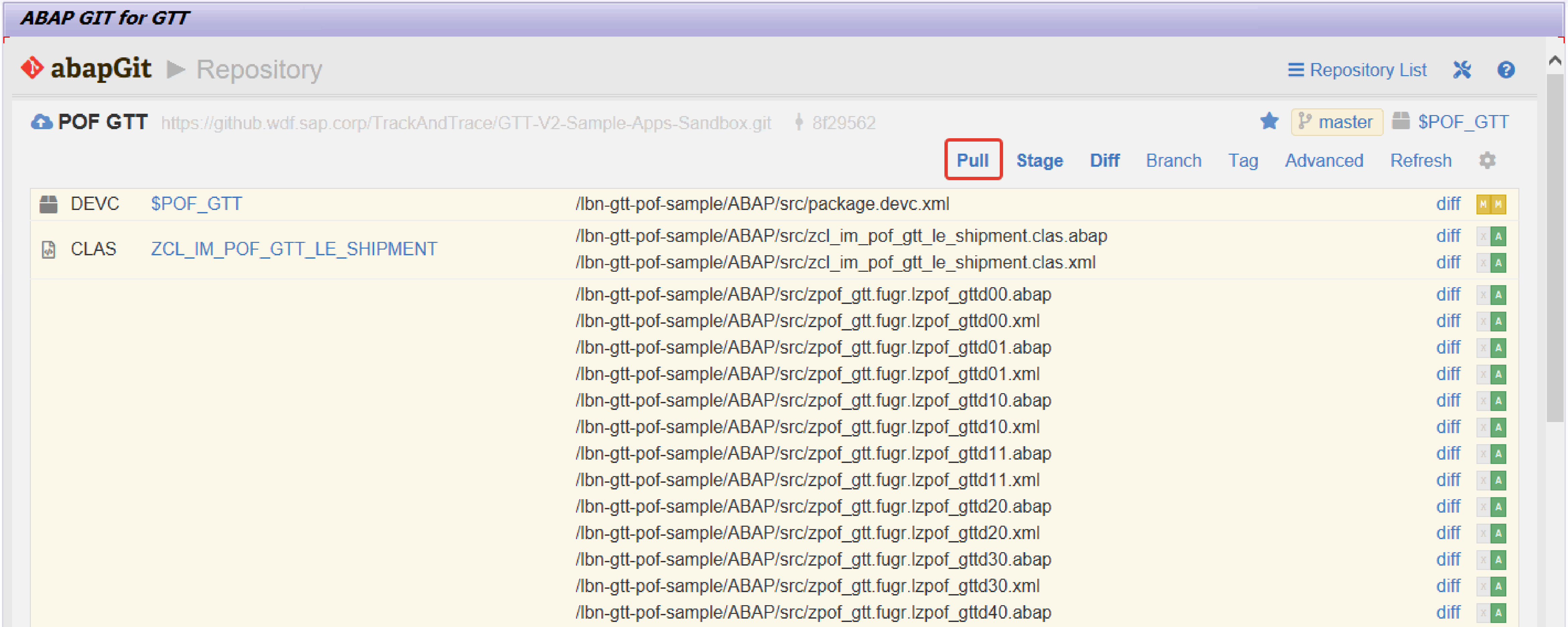
Display name

□ Ignore subpackages

□ Serialize master language only

# STEP 2: Download ABAP Code

2-7: Click **Pull** to pull down the latest version code



ABAP GIT for GTT

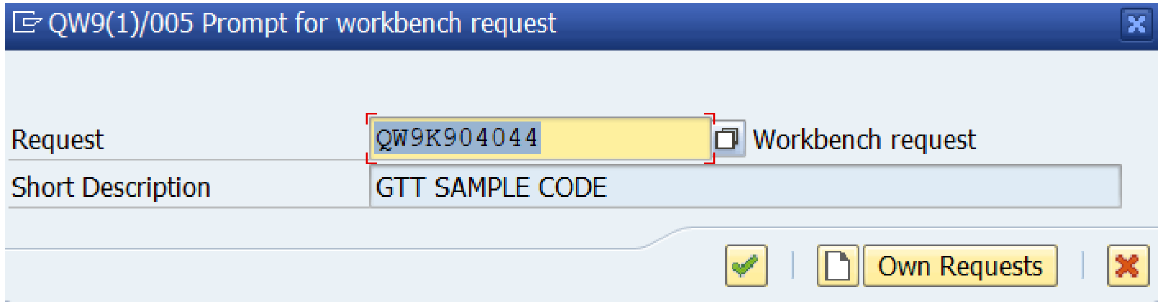
abapGit ▶ Repository

POF GTT https://github.wdf.sap.corp/TrackAndTrace/GTT-V2-Sample-Apps-Sandbox.git 8f29562

| | | | | |
|---|---|---|---|---|
| DEVC | $POF_GTT | /lbn-gtt-pof-sample/ABAP/src/package.devc.xml | diff | M M |
| CLAS | ZCL_IM_POF_GTT_LE_SHIPMENT | /lbn-gtt-pof-sample/ABAP/src/zcl_im_pof_gtt_le_shipment.clas.abap | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zcl_im_pof_gtt_le_shipment.clas.xml | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd00.abap | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd00.xml | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd01.abap | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd01.xml | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd10.abap | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd10.xml | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd11.abap | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd11.xml | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd20.abap | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd20.xml | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd30.abap | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd30.xml | diff | A |
| | | /lbn-gtt-pof-sample/ABAP/src/zpof_gtt.fugr.lzpof_gttd40.abap | diff | A |

# STEP 2: Download ABAP Code

2-8: Assign the change to a change request. If you do not have any available change request, you need to create a new one.

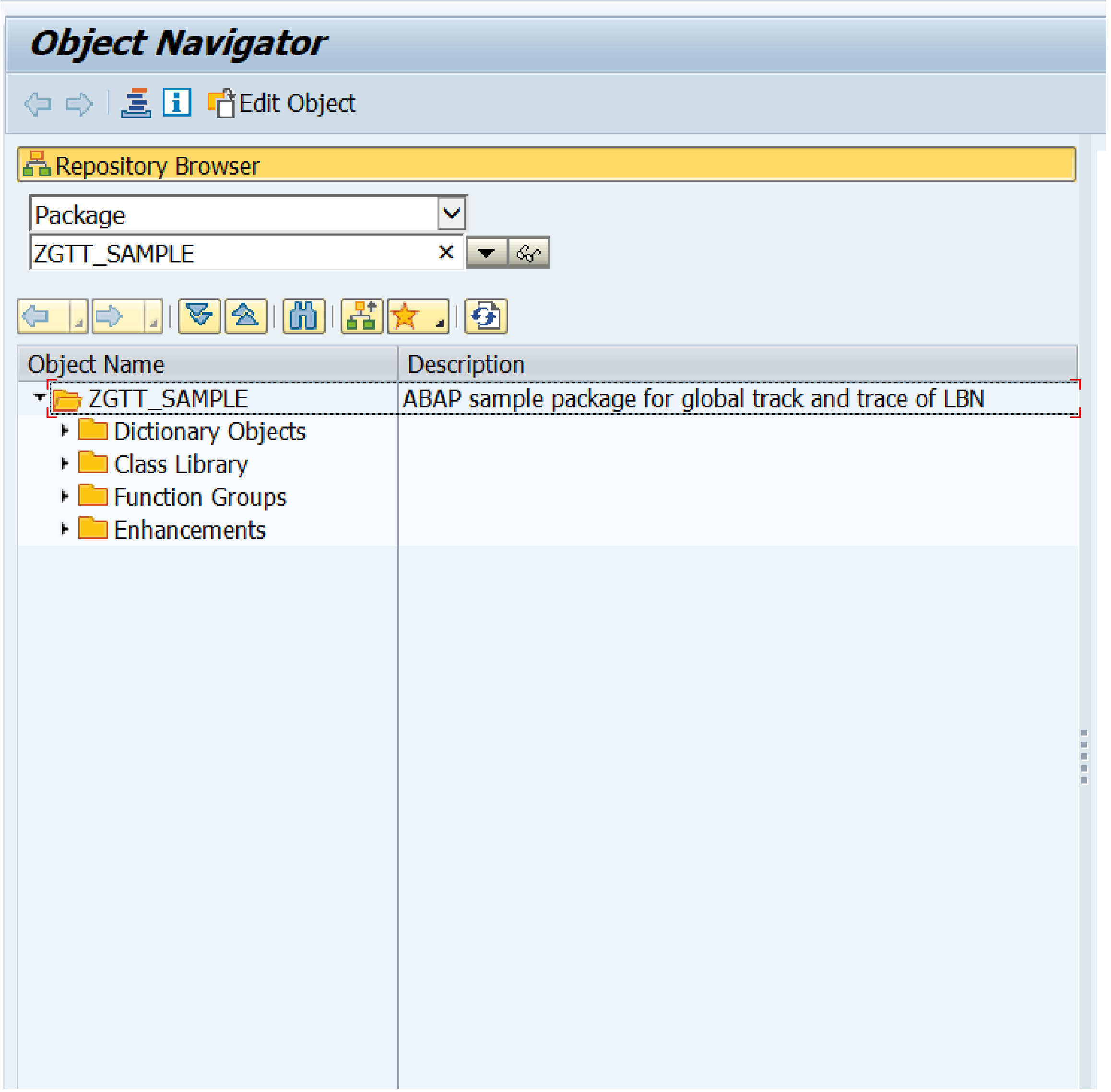


QW9(1)/005 Prompt for workbench request

|  |  |  |
|---|---|---|
| Request | QW9K904044 | Workbench request |
| Short Description | GTT SAMPLE CODE |  |

# STEP 2: Download ABAP Code

2-9: After you download the code, you can check them with T-code *SE80*.



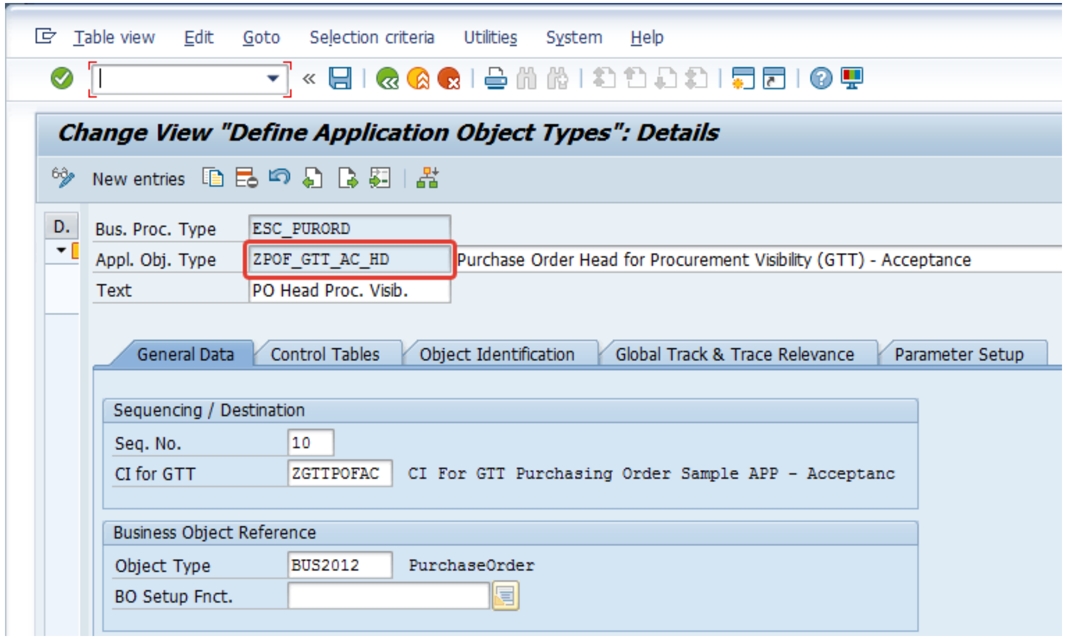
Package: ZGTT_SAMPLE

| Object Name | Description |
|---|---|
| ZGTT_SAMPLE | ABAP sample package for global track and trace of LBN |
| Dictionary Objects | |
| Class Library | |
| Function Groups | |
| Enhancements | |

# D) Configuration and Coding Guide - Advanced

# 1: Maintain AOT type

When you are creating Application Object Type for one Business Process Type, please make sure the AOT name must be the same as the name which is defined in the corresponding model in Manage Models application in GTT V2.



**Change View "Define Application Object Types": Details**

| | |
|---|---|
| Bus. Proc. Type | ESC_PURORD |
| Appl. Obj. Type | ZPOF_GTT_AC_HD Purchase Order Head for Procurement Visibility (GTT) - Acceptance |
| Text | PO Head Proc. Visib. |

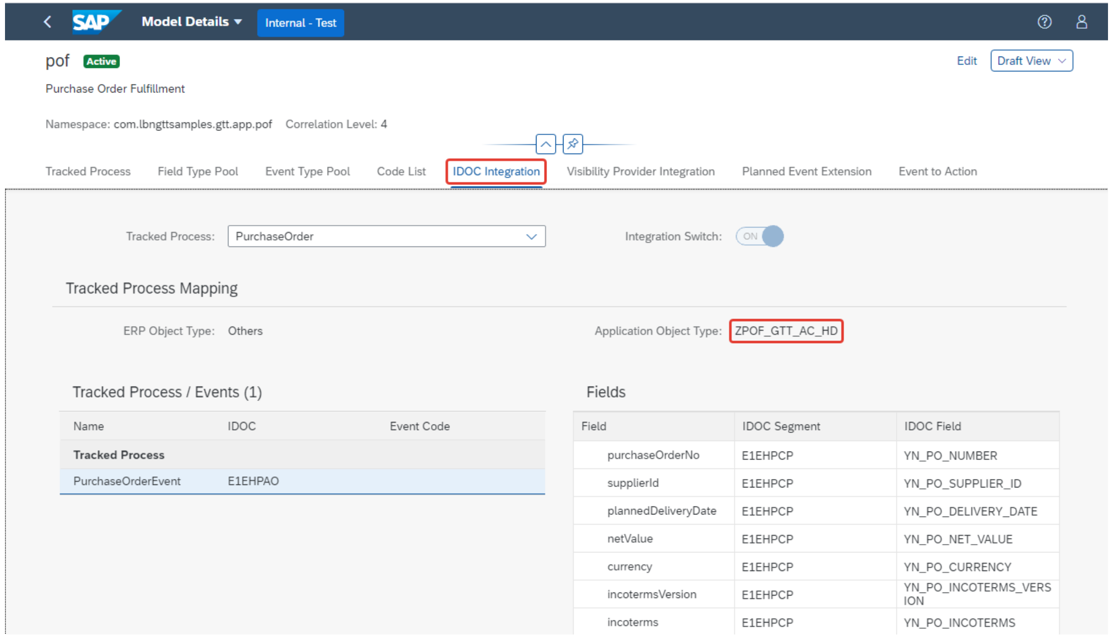
General Data | Control Tables | Object Identification | Global Track & Trace Relevance | Parameter Setup

Sequencing / Destination

| | |
|---|---|
| Seq. No. | 10 |
| CI for GTT | ZGTTPOFAC CI For GTT Purchasing Order Sample APP - Acceptanc |

Business Object Reference

| | |
|---|---|
| Object Type | BUS2012 PurchaseOrder |
| BO Setup Fnct. | |



pof Active

Purchase Order Fulfillment

Namespace: com.lbngttsamples.gtt.app.pof Correlation Level: 4

Tracked Process: PurchaseOrder

Integration Switch: ON

Tracked Process Mapping

ERP Object Type: Others

Application Object Type: ZPOF_GTT_AC_HD

Tracked Process / Events (1)

| Name | IDOC | Event Code |
|---|---|---|
| **Tracked Process** | | |
| PurchaseOrderEvent | E1EHPAO | |

