

Synchronize Actual Events back to SAP TM

To send event notifications for freight orders or freight bookings from the GTT system to your TM system, you need to set up a connection between your SAP TM or S/4HANA system and SAP Logistics Business Network, global track and trace option.

1. Configure web service in your TM system
2. Configure connection in your Cloud Connector
3. Define destination service in your SCP cockpit
4. Implement forwardEventToTM function in your GTT model

1. Configure Web Service in Your TM System

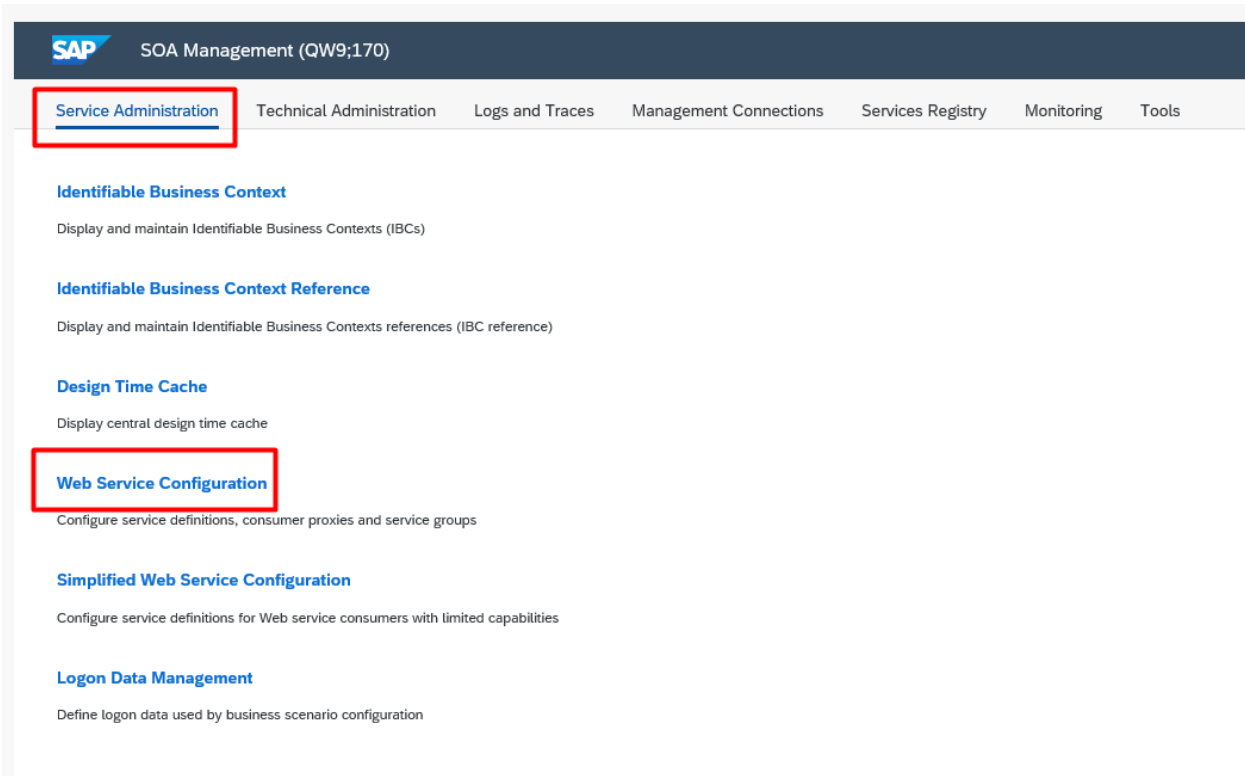
1.0 Check whether interface “TransportationEventBulkNotification_In” is available in your TM system.

To receive event notifications, you must use the B2B message interfaces: “TransportationEventBulkNotification_In” in your TM system.

For more information about the required message enhancement, see SAP Notes [2721729](#) and [2743015](#).

1.1 Execute T-CODE “SOAMANAGER” in your TM system.

Under “Service Administration”, select “Web Service Configuration”.




1.2 Search for “TransportationEventBulkNotification_In” and select the service “/SCMTMS/TPEVTBNTFI” to enter the detail page

[illegible]

1.3 In the “Configurations” Tab, click “Create Service”

[illegible]