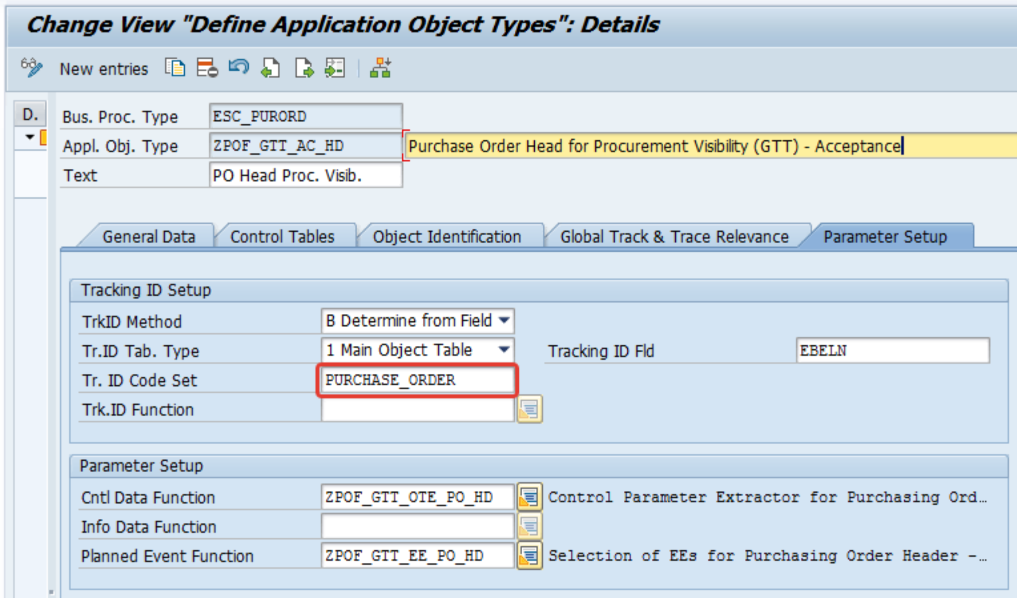
Fields

| Field | IDOC Segment | IDOC Field |
|---|---|---|
| purchaseOrderNo | E1EHPCP | YN_PO_NUMBER |
| supplierId | E1EHPCP | YN_PO_SUPPLIER_ID |
| plannedDeliveryDate | E1EHPCP | YN_PO_DELIVERY_DATE |
| netValue | E1EHPCP | YN_PO_NET_VALUE |
| currency | E1EHPCP | YN_PO_CURRENCY |
| incotermsVersion | E1EHPCP | YN_PO_INCOTERMS_VERSION |
| incoterms | E1EHPCP | YN_PO_INCOTERMS |

# 2: Maintain Tracking ID Type

In the AOT you maintained, please make sure the Tracking ID Type is the same as the name which is defined in the corresponding process type of the model in Manage Models application in GTT V2.

If the Tracking ID Type is determined by Field, then input the value source field in the Tracking ID field, and the Code Set which is referring to the Tracking ID Type for the AOT like below.



**Change View "Define Application Object Types": Details**

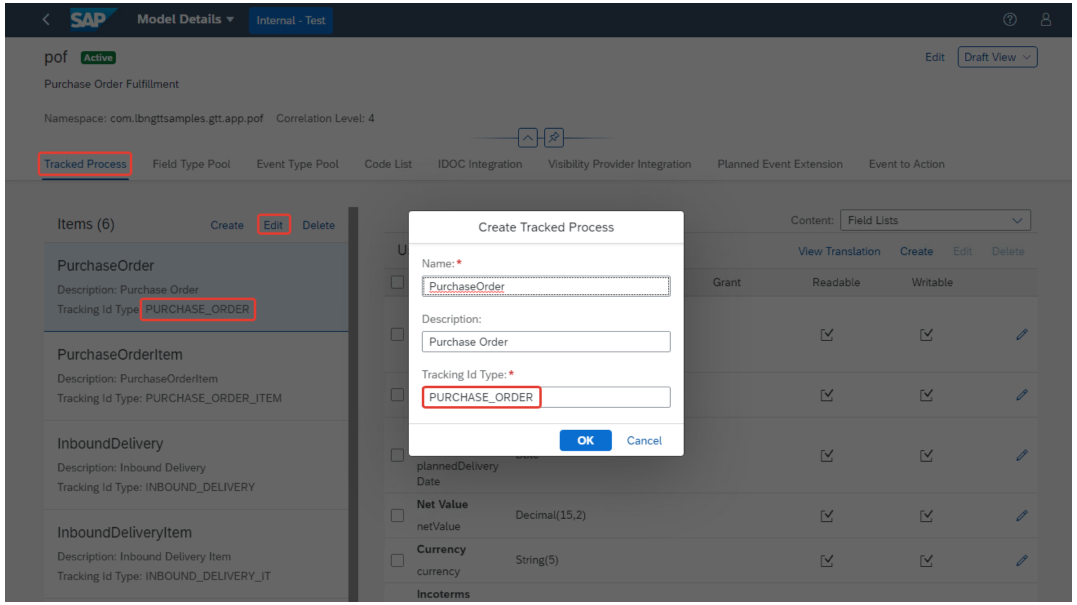
| | |
|---|---|
| Bus. Proc. Type | ESC_PURORD |
| Appl. Obj. Type | ZPOF_GTT_AC_HD Purchase Order Head for Procurement Visibility (GTT) - Acceptance |
| Text | PO Head Proc. Visib. |

Tracking ID Setup

| | | | |
|---|---|---|---|
| TrkID Method | B Determine from Field | | |
| Tr.ID Tab. Type | 1 Main Object Table | Tracking ID Fld | EBELN |
| Tr. ID Code Set | PURCHASE_ORDER | | |
| Trk.ID Function | | | |

Parameter Setup

| | | |
|---|---|---|
| Cntl Data Function | ZPOF_GTT_OTE_PO_HD | Control Parameter Extractor for Purchasing Ord… |
| Info Data Function | | |
| Planned Event Function | ZPOF_GTT_EE_PO_HD | Selection of EEs for Purchasing Order Header -… |



pof Active

Purchase Order Fulfillment

Namespace: com.lbngttsamples.gtt.app.pof Correlation Level: 4

PurchaseOrder
Description: Purchase Order
Tracking Id Type: PURCHASE_ORDER

PurchaseOrderItem
Description: PurchaseOrderItem
Tracking Id Type: PURCHASE_ORDER_ITEM

InboundDelivery
Description: Inbound Delivery
Tracking Id Type: INBOUND_DELIVERY

InboundDeliveryItem
Description: Inbound Delivery Item
Tracking Id Type: INBOUND_DELIVERY_IT

Create Tracked Process

Name:* PurchaseOrder

Description: Purchase Order

Tracking Id Type:* PURCHASE_ORDER

| | Grant | Readable | Writable |
|---|---|---|---|
| plannedDelivery Date | | ✓ | ✓ |
| Net Value netValue | Decimal(15,2) | ✓ | ✓ |
| Currency currency | String(5) | ✓ | ✓ |
| Incoterms | | | |

# 3: Make the customization logic in the function modules and assign them to the extractor function.

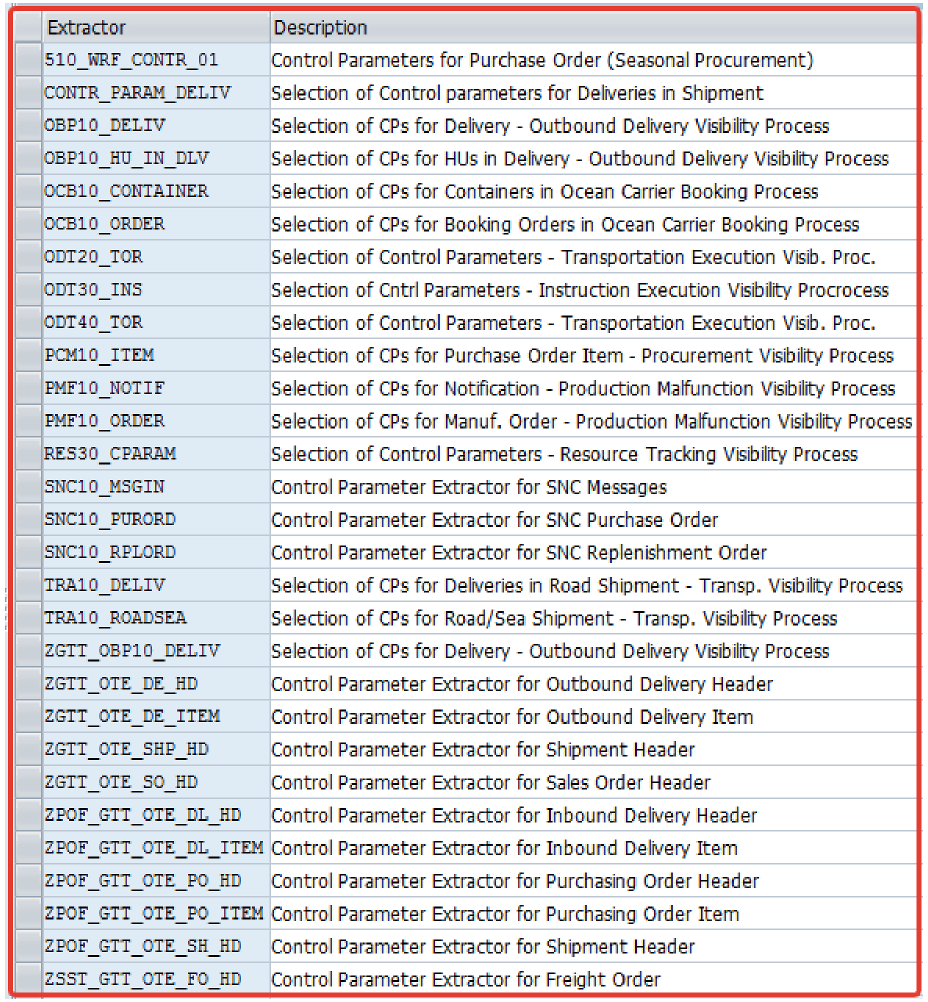
You can assign customization function models to the following extractor function:

1. GTT relevance function of AOT for tracked process tracking
2. GTT relevance function of Event Type for event tracking
3. Planned Event Extractors
4. Control Parameter Extractors
5. Info Parameter Extractors(optional)
6. Tracking ID Extractors
7. Event Data Extractors
8. AOT ID Extractors

Please select one category above, create the extractor function and assign the corresponding modules.

For customization of GTT relevance and AOT ID, you need to enable *Determine by Function* option.

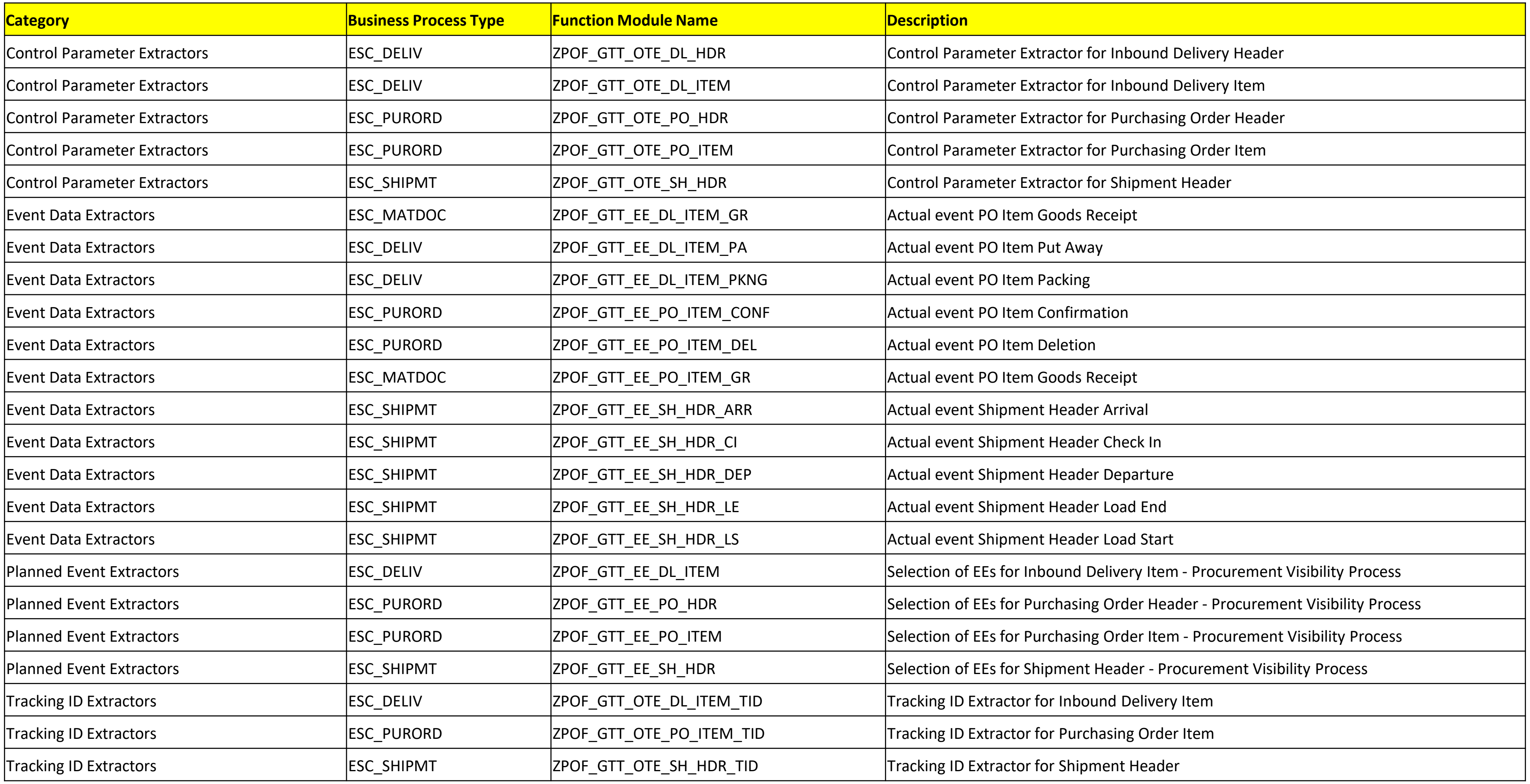
For customization of Tracking ID Type, you need to enable *Check Function(Function Module)* option.



| Extractor | Description |
|---|---|
| 510_WRF_CONTR_01 | Control Parameters for Purchase Order (Seasonal Procurement) |
| CONTR_PARAM_DELIV | Selection of Control parameters for Deliveries in Shipment |
| OBP10_DELIV | Selection of CPs for Delivery - Outbound Delivery Visibility Process |
| OBP10_HU_IN_DLV | Selection of CPs for HUs in Delivery - Outbound Delivery Visibility Process |
| OCB10_CONTAINER | Selection of CPs for Containers in Ocean Carrier Booking Process |
| OCB10_ORDER | Selection of CPs for Booking Orders in Ocean Carrier Booking Process |
| ODT20_TOR | Selection of Control Parameters - Transportation Execution Visib. Proc. |
| ODT30_INS | Selection of Cntrl Parameters - Instruction Execution Visibility Procrocess |
| ODT40_TOR | Selection of Control Parameters - Transportation Execution Visib. Proc. |
| PCM10_ITEM | Selection of CPs for Purchase Order Item - Procurement Visibility Process |
| PMF10_NOTIF | Selection of CPs for Notification - Production Malfunction Visibility Process |
| PMF10_ORDER | Selection of CPs for Manuf. Order - Production Malfunction Visibility Process |
| RES30_CPARAM | Selection of Control Parameters - Resource Tracking Visibility Process |
| SNC10_MSGIN | Control Parameter Extractor for SNC Messages |
| SNC10_PURORD | Control Parameter Extractor for SNC Purchase Order |
| SNC10_RPLORD | Control Parameter Extractor for SNC Replenishment Order |
| TRA10_DELIV | Selection of CPs for Deliveries in Road Shipment - Transp. Visibility Process |
| TRA10_ROADSEA | Selection of CPs for Road/Sea Shipment - Transp. Visibility Process |
| ZGTT_OBP10_DELIV | Selection of CPs for Delivery - Outbound Delivery Visibility Process |
| ZGTT_OTE_DE_HD | Control Parameter Extractor for Outbound Delivery Header |
| ZGTT_OTE_DE_ITEM | Control Parameter Extractor for Outbound Delivery Item |
| ZGTT_OTE_SHP_HD | Control Parameter Extractor for Shipment Header |
| ZGTT_OTE_SO_HD | Control Parameter Extractor for Sales Order Header |
| ZPOF_GTT_OTE_DL_HD | Control Parameter Extractor for Inbound Delivery Header |
| ZPOF_GTT_OTE_DL_ITEM | Control Parameter Extractor for Inbound Delivery Item |
| ZPOF_GTT_OTE_PO_HD | Control Parameter Extractor for Purchasing Order Header |
| ZPOF_GTT_OTE_PO_ITEM | Control Parameter Extractor for Purchasing Order Item |
| ZPOF_GTT_OTE_SH_HD | Control Parameter Extractor for Shipment Header |
| ZSST_GTT_OTE_FO_HD | Control Parameter Extractor for Freight Order |

# 4: Sample Codes for Track Purchase Order Application

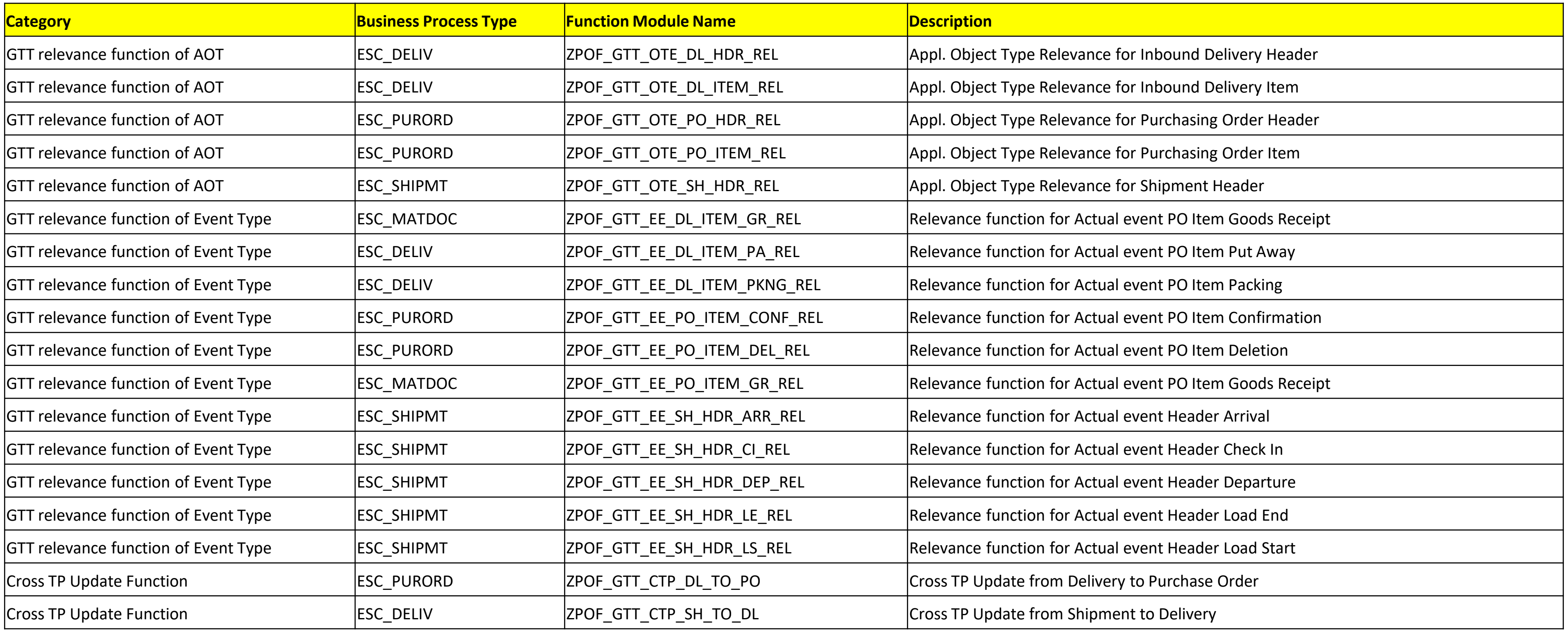
4-1 To support the Track Purchase Order Application, the sample codes covers the following cases by function group ZPOF_GTT:



| Category | Business Process Type | Function Module Name | Description |
|---|---|---|---|
| Control Parameter Extractors | ESC_DELIV | ZPOF_GTT_OTE_DL_HDR | Control Parameter Extractor for Inbound Delivery Header |
| Control Parameter Extractors | ESC_DELIV | ZPOF_GTT_OTE_DL_ITEM | Control Parameter Extractor for Inbound Delivery Item |
| Control Parameter Extractors | ESC_PURORD | ZPOF_GTT_OTE_PO_HDR | Control Parameter Extractor for Purchasing Order Header |
| Control Parameter Extractors | ESC_PURORD | ZPOF_GTT_OTE_PO_ITEM | Control Parameter Extractor for Purchasing Order Item |
| Control Parameter Extractors | ESC_SHIPMT | ZPOF_GTT_OTE_SH_HDR | Control Parameter Extractor for Shipment Header |
| Event Data Extractors | ESC_MATDOC | ZPOF_GTT_EE_DL_ITEM_GR | Actual event PO Item Goods Receipt |
| Event Data Extractors | ESC_DELIV | ZPOF_GTT_EE_DL_ITEM_PA | Actual event PO Item Put Away |
| Event Data Extractors | ESC_DELIV | ZPOF_GTT_EE_DL_ITEM_PKNG | Actual event PO Item Packing |
| Event Data Extractors | ESC_PURORD | ZPOF_GTT_EE_PO_ITEM_CONF | Actual event PO Item Confirmation |
| Event Data Extractors | ESC_PURORD | ZPOF_GTT_EE_PO_ITEM_DEL | Actual event PO Item Deletion |
| Event Data Extractors | ESC_MATDOC | ZPOF_GTT_EE_PO_ITEM_GR | Actual event PO Item Goods Receipt |
| Event Data Extractors | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_ARR | Actual event Shipment Header Arrival |
| Event Data Extractors | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_CI | Actual event Shipment Header Check In |
| Event Data Extractors | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_DEP | Actual event Shipment Header Departure |
| Event Data Extractors | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_LE | Actual event Shipment Header Load End |
| Event Data Extractors | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_LS | Actual event Shipment Header Load Start |
| Planned Event Extractors | ESC_DELIV | ZPOF_GTT_EE_DL_ITEM | Selection of EEs for Inbound Delivery Item - Procurement Visibility Process |
| Planned Event Extractors | ESC_PURORD | ZPOF_GTT_EE_PO_HDR | Selection of EEs for Purchasing Order Header - Procurement Visibility Process |
| Planned Event Extractors | ESC_PURORD | ZPOF_GTT_EE_PO_ITEM | Selection of EEs for Purchasing Order Item - Procurement Visibility Process |
| Planned Event Extractors | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR | Selection of EEs for Shipment Header - Procurement Visibility Process |
| Tracking ID Extractors | ESC_DELIV | ZPOF_GTT_OTE_DL_ITEM_TID | Tracking ID Extractor for Inbound Delivery Item |
| Tracking ID Extractors | ESC_PURORD | ZPOF_GTT_OTE_PO_ITEM_TID | Tracking ID Extractor for Purchasing Order Item |
| Tracking ID Extractors | ESC_SHIPMT | ZPOF_GTT_OTE_SH_HDR_TID | Tracking ID Extractor for Shipment Header |

# 4: Sample Codes for Track Purchase Order Application

4-2 To support the Track Purchase Order Application, the sample codes covers the following cases by function group ZPOF_GTT:



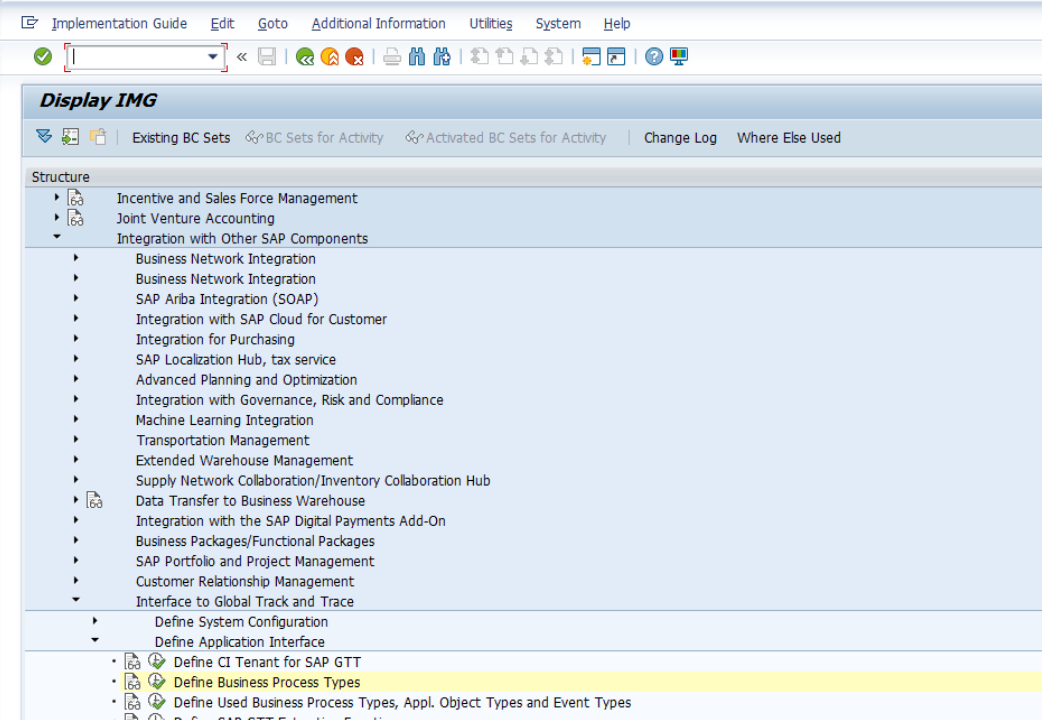
| Category | Business Process Type | Function Module Name | Description |
|---|---|---|---|
| GTT relevance function of AOT | ESC_DELIV | ZPOF_GTT_OTE_DL_HDR_REL | Appl. Object Type Relevance for Inbound Delivery Header |
| GTT relevance function of AOT | ESC_DELIV | ZPOF_GTT_OTE_DL_ITEM_REL | Appl. Object Type Relevance for Inbound Delivery Item |
| GTT relevance function of AOT | ESC_PURORD | ZPOF_GTT_OTE_PO_HDR_REL | Appl. Object Type Relevance for Purchasing Order Header |
| GTT relevance function of AOT | ESC_PURORD | ZPOF_GTT_OTE_PO_ITEM_REL | Appl. Object Type Relevance for Purchasing Order Item |
| GTT relevance function of AOT | ESC_SHIPMT | ZPOF_GTT_OTE_SH_HDR_REL | Appl. Object Type Relevance for Shipment Header |
| GTT relevance function of Event Type | ESC_MATDOC | ZPOF_GTT_EE_DL_ITEM_GR_REL | Relevance function for Actual event PO Item Goods Receipt |
| GTT relevance function of Event Type | ESC_DELIV | ZPOF_GTT_EE_DL_ITEM_PA_REL | Relevance function for Actual event PO Item Put Away |
| GTT relevance function of Event Type | ESC_DELIV | ZPOF_GTT_EE_DL_ITEM_PKNG_REL | Relevance function for Actual event PO Item Packing |
| GTT relevance function of Event Type | ESC_PURORD | ZPOF_GTT_EE_PO_ITEM_CONF_REL | Relevance function for Actual event PO Item Confirmation |
| GTT relevance function of Event Type | ESC_PURORD | ZPOF_GTT_EE_PO_ITEM_DEL_REL | Relevance function for Actual event PO Item Deletion |
| GTT relevance function of Event Type | ESC_MATDOC | ZPOF_GTT_EE_PO_ITEM_GR_REL | Relevance function for Actual event PO Item Goods Receipt |
| GTT relevance function of Event Type | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_ARR_REL | Relevance function for Actual event Header Arrival |
| GTT relevance function of Event Type | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_CI_REL | Relevance function for Actual event Header Check In |
| GTT relevance function of Event Type | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_DEP_REL | Relevance function for Actual event Header Departure |
| GTT relevance function of Event Type | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_LE_REL | Relevance function for Actual event Header Load End |
| GTT relevance function of Event Type | ESC_SHIPMT | ZPOF_GTT_EE_SH_HDR_LS_REL | Relevance function for Actual event Header Load Start |
| Cross TP Update Function | ESC_PURORD | ZPOF_GTT_CTP_DL_TO_PO | Cross TP Update from Delivery to Purchase Order |
| Cross TP Update Function | ESC_DELIV | ZPOF_GTT_CTP_SH_TO_DL | Cross TP Update from Shipment to Delivery |

# 5: Available Contexts for the extractors’ modules

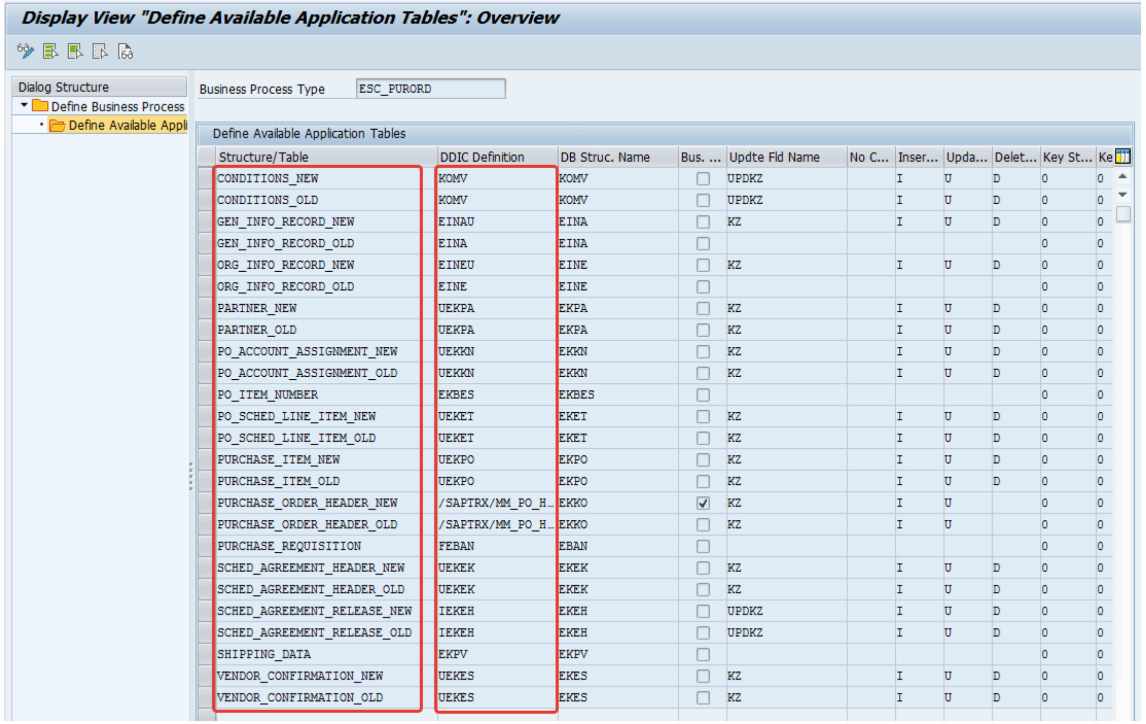
5-1: In **Display IMG** page, click
**Integration with Other SAP Components** -> **Interface to Global Track and Trace** ->
**Define Application Interface**

5-2: Choose activity **Define Business Process Types**

5-3: Please select the Business Process Types to find all the context tables and their structure info.



Structure
- Incentive and Sales Force Management
- Joint Venture Accounting
- Integration with Other SAP Components
  - Business Network Integration
  - Business Network Integration
  - SAP Ariba Integration (SOAP)
  - Integration with SAP Cloud for Customer
  - Integration for Purchasing
  - SAP Localization Hub, tax service
  - Advanced Planning and Optimization
  - Integration with Governance, Risk and Compliance
  - Machine Learning Integration
  - Transportation Management
  - Extended Warehouse Management
  - Supply Network Collaboration/Inventory Collaboration Hub
  - Data Transfer to Business Warehouse
  - Integration with the SAP Digital Payments Add-On
  - Business Packages/Functional Packages
  - SAP Portfolio and Project Management
  - Customer Relationship Management
  - Interface to Global Track and Trace
    - Define System Configuration
    - Define Application Interface
      - Define CI Tenant for SAP GTT
      - Define Business Process Types
      - Define Used Business Process Types, Appl. Object Types and Event Types



Business Process Type ESC_PURORD

Define Available Application Tables

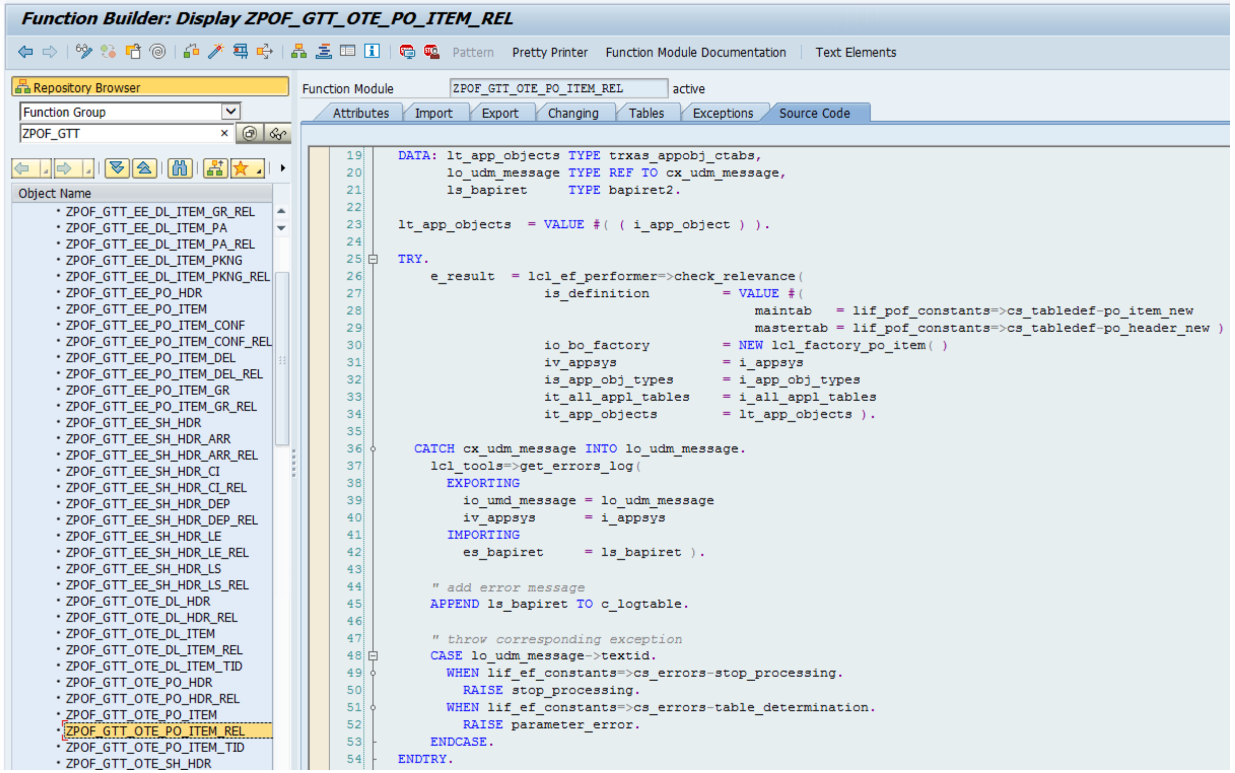
| Structure/Table | DDIC Definition | DB Struc. Name | Bus. ... | Updte Fld Name | No C... | Inser... | Upda... | Delet... | Key St... | Ke |
|---|---|---|---|---|---|---|---|---|---|---|
| CONDITIONS_NEW | KOMV | KOMV | □ | UPDKZ | | I | U | D | 0 | 0 |
| CONDITIONS_OLD | KOMV | KOMV | □ | UPDKZ | | I | U | D | 0 | 0 |
| GEN_INFO_RECORD_NEW | EINAU | EINA | □ | KZ | | I | U | D | 0 | 0 |
| GEN_INFO_RECORD_OLD | EINA | EINA | □ | | | | | | 0 | 0 |
| ORG_INFO_RECORD_NEW | EINEU | EINE | □ | KZ | | I | U | D | 0 | 0 |
| ORG_INFO_RECORD_OLD | EINE | EINE | □ | | | | | | 0 | 0 |
| PARTNER_NEW | UEKPA | EKPA | □ | KZ | | I | U | D | 0 | 0 |
| PARTNER_OLD | UEKPA | EKPA | □ | KZ | | I | U | D | 0 | 0 |
| PO_ACCOUNT_ASSIGNMENT_NEW | UEKKN | EKKN | □ | KZ | | I | U | D | 0 | 0 |
| PO_ACCOUNT_ASSIGNMENT_OLD | UEKKN | EKKN | □ | KZ | | I | U | D | 0 | 0 |
| PO_ITEM_NUMBER | EKBES | EKBES | □ | | | | | | 0 | 0 |
| PO_SCHED_LINE_ITEM_NEW | UEKET | EKET | □ | KZ | | I | U | D | 0 | 0 |
| PO_SCHED_LINE_ITEM_OLD | UEKET | EKET | □ | KZ | | I | U | D | 0 | 0 |
| PURCHASE_ITEM_NEW | UEKPO | EKPO | □ | KZ | | I | U | D | 0 | 0 |
| PURCHASE_ITEM_OLD | UEKPO | EKPO | □ | KZ | | I | U | D | 0 | 0 |
| PURCHASE_ORDER_HEADER_NEW | /SAPTRX/MM_PO_H... | EKKO | ☑ | KZ | | I | U | | 0 | 0 |
| PURCHASE_ORDER_HEADER_OLD | /SAPTRX/MM_PO_H... | EKKO | □ | KZ | | I | U | | 0 | 0 |
| PURCHASE_REQUISITION | FEBAN | EBAN | □ | | | | | | 0 | 0 |
| SCHED_AGREEMENT_HEADER_NEW | UEKEK | EKEK | □ | KZ | | I | U | D | 0 | 0 |
| SCHED_AGREEMENT_HEADER_OLD | UEKEK | EKEK | □ | KZ | | I | U | D | 0 | 0 |
| SCHED_AGREEMENT_RELEASE_NEW | IEKEH | EKEH | □ | UPDKZ | | I | U | D | 0 | 0 |
| SCHED_AGREEMENT_RELEASE_OLD | IEKEH | EKEH | □ | UPDKZ | | I | U | D | 0 | 0 |
| SHIPPING_DATA | EKPV | EKPV | □ | | | | | | 0 | 0 |
| VENDOR_CONFIRMATION_NEW | UEKES | EKES | □ | KZ | | I | U | D | 0 | 0 |
| VENDOR_CONFIRMATION_OLD | UEKES | EKES | □ | KZ | | I | U | D | 0 | 0 |

# 6: Coding Tips in the GTT relevance function modules

To customize the GTT relevance function modules, key points are as below:

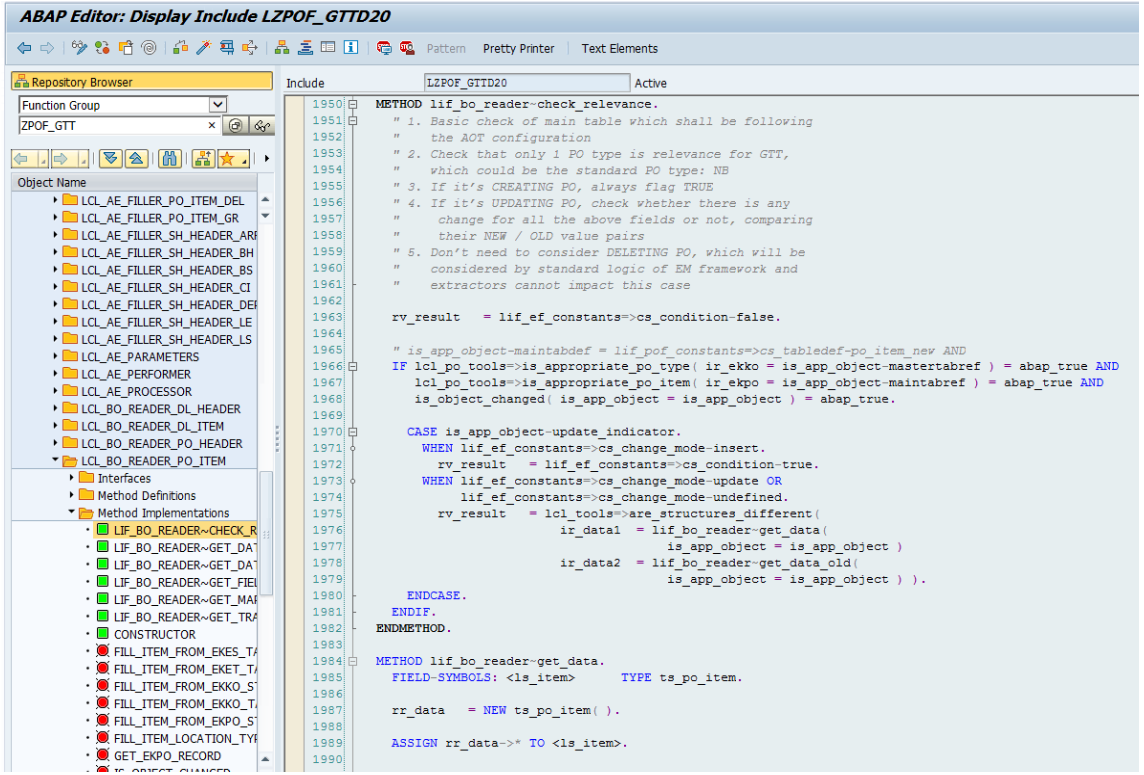
1. Make sure that the Main / Master tables are following the configuration of corresponding AOT or Event Type.
2. Add customization logics to determine the output parameters *E_RESULT*.

See sample code of function: *ZPOF_GTT_OTE_PO_ITEM_REL*



Function Module ZPOF_GTT_OTE_PO_ITEM_REL active

```
DATA: lt_app_objects TYPE trxas_appobj_ctabs,
      lo_udm_message TYPE REF TO cx_udm_message,
      ls_bapiret     TYPE bapiret2.

lt_app_objects  = VALUE #( ( i_app_object ) ).

TRY.
    e_result  = lcl_ef_performer=>check_relevance(
                  is_definition         = VALUE #(
                                              maintab   = lif_pof_constants=>cs_tabledef-po_item_new
                                              mastertab = lif_pof_constants=>cs_tabledef-po_header_new )
                  io_bo_factory         = NEW lcl_factory_po_item( )
                  iv_appsys             = i_appsys
                  is_app_obj_types      = i_app_obj_types
                  it_all_appl_tables    = i_all_appl_tables
                  it_app_objects        = lt_app_objects ).

  CATCH cx_udm_message INTO lo_udm_message.
    lcl_tools=>get_errors_log(
      EXPORTING
        io_umd_message = lo_udm_message
        iv_appsys      = i_appsys
      IMPORTING
        es_bapiret     = ls_bapiret ).

    " add error message
    APPEND ls_bapiret TO c_logtable.

    " throw corresponding exception
    CASE lo_udm_message->textid.
      WHEN lif_ef_constants=>cs_errors-stop_processing.
        RAISE stop_processing.
      WHEN lif_ef_constants=>cs_errors-table_determination.
        RAISE parameter_error.
    ENDCASE.
ENDTRY.
```



Include LZPOF_GTTD20 Active

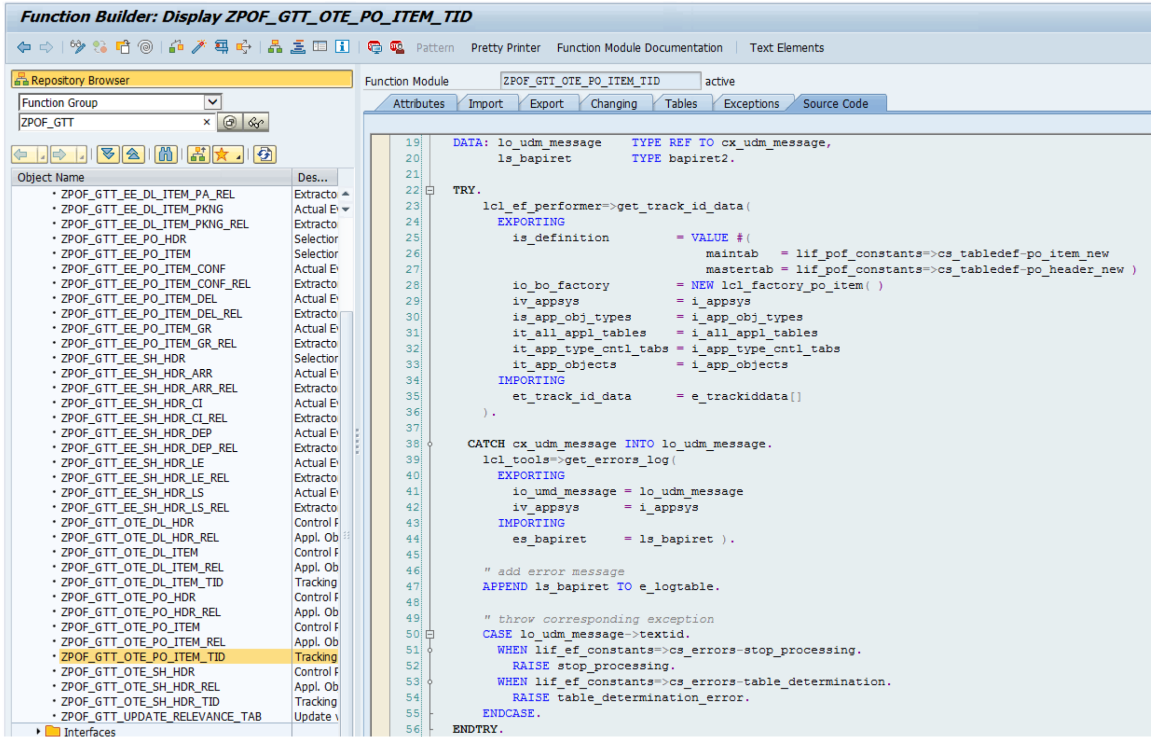
```
METHOD lif_bo_reader~check_relevance.
  " 1. Basic check of main table which shall be following
  "    the AOT configuration
  " 2. Check that only 1 PO type is relevance for GTT,
  "    which could be the standard PO type: NB
  " 3. If it’s CREATING PO, always flag TRUE
  " 4. If it’s UPDATING PO, check whether there is any
  "     change for all the above fields or not, comparing
  "     their NEW / OLD value pairs
  " 5. Don’t need to consider DELETING PO, which will be
  "    considered by standard logic of EM framework and
  "    extractors cannot impact this case

  rv_result   = lif_ef_constants=>cs_condition-false.

  " is_app_object-maintabdef = lif_pof_constants=>cs_tabledef-po_item_new AND
  IF lcl_po_tools=>is_appropriate_po_type( ir_ekko = is_app_object-mastertabref ) = abap_true AND
     lcl_po_tools=>is_appropriate_po_item( ir_ekpo = is_app_object-maintabref ) = abap_true AND
     is_object_changed( is_app_object = is_app_object ) = abap_true.

    CASE is_app_object-update_indicator.
      WHEN lif_ef_constants=>cs_change_mode-insert.
        rv_result   = lif_ef_constants=>cs_condition-true.
      WHEN lif_ef_constants=>cs_change_mode-update OR
           lif_ef_constants=>cs_change_mode-undefined.
        rv_result   = lcl_tools=>are_structures_different(
                        ir_data1  = lif_bo_reader~get_data(
                                      is_app_object = is_app_object )
                        ir_data2  = lif_bo_reader~get_data_old(
                                      is_app_object = is_app_object ) ).
    ENDCASE.
  ENDIF.
ENDMETHOD.

METHOD lif_bo_reader~get_data.
  FIELD-SYMBOLS: <ls_item>      TYPE ts_po_item.

  rr_data   = NEW ts_po_item( ).

  ASSIGN rr_data->* TO <ls_item>.
```

# 7: Coding Tips in the Tracking ID function modules

To customize the Tracking ID function modules, key points are as below:

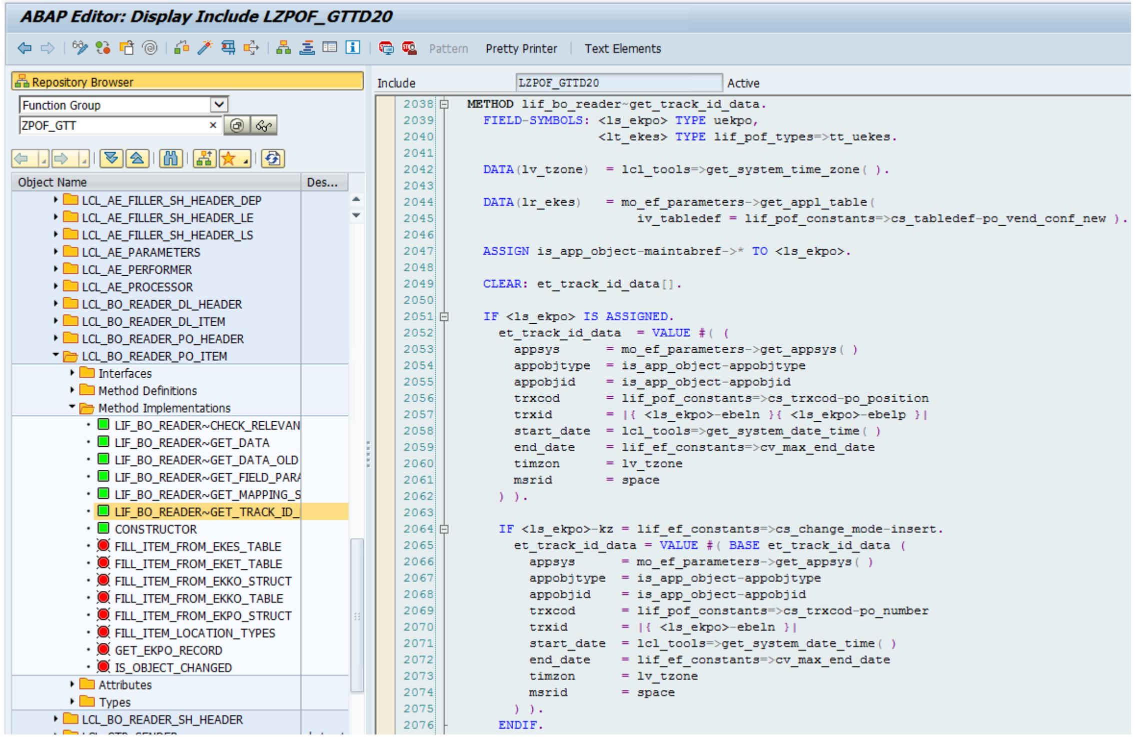
1. Make sure that the Main / Master tables are following the configuration of corresponding AOT.
2. Add customization logics to fill the output table *E_TRACKIDDATA*.
3. The Tracking ID Type need to be the same as the definition in the process type of model in Manage Models application.
4. GTT v2 accepts delta transport for tracking IDs, which means that only the newly-created / changed / deleted tracking IDs shall be filled, while the ones without change need to be ignored in the logic.
5. The tracking ID for its own process type needs to be filled for each process update.
6. In case of tracking ID deletion, the field *ACTION* shall be filled with ‘D’.

See sample code of function: *ZPOF_GTT_OTE_PO_ITEM_TID*



| Object Name | Des... |
|---|---|
| ZPOF_GTT_EE_DL_ITEM_PA_REL | Extracto |
| ZPOF_GTT_EE_DL_ITEM_PKNG | Actual E |
| ZPOF_GTT_EE_DL_ITEM_PKNG_REL | Extracto |
| ZPOF_GTT_EE_PO_HDR | Selection |
| ZPOF_GTT_EE_PO_ITEM | Selection |
| ZPOF_GTT_EE_PO_ITEM_CONF | Actual E |
| ZPOF_GTT_EE_PO_ITEM_CONF_REL | Extracto |
| ZPOF_GTT_EE_PO_ITEM_DEL | Actual E |
| ZPOF_GTT_EE_PO_ITEM_DEL_REL | Extracto |
| ZPOF_GTT_EE_PO_ITEM_GR | Actual E |
| ZPOF_GTT_EE_PO_ITEM_GR_REL | Extracto |
| ZPOF_GTT_EE_SH_HDR | Selection |
| ZPOF_GTT_EE_SH_HDR_ARR | Actual E |
| ZPOF_GTT_EE_SH_HDR_ARR_REL | Extracto |
| ZPOF_GTT_EE_SH_HDR_CI | Actual E |
| ZPOF_GTT_EE_SH_HDR_CI_REL | Extracto |
| ZPOF_GTT_EE_SH_HDR_DEP | Actual E |
| ZPOF_GTT_EE_SH_HDR_DEP_REL | Extracto |
| ZPOF_GTT_EE_SH_HDR_LE | Actual E |
| ZPOF_GTT_EE_SH_HDR_LE_REL | Extracto |
| ZPOF_GTT_EE_SH_HDR_LS | Actual E |
| ZPOF_GTT_EE_SH_HDR_LS_REL | Extracto |
| ZPOF_GTT_OTE_DL_HDR | Control F |
| ZPOF_GTT_OTE_DL_HDR_REL | Appl. Ob |
| ZPOF_GTT_OTE_DL_ITEM | Control F |
| ZPOF_GTT_OTE_DL_ITEM_REL | Appl. Ob |
| ZPOF_GTT_OTE_DL_ITEM_TID | Tracking |
| ZPOF_GTT_OTE_PO_HDR | Control F |
| ZPOF_GTT_OTE_PO_HDR_REL | Appl. Ob |
| ZPOF_GTT_OTE_PO_ITEM | Control F |
| ZPOF_GTT_OTE_PO_ITEM_REL | Appl. Ob |
| ZPOF_GTT_OTE_PO_ITEM_TID | Tracking |
| ZPOF_GTT_OTE_SH_HDR | Control F |
| ZPOF_GTT_OTE_SH_HDR_REL | Appl. Ob |
| ZPOF_GTT_OTE_SH_HDR_TID | Tracking |
| ZPOF_GTT_UPDATE_RELEVANCE_TAB | Update \ |

```
  DATA: lo_udm_message    TYPE REF TO cx_udm_message,
        ls_bapiret        TYPE bapiret2.

  TRY.
      lcl_ef_performer=>get_track_id_data(
        EXPORTING
          is_definition         = VALUE #(
                                    maintab   = lif_pof_constants=>cs_tabledef-po_item_new
                                    mastertab = lif_pof_constants=>cs_tabledef-po_header_new )
          io_bo_factory         = NEW lcl_factory_po_item( )
          iv_appsys             = i_appsys
          is_app_obj_types      = i_app_obj_types
          it_all_appl_tables    = i_all_appl_tables
          it_app_type_cntl_tabs = i_app_type_cntl_tabs
          it_app_objects        = i_app_objects
        IMPORTING
          et_track_id_data      = e_trackiddata[]
      ).

    CATCH cx_udm_message INTO lo_udm_message.
      lcl_tools=>get_errors_log(
        EXPORTING
          io_umd_message = lo_udm_message
          iv_appsys      = i_appsys
        IMPORTING
          es_bapiret     = ls_bapiret ).

      " add error message
      APPEND ls_bapiret TO e_logtable.

      " throw corresponding exception
      CASE lo_udm_message->textid.
        WHEN lif_ef_constants=>cs_errors-stop_processing.
          RAISE stop_processing.
        WHEN lif_ef_constants=>cs_errors-table_determination.
          RAISE table_determination_error.
      ENDCASE.
  ENDTRY.
```

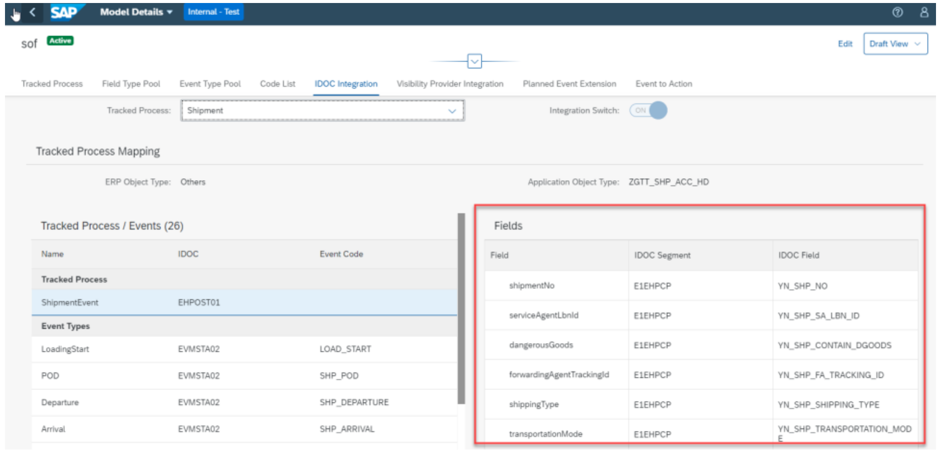


```
  METHOD lif_bo_reader~get_track_id_data.
    FIELD-SYMBOLS: <ls_ekpo> TYPE uekpo,
                   <lt_ekes> TYPE lif_pof_types=>tt_uekes.

    DATA(lv_tzone)  = lcl_tools=>get_system_time_zone( ).

    DATA(lr_ekes)   = mo_ef_parameters->get_appl_table(
                        iv_tabledef = lif_pof_constants=>cs_tabledef-po_vend_conf_new ).

    ASSIGN is_app_object-maintabref->* TO <ls_ekpo>.

    CLEAR: et_track_id_data[].

    IF <ls_ekpo> IS ASSIGNED.
      et_track_id_data  = VALUE #( (
        appsys      = mo_ef_parameters->get_appsys( )
        appobjtype  = is_app_object-appobjtype
        appobjid    = is_app_object-appobjid
        trxcod      = lif_pof_constants=>cs_trxcod-po_position
        trxid       = |{ <ls_ekpo>-ebeln }{ <ls_ekpo>-ebelp }|
        start_date  = lcl_tools=>get_system_date_time( )
        end_date    = lif_ef_constants=>cv_max_end_date
        timzon      = lv_tzone
        msrid       = space
      ) ).

      IF <ls_ekpo>-kz = lif_ef_constants=>cs_change_mode-insert.
        et_track_id_data = VALUE #( BASE et_track_id_data (
          appsys      = mo_ef_parameters->get_appsys( )
          appobjtype  = is_app_object-appobjtype
          appobjid    = is_app_object-appobjid
          trxcod      = lif_pof_constants=>cs_trxcod-po_number
          trxid       = |{ <ls_ekpo>-ebeln }|
          start_date  = lcl_tools=>get_system_date_time( )
          end_date    = lif_ef_constants=>cv_max_end_date
          timzon      = lv_tzone
          msrid       = space
        ) ).
      ENDIF.
```

# 8: Coding Tips in the Control Parameter function modules

## To customize the Control Parameter function modules, key points are as below:

1. Make sure that the Main / Master tables are following the configuration of corresponding AOT.
2. Add customization logics to fill the output table *E_CONTROL_DATA*.
3. GTT v2 asks for full transport for all the control parameters, which means that all the fields needs to be extracted in all cases, no matter whether their values have been changed.
4. To fill up the composition (table) fields defined in Manage Model applications, use the parameter field *PARAMINDEX* to specify the line number. If the field is empty, GTT regards it as a simple flat field.
5. To clear a composition, fill the key field using invalid values, for which key attribute has been checked in Manage Model application. It's not recommended to fill a code list type field to clear a composition even if it's a key field.
6. The field with fixed name 'ACTUAL_BUSINESS_DATETIME' and 'ACTUAL_BUSINESS_TIMEZONE' are mandatory fields to be transported for event handling sequencing in GTT V2.
7. In Manage Model application, click tab *IDOC Integration* to map the parameter names and model field names.
8. For DATE or DATETIME fields, when the source value is initial like '00000000' '000000000000000', then please ensure to only enable *PARAMNAME and PARAMINDEX in the extractor code, not enable VALUE for IDOC sending.*
9. For Amount field which has reference currency, please ensure to call BAPI 'BAPI_CURRENCY_CONV_TO_EXTERNAL' using the reference currency to make the amount tracked correctly by GTT v2. The BAPI will output the conversion result in 4 decimals as fixed, which needs additional rounding in the extractor if the corresponding field defined in the tracking model is of less then 4 decimals.



Tracked Process: Shipment

Integration Switch: ON

Tracked Process Mapping

ERP Object Type: Others

Application Object Type: ZGTT_SHP_ACC_HD

Tracked Process / Events (26)

| Name | IDOC | Event Code |
|---|---|---|
| **Tracked Process** | | |
| ShipmentEvent | EHPOST01 | |
| **Event Types** | | |
| LoadingStart | EVMSTA02 | LOAD_START |
| POD | EVMSTA02 | SHP_POD |
| Departure | EVMSTA02 | SHP_DEPARTURE |
| Arrival | EVMSTA02 | SHP_ARRIVAL |

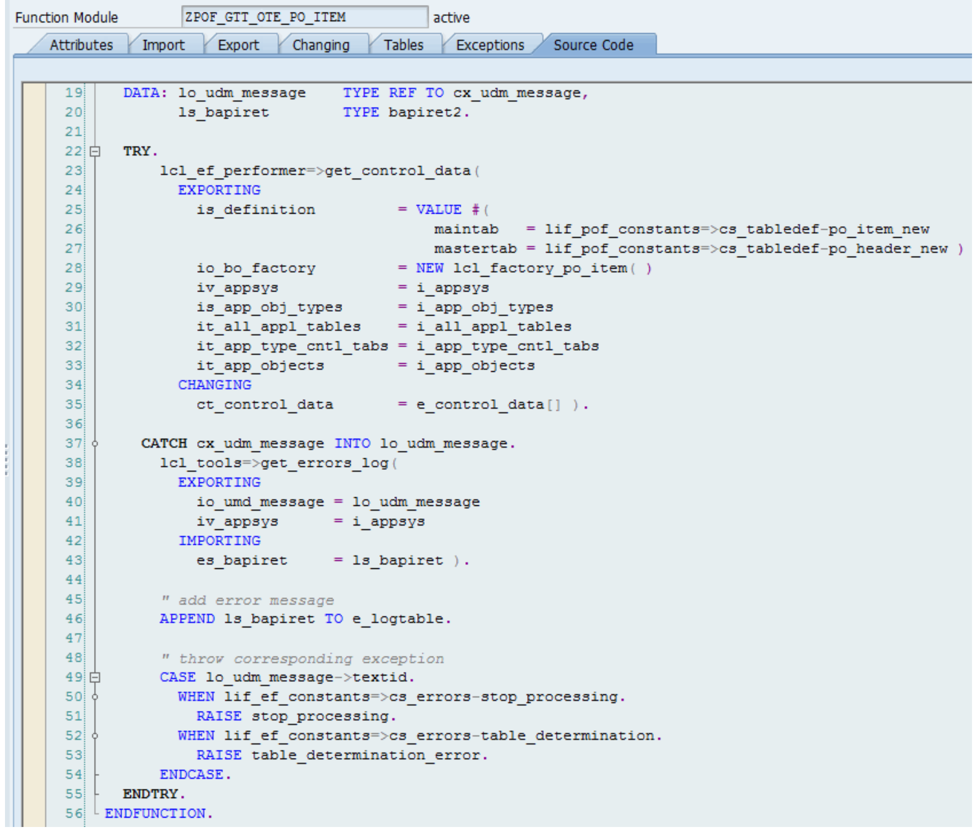
Fields

| Field | IDOC Segment | IDOC Field |
|---|---|---|
| shipmentNo | E1EHPCP | YN_SHP_NO |
| serviceAgentLbnId | E1EHPCP | YN_SHP_SA_LBN_ID |
| dangerousGoods | E1EHPCP | YN_SHP_CONTAIN_DGOODS |
| forwardingAgentTrackingId | E1EHPCP | YN_SHP_FA_TRACKING_ID |
| shippingType | E1EHPCP | YN_SHP_SHIPPING_TYPE |
| transportationMode | E1EHPCP | YN_SHP_TRANSPORTATION_MODE |

## See sample code of function: *ZPOF_GTT_OTE_PO_ITEM*

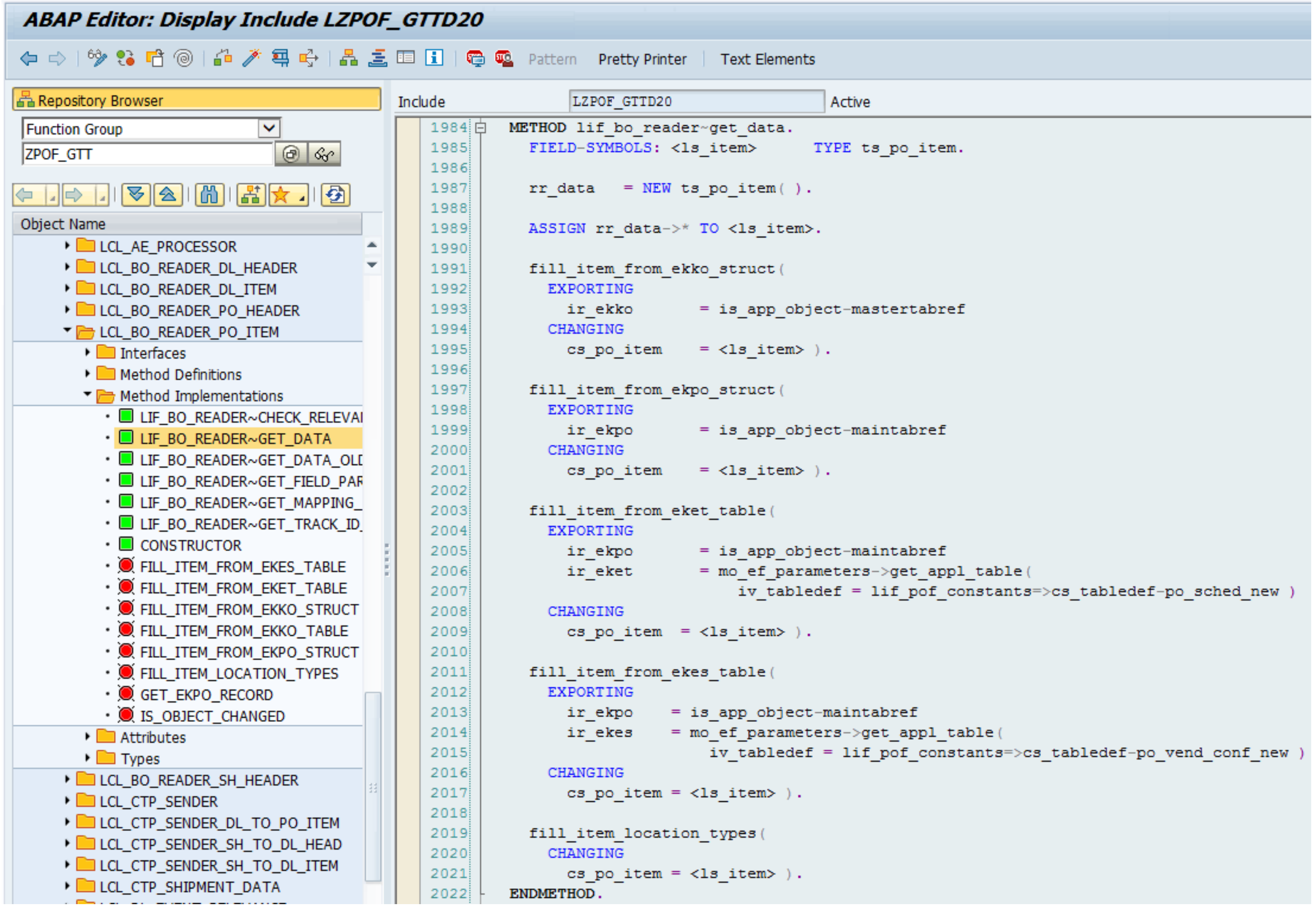
# 8: Coding Tips in the Control Parameter function modules

Main logic of Purchase Order Item is implemented in class LCL_BO_READER_PO_ITEM



Function Module ZPOF_GTT_OTE_PO_ITEM active

```
DATA: lo_udm_message    TYPE REF TO cx_udm_message,
      ls_bapiret        TYPE bapiret2.

TRY.
    lcl_ef_performer=>get_control_data(
      EXPORTING
        is_definition          = VALUE #(
                                   maintab   = lif_pof_constants=>cs_tabledef-po_item_new
                                   mastertab = lif_pof_constants=>cs_tabledef-po_header_new )
        io_bo_factory          = NEW lcl_factory_po_item( )
        iv_appsys              = i_appsys
        is_app_obj_types       = i_app_obj_types
        it_all_appl_tables     = i_all_appl_tables
        it_app_type_cntl_tabs  = i_app_type_cntl_tabs
        it_app_objects         = i_app_objects
      CHANGING
        ct_control_data        = e_control_data[] ).

  CATCH cx_udm_message INTO lo_udm_message.
    lcl_tools=>get_errors_log(
      EXPORTING
        io_umd_message = lo_udm_message
        iv_appsys      = i_appsys
      IMPORTING
        es_bapiret     = ls_bapiret ).

    " add error message
    APPEND ls_bapiret TO e_logtable.

    " throw corresponding exception
    CASE lo_udm_message->textid.
      WHEN lif_ef_constants=>cs_errors-stop_processing.
        RAISE stop_processing.
      WHEN lif_ef_constants=>cs_errors-table_determination.
        RAISE table_determination_error.
    ENDCASE.
ENDTRY.
ENDFUNCTION.
```


ABAP Editor: Display Include LZPOF_GTTD20
Pattern
Pretty Printer
Text Elements
Repository Browser
Object Name
LCL_AE_PROCESSOR
LCL_BO_READER_DL_HEADER
LCL_BO_READER_DL_ITEM
LCL_BO_READER_PO_HEADER
LCL_BO_READER_PO_ITEM
Interfaces
Method Definitions
Method Implementations
LIF_BO_READER~CHECK_RELEVAI
LIF_BO_READER~GET_DATA
LIF_BO_READER~GET_DATA_OLI
LIF_BO_READER~GET_FIELD_PAR
LIF_BO_READER~GET_MAPPING_
LIF_BO_READER~GET_TRACK_ID_
CONSTRUCTOR
FILL_ITEM_FROM_EKES_TABLE
FILL_ITEM_FROM_EKET_TABLE
FILL_ITEM_FROM_EKKO_STRUCT
FILL_ITEM_FROM_EKKO_TABLE
FILL_ITEM_FROM_EKPO_STRUCT
FILL_ITEM_LOCATION_TYPES
GET_EKPO_RECORD
IS_OBJECT_CHANGED
Attributes
Types
LCL_BO_READER_SH_HEADER
LCL_CTP_SENDER
LCL_CTP_SENDER_DL_TO_PO_ITEM
LCL_CTP_SENDER_SH_TO_DL_HEAD
LCL_CTP_SENDER_SH_TO_DL_ITEM
LCL_CTP_SHIPMENT_DATA

Function Group
ZPOF_GTT

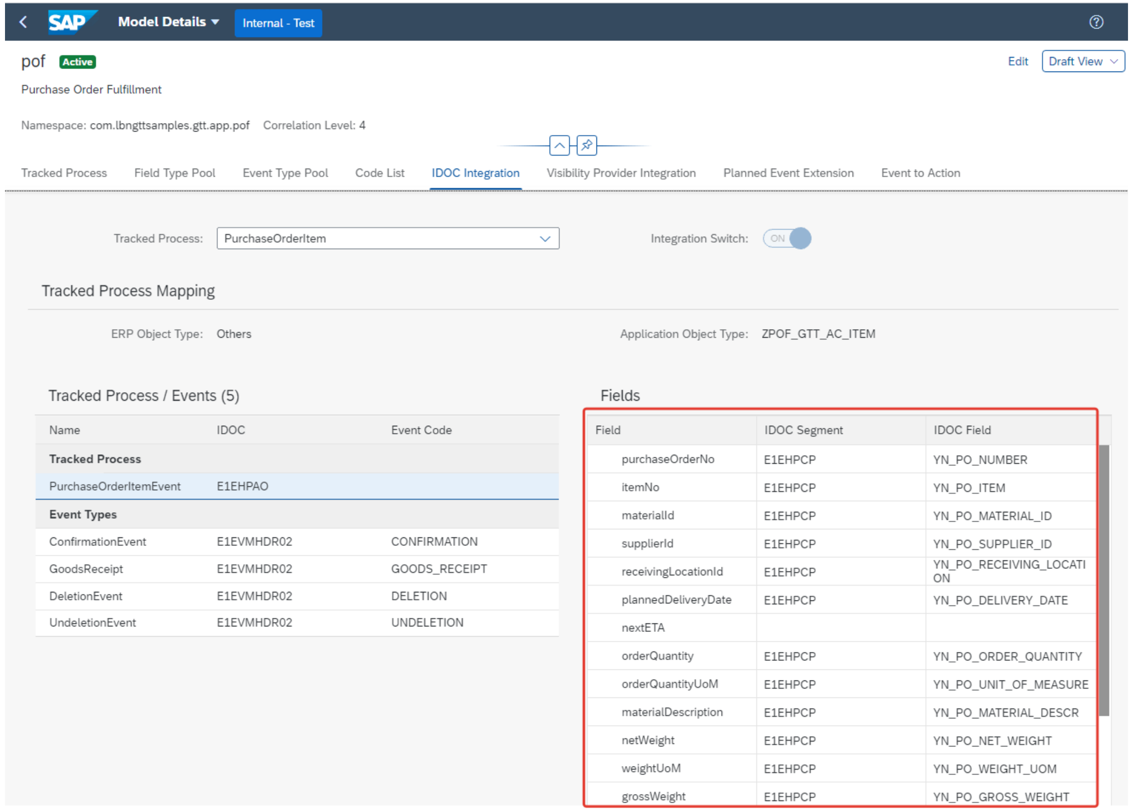
Include LZPOF_GTTD20 Active

```
METHOD lif_bo_reader~get_data.
  FIELD-SYMBOLS: <ls_item>      TYPE ts_po_item.

  rr_data    = NEW ts_po_item( ).

  ASSIGN rr_data->* TO <ls_item>.

  fill_item_from_ekko_struct(
    EXPORTING
      ir_ekko        = is_app_object-mastertabref
    CHANGING
      cs_po_item     = <ls_item> ).

  fill_item_from_ekpo_struct(
    EXPORTING
      ir_ekpo        = is_app_object-maintabref
    CHANGING
      cs_po_item     = <ls_item> ).

  fill_item_from_eket_table(
    EXPORTING
      ir_ekpo        = is_app_object-maintabref
      ir_eket        = mo_ef_parameters->get_appl_table(
                         iv_tabledef = lif_pof_constants=>cs_tabledef-po_sched_new )
    CHANGING
      cs_po_item   = <ls_item> ).

  fill_item_from_ekes_table(
    EXPORTING
      ir_ekpo     = is_app_object-maintabref
      ir_ekes     = mo_ef_parameters->get_appl_table(
                      iv_tabledef = lif_pof_constants=>cs_tabledef-po_vend_conf_new )
    CHANGING
      cs_po_item = <ls_item> ).

  fill_item_location_types(
    CHANGING
      cs_po_item = <ls_item> ).
ENDMETHOD.
```

# 9: Coding Tips in the Planned Event function modules

To customize the Planned Event function modules, key points are as below:

1. Make sure that the Main / Master tables are following the configuration of corresponding AOT.
2. Add customization logics to fill the output table *E_EXPEVENTDATA*.
3. GTT v2 asks for full transport for all the planned events, which means that all the events needs to be extracted in all cases, no matter whether their values have been changed.
4. The field *MILESTONE* is mandatory to be transported.
5. The field *EVT_EXP_DATETIME* is optional, but need to be filled with relevant time zone *EVT_EXP_TZONE* together if it needs to be transported.
6. The field *LOC_ID1* is optional, but need to be filled with relevant location type *LOCTYPE* together if it needs to be transported. The values for field *LOCTYPE* are limited by *Manage Locations* application in GTT V2.
7. The field *LOCID2* is mandatory to specify the stop ID (match key) in case of shipment tracking.



pof Active

Purchase Order Fulfillment

Namespace: com.lbngttsamples.gtt.app.pof   Correlation Level: 4

Tracked Process: PurchaseOrderItem

Integration Switch: ON

Tracked Process Mapping

ERP Object Type: Others

Application Object Type: ZPOF_GTT_AC_ITEM

Tracked Process / Events (5)

| Name | IDOC | Event Code |
|---|---|---|
| **Tracked Process** | | |
| PurchaseOrderItemEvent | E1EHPAO | |
| **Event Types** | | |
| ConfirmationEvent | E1EVMHDR02 | CONFIRMATION |
| GoodsReceipt | E1EVMHDR02 | GOODS_RECEIPT |
| DeletionEvent | E1EVMHDR02 | DELETION |
| UndeletionEvent | E1EVMHDR02 | UNDELETION |

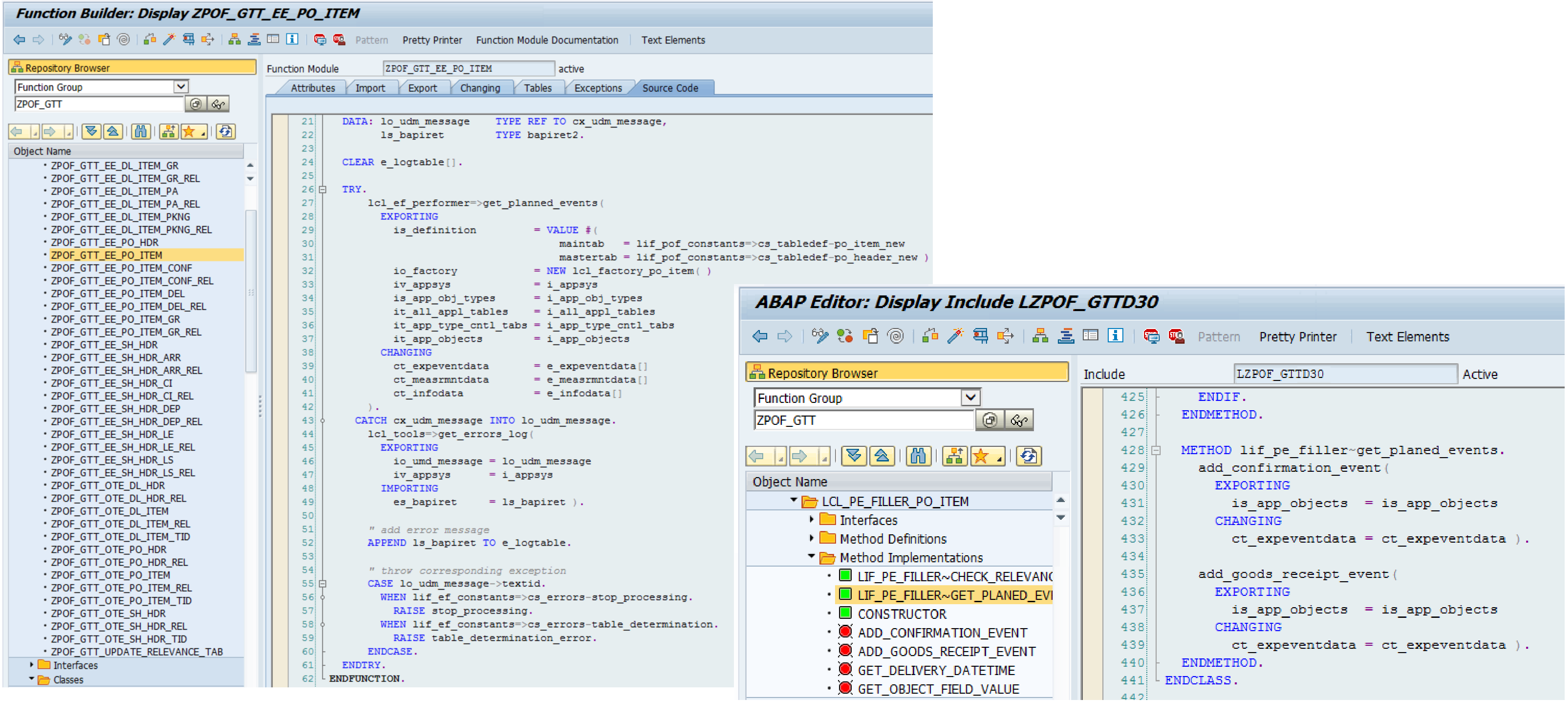
Fields

| Field | IDOC Segment | IDOC Field |
|---|---|---|
| purchaseOrderNo | E1EHPCP | YN_PO_NUMBER |
| itemNo | E1EHPCP | YN_PO_ITEM |
| materialId | E1EHPCP | YN_PO_MATERIAL_ID |
| supplierId | E1EHPCP | YN_PO_SUPPLIER_ID |
| receivingLocationId | E1EHPCP | YN_PO_RECEIVING_LOCATION |
| plannedDeliveryDate | E1EHPCP | YN_PO_DELIVERY_DATE |
| nextETA | | |
| orderQuantity | E1EHPCP | YN_PO_ORDER_QUANTITY |
| orderQuantityUoM | E1EHPCP | YN_PO_UNIT_OF_MEASURE |
| materialDescription | E1EHPCP | YN_PO_MATERIAL_DESCR |
| netWeight | E1EHPCP | YN_PO_NET_WEIGHT |
| weightUoM | E1EHPCP | YN_PO_WEIGHT_UOM |
| grossWeight | E1EHPCP | YN_PO_GROSS_WEIGHT |

See sample code of function: *ZPOF_GTT_EE_PO_ITEM*

# 9: Coding Tips in the Planned Event function modules

Main logic of Purchase Order Item Planned Events is implemented in class LCL_PE_FILLER_PO_ITEM



Function Module ZPOF_GTT_EE_PO_ITEM active

```
  DATA: lo_udm_message    TYPE REF TO cx_udm_message,
        ls_bapiret        TYPE bapiret2.

  CLEAR e_logtable[].

  TRY.
      lcl_ef_performer=>get_planned_events(
        EXPORTING
          is_definition          = VALUE #(
                                     maintab   = lif_pof_constants=>cs_tabledef-po_item_new
                                     mastertab = lif_pof_constants=>cs_tabledef-po_header_new )
          io_factory             = NEW lcl_factory_po_item( )
          iv_appsys              = i_appsys
          is_app_obj_types       = i_app_obj_types
          it_all_appl_tables     = i_all_appl_tables
          it_app_type_cntl_tabs = i_app_type_cntl_tabs
          it_app_objects         = i_app_objects
        CHANGING
          ct_expeventdata        = e_expeventdata[]
          ct_measrmntdata        = e_measrmntdata[]
          ct_infodata            = e_infodata[]
      ).
    CATCH cx_udm_message INTO lo_udm_message.
      lcl_tools=>get_errors_log(
        EXPORTING
          io_umd_message = lo_udm_message
          iv_appsys      = i_appsys
        IMPORTING
          es_bapiret     = ls_bapiret ).

      " add error message
      APPEND ls_bapiret TO e_logtable.

      " throw corresponding exception
      CASE lo_udm_message->textid.
        WHEN lif_ef_constants=>cs_errors-stop_processing.
          RAISE stop_processing.
        WHEN lif_ef_constants=>cs_errors-table_determination.
          RAISE table_determination_error.
      ENDCASE.
  ENDTRY.
ENDFUNCTION.
```

Include LZPOF_GTTD30 Active

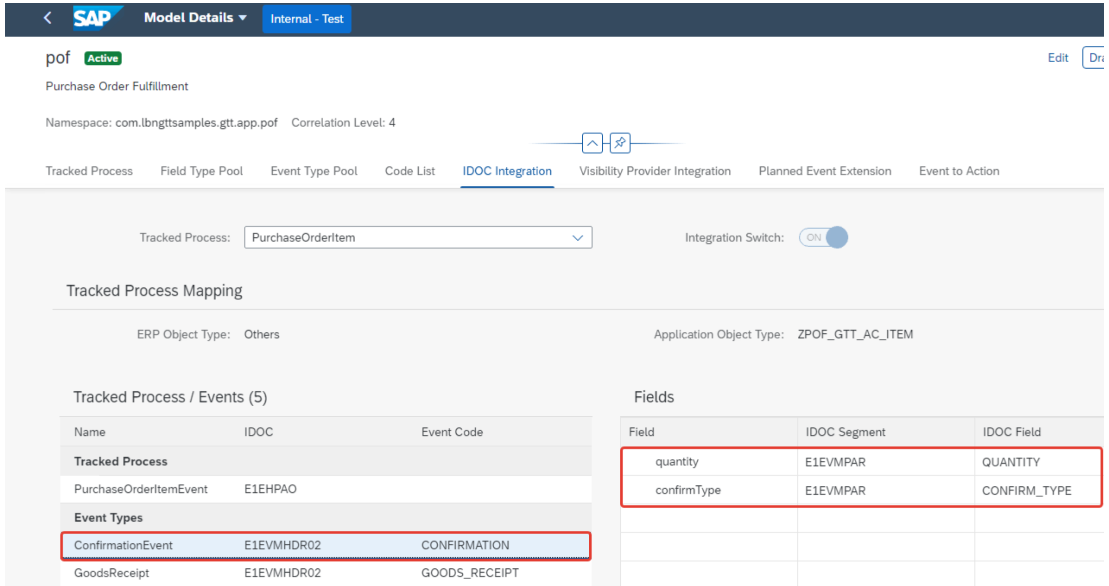
```
    ENDIF.
  ENDMETHOD.

  METHOD lif_pe_filler~get_planed_events.
    add_confirmation_event(
      EXPORTING
        is_app_objects  = is_app_objects
      CHANGING
        ct_expeventdata = ct_expeventdata ).

    add_goods_receipt_event(
      EXPORTING
        is_app_objects  = is_app_objects
      CHANGING
        ct_expeventdata = ct_expeventdata ).
  ENDMETHOD.
ENDCLASS.
```

# 10: Coding Tips in the Event Data function modules

To customize the Event Data function modules, key points are as below:

1. Make sure that the Main / Master tables are following the configuration of corresponding Event Type.
2. Add customization logics to fill the output table *CT_TRACKINGHEADER, CT_TRACKLOCATION, C_EVENTID_MAP.*
3. If the event has user-defined fields in Manage Models application, fill the table *CT_TRACKPARAMETERS.*
4. If the event has reference table information, fill the table *CT_TRACKREFERENCES.*
5. The field *CT_TRACKINGHEADER-SRCCOD, SRCID, SRCTX* is used for event reason transport.
6. In Manage Model application, click tab IDOC Integration to map the user-defined parameter names and model field names.

See sample code of function: *ZPOF_GTT_EE_PO_ITEM_CONF*



pof Active

Purchase Order Fulfillment

Namespace: com.lbngttsamples.gtt.app.pof Correlation Level: 4

Tracked Process: PurchaseOrderItem

Integration Switch: ON

Tracked Process Mapping

ERP Object Type: Others

Application Object Type: ZPOF_GTT_AC_ITEM

Tracked Process / Events (5)

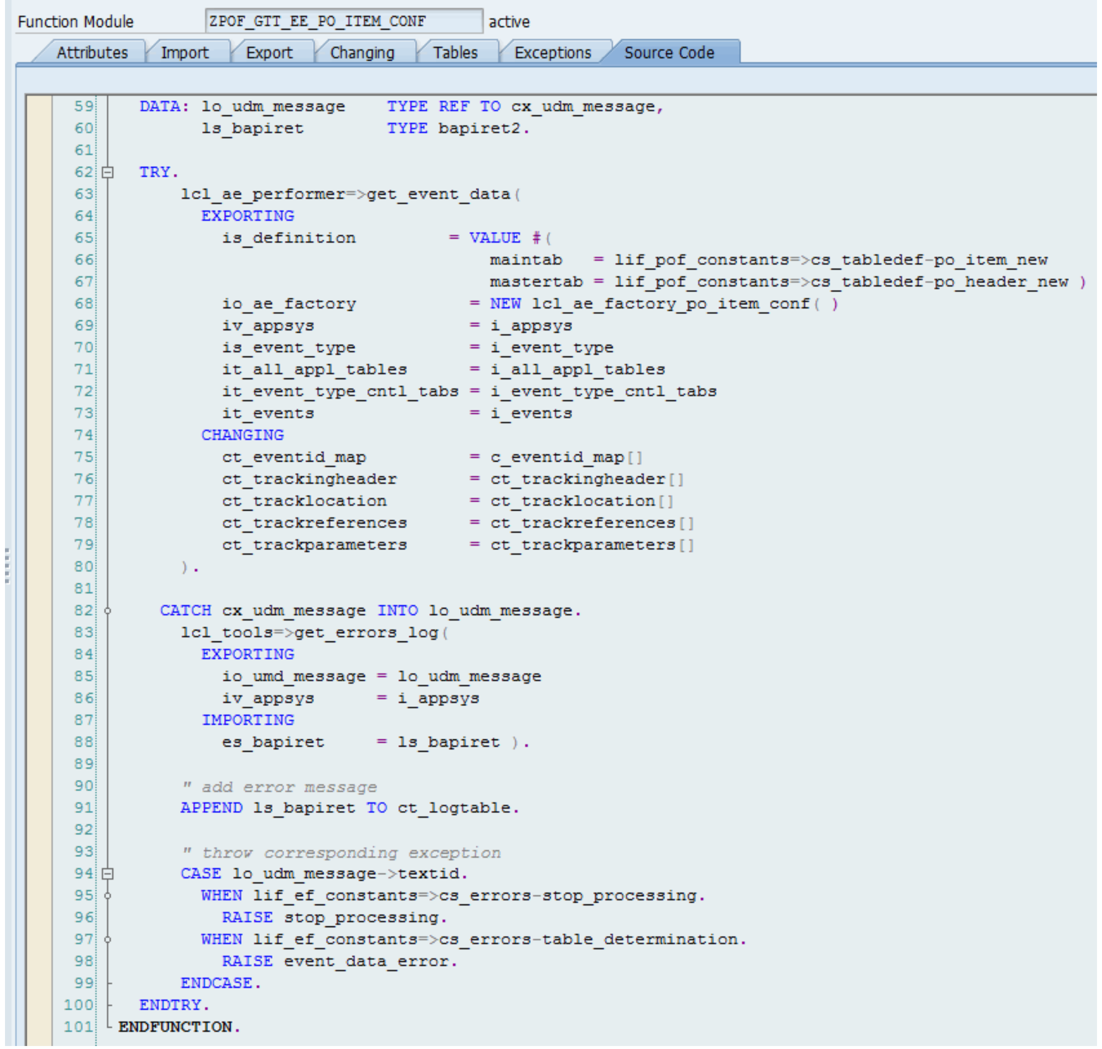
| Name | IDOC | Event Code |
|---|---|---|
| **Tracked Process** | | |
| PurchaseOrderItemEvent | E1EHPAO | |
| **Event Types** | | |
| ConfirmationEvent | E1EVMHDR02 | CONFIRMATION |
| GoodsReceipt | E1EVMHDR02 | GOODS_RECEIPT |

Fields

| Field | IDOC Segment | IDOC Field |
|---|---|---|
| quantity | E1EVMPAR | QUANTITY |
| confirmType | E1EVMPAR | CONFIRM_TYPE |

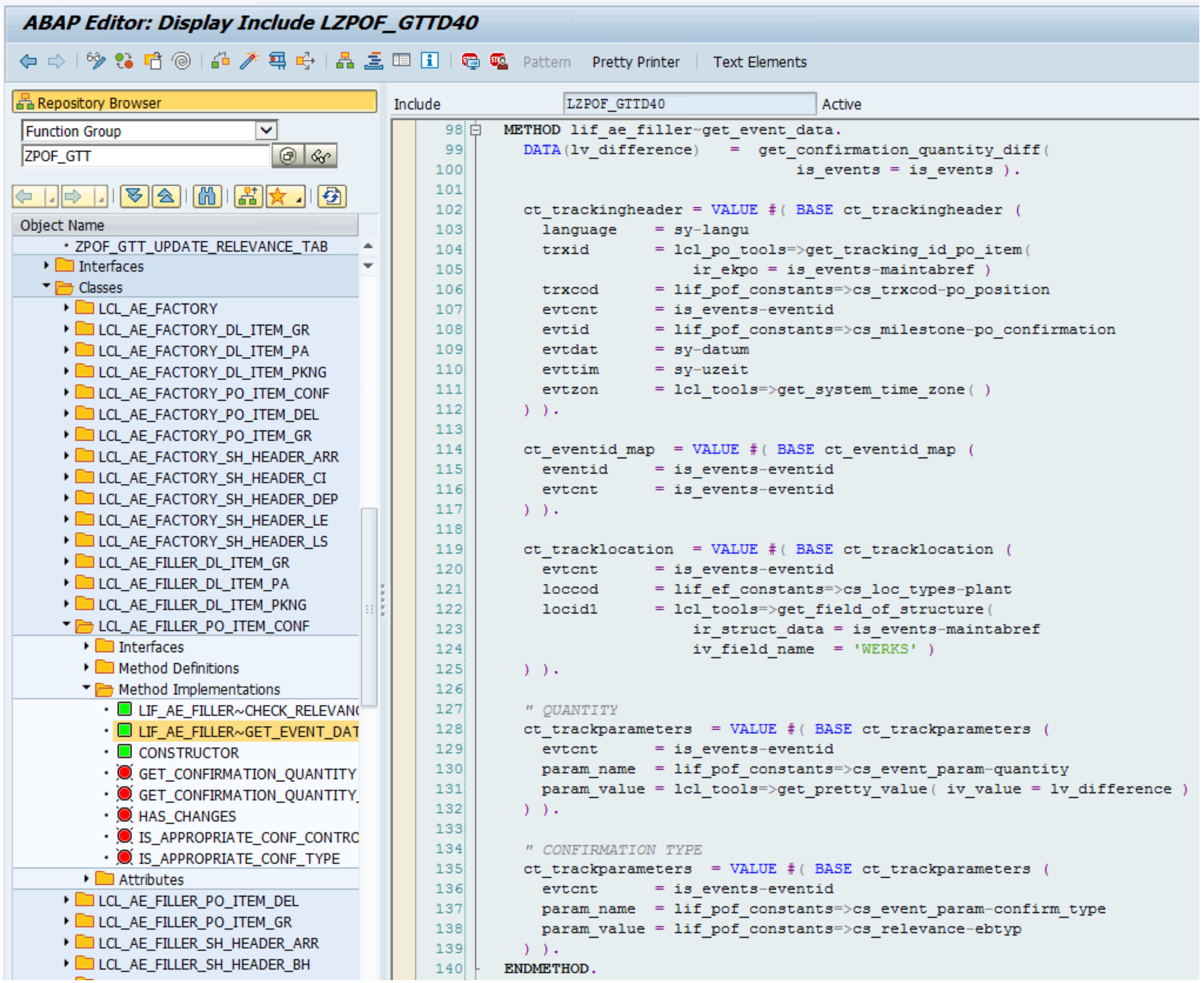
# 10: Coding Tips in the Event Data function modules

Main logic of Purchase Order Item Confirmation event is implemented in class LCL_AE_FILLER_PO_ITEM_CONF



Function Module ZPOF_GTT_EE_PO_ITEM_CONF active

```
  DATA: lo_udm_message    TYPE REF TO cx_udm_message,
        ls_bapiret        TYPE bapiret2.

  TRY.
      lcl_ae_performer=>get_event_data(
        EXPORTING
          is_definition         = VALUE #(
                                    maintab   = lif_pof_constants=>cs_tabledef-po_item_new
                                    mastertab = lif_pof_constants=>cs_tabledef-po_header_new )
          io_ae_factory           = NEW lcl_ae_factory_po_item_conf( )
          iv_appsys               = i_appsys
          is_event_type           = i_event_type
          it_all_appl_tables      = i_all_appl_tables
          it_event_type_cntl_tabs = i_event_type_cntl_tabs
          it_events               = i_events
        CHANGING
          ct_eventid_map          = c_eventid_map[]
          ct_trackingheader       = ct_trackingheader[]
          ct_tracklocation        = ct_tracklocation[]
          ct_trackreferences      = ct_trackreferences[]
          ct_trackparameters      = ct_trackparameters[]
      ).

    CATCH cx_udm_message INTO lo_udm_message.
      lcl_tools=>get_errors_log(
        EXPORTING
          io_umd_message = lo_udm_message
          iv_appsys      = i_appsys
        IMPORTING
          es_bapiret     = ls_bapiret ).

      " add error message
      APPEND ls_bapiret TO ct_logtable.

      " throw corresponding exception
      CASE lo_udm_message->textid.
        WHEN lif_ef_constants=>cs_errors-stop_processing.
          RAISE stop_processing.
        WHEN lif_ef_constants=>cs_errors-table_determination.
          RAISE event_data_error.
      ENDCASE.
  ENDTRY.
ENDFUNCTION.
```



ABAP Editor: Display Include LZPOF_GTTD40

Pattern
Pretty Printer
Text Elements
Repository Browser
Function Group
ZPOF_GTT
Object Name
ZPOF_GTT_UPDATE_RELEVANCE_TAB
Interfaces
Classes
LCL_AE_FACTORY
LCL_AE_FACTORY_DL_ITEM_GR
LCL_AE_FACTORY_DL_ITEM_PA
LCL_AE_FACTORY_DL_ITEM_PKNG
LCL_AE_FACTORY_PO_ITEM_CONF
LCL_AE_FACTORY_PO_ITEM_DEL
LCL_AE_FACTORY_PO_ITEM_GR
LCL_AE_FACTORY_SH_HEADER_ARR
LCL_AE_FACTORY_SH_HEADER_CI
LCL_AE_FACTORY_SH_HEADER_DEP
LCL_AE_FACTORY_SH_HEADER_LE
LCL_AE_FACTORY_SH_HEADER_LS
LCL_AE_FILLER_DL_ITEM_GR
LCL_AE_FILLER_DL_ITEM_PA
LCL_AE_FILLER_DL_ITEM_PKNG
LCL_AE_FILLER_PO_ITEM_CONF
Interfaces
Method Definitions
Method Implementations
LIF_AE_FILLER~CHECK_RELEVANC
LIF_AE_FILLER~GET_EVENT_DAT
CONSTRUCTOR
GET_CONFIRMATION_QUANTITY
GET_CONFIRMATION_QUANTITY_
HAS_CHANGES
IS_APPROPRIATE_CONF_CONTRC
IS_APPROPRIATE_CONF_TYPE
Attributes
LCL_AE_FILLER_PO_ITEM_DEL
LCL_AE_FILLER_PO_ITEM_GR
LCL_AE_FILLER_SH_HEADER_ARR
LCL_AE_FILLER_SH_HEADER_BH

Include LZPOF_GTTD40 Active

```
METHOD lif_ae_filler~get_event_data.
  DATA(lv_difference)   =  get_confirmation_quantity_diff(
                              is_events = is_events ).

  ct_trackingheader = VALUE #( BASE ct_trackingheader (
    language     = sy-langu
    trxid        = lcl_po_tools=>get_tracking_id_po_item(
                     ir_ekpo = is_events-maintabref )
    trxcod       = lif_pof_constants=>cs_trxcod-po_position
    evtcnt       = is_events-eventid
    evtid        = lif_pof_constants=>cs_milestone-po_confirmation
    evtdat       = sy-datum
    evttim       = sy-uzeit
    evtzon       = lcl_tools=>get_system_time_zone( )
  ) ).

  ct_eventid_map  = VALUE #( BASE ct_eventid_map (
    eventid      = is_events-eventid
    evtcnt       = is_events-eventid
  ) ).

  ct_tracklocation  = VALUE #( BASE ct_tracklocation (
    evtcnt       = is_events-eventid
    loccod       = lif_ef_constants=>cs_loc_types-plant
    locid1       = lcl_tools=>get_field_of_structure(
                     ir_struct_data = is_events-maintabref
                     iv_field_name  = 'WERKS' )
  ) ).

  " QUANTITY
  ct_trackparameters  = VALUE #( BASE ct_trackparameters (
    evtcnt       = is_events-eventid
    param_name   = lif_pof_constants=>cs_event_param-quantity
    param_value  = lcl_tools=>get_pretty_value( iv_value = lv_difference )
  ) ).

  " CONFIRMATION TYPE
  ct_trackparameters  = VALUE #( BASE ct_trackparameters (
    evtcnt       = is_events-eventid
    param_name   = lif_pof_constants=>cs_event_param-confirm_type
    param_value  = lif_pof_constants=>cs_relevance-ebtyp
  ) ).
ENDMETHOD.
```

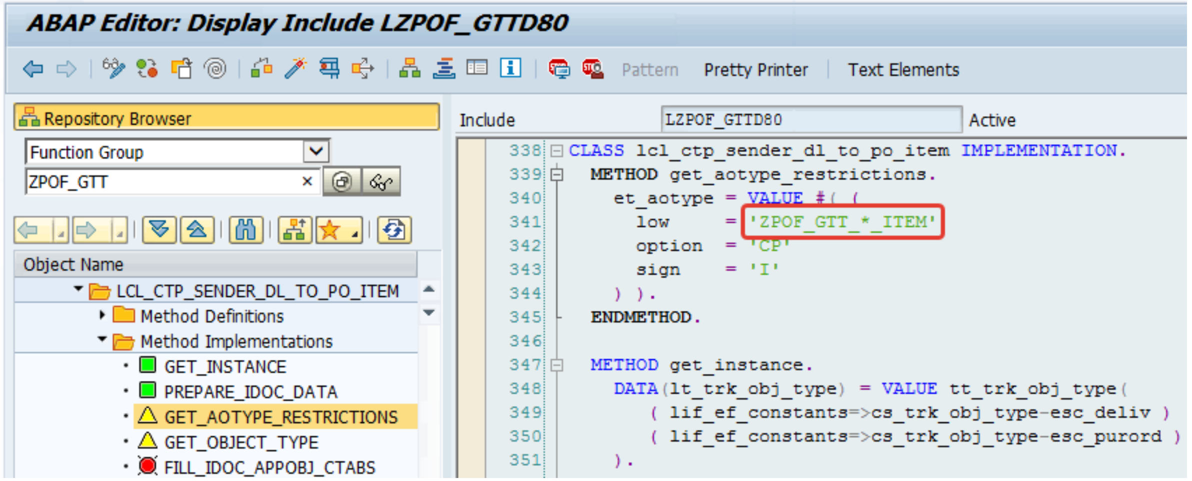
# 11: Enhancement codes for cross-processes tracking

The Tracking Purchasing Order application asks for cross-processes tracking, which is used in below cases:

1. When the inbound delivery item process is updated and transported to GTT, the preceding purchasing order item process needs to be updated and transported to GTT.
2. When the shipment process is updated and transported to GTT, the preceding inbound delivery and item process, and their planned events needs to be updated and transported to GTT.

IMPORTANT: To enable cross-processes tracking, please update the below sample codes after downloading:

1. Update Purchase Order Item AOT type Mask in Method GET_AOTYPE_RESTRICTIONS of LCL_CTP_SENDER_DL_TO_PO_ITEM
2. Update Inbound Delivery Header and Item AOT type Mask in Method GET_AOTYPE_RESTRICTIONS of LCL_CTP_SENDER_SH_TO_DL_HEAD and LCL_CTP_SENDER_SH_TO_DL_ITEM

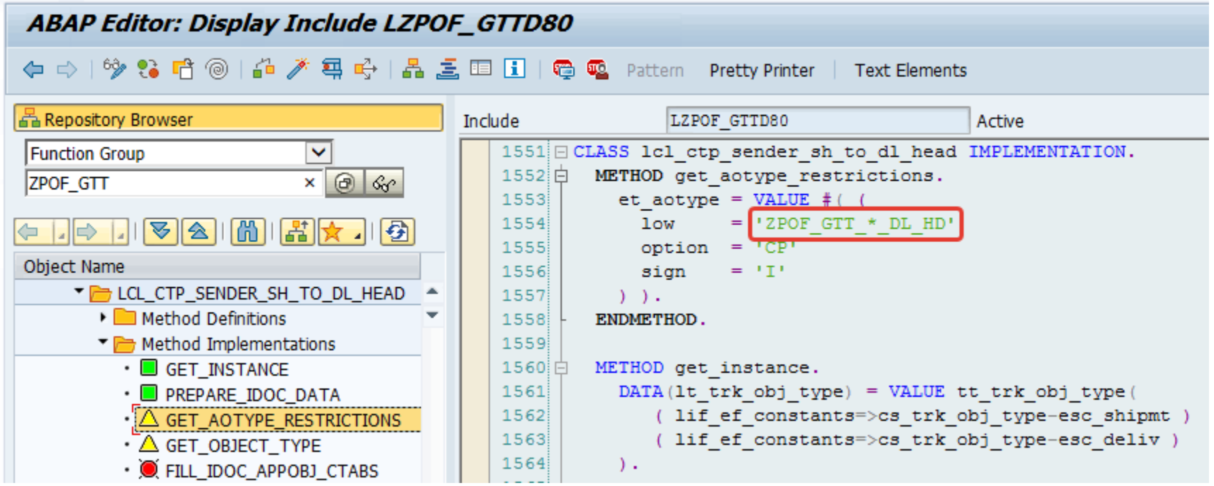


ABAP Editor: Display Include LZPOF_GTTD80

Function Group: ZPOF_GTT

Include: LZPOF_GTTD80 Active

```
CLASS lcl_ctp_sender_dl_to_po_item IMPLEMENTATION.
  METHOD get_aotype_restrictions.
    et_aotype = VALUE #( (
      low     = 'ZPOF_GTT_*_ITEM'
      option  = 'CP'
      sign    = 'I'
    ) ).
  ENDMETHOD.

  METHOD get_instance.
    DATA(lt_trk_obj_type) = VALUE tt_trk_obj_type(
       ( lif_ef_constants=>cs_trk_obj_type-esc_deliv )
       ( lif_ef_constants=>cs_trk_obj_type-esc_purord )
    ).
```



ABAP Editor: Display Include LZPOF_GTTD80

Function Group: ZPOF_GTT

Include: LZPOF_GTTD80 Active

```
CLASS lcl_ctp_sender_sh_to_dl_head IMPLEMENTATION.
  METHOD get_aotype_restrictions.
    et_aotype = VALUE #( (
      low     = 'ZPOF_GTT_*_DL_HD'
      option  = 'CP'
      sign    = 'I'
    ) ).
  ENDMETHOD.

  METHOD get_instance.
    DATA(lt_trk_obj_type) = VALUE tt_trk_obj_type(
       ( lif_ef_constants=>cs_trk_obj_type-esc_shipmt )
       ( lif_ef_constants=>cs_trk_obj_type-esc_deliv )
    ).
```

# 11: Enhancement codes for cross-processes tracking

The cross processes tracking scenarios cover below:

**Delivery Item -> Purchase Order Item:**

1\ Inbound Delivery Item Composition (Full Transport)
- Case: Inbound Delivery Item Create / Delete
- Case: Inbound Delivery Create / Delete

**Shipment -> Inbound Delivery and Inbound Delivery Item:**

1\ Tracking ID (Delta Transport)
- Case: Shipment Create / Delete with Delivery
- Case: Shipment Assign / Unassign Delivery

2\ Shipment Composition (Full Transport)
- Case: Shipment Create / Delete with Delivery
- Case: Shipment Assign / Unassign Delivery

3\ Planned Event in Delivery (Full Transport)
- Case: Shipment Create / Delete with Delivery / with stage
- Case: Shipment Assign / Unassign Delivery / with stage
- Case: Stage Assign / Unassign Delivery
- Case: Stage Insert / Delete
- Case: Stage Location Update
- Case: Stage Planned Datetime Update

4\ Planned Event in Delivery Item (Full Transport)
- Case: Shipment Create / Delete with Delivery / with stage
- Case: Shipment Assign / Unassign Delivery / with stage
- Case: Stage Assign / Unassign Delivery
- Case: Stage Insert / Delete
- Case: Stage Location Update
- Case: Stage Planned Datetime Update

# 12: Known Issues

## 1. Planned Event Extension not enabled

By now, on ERP side, the EXTENSION segment of process IDOC is not enabled for the planned event part, which means that user cannot make the user-defined fields based on the planned event level in Manage Models.

The workaround is to take use of Control Parameter’s segment in IDOC and make the field mapping on process level in Manage Models.

## 2. IDOC sequencing issue

By now, on ERP side, when the user is reporting actual events while creating the process, the IDOCs will be sent out of sequence. For example, entering a PICK quantity and saving the new delivery in ERP will generate a PICK event IDOC and a delivery order IDOC. If the event IDOC  approaches GTT prior to the order IDOC, which will lead into processing failure.

This issue will be covered in short future.

Follow us

www.sap.com/contactsap

THE BEST RUN SAP®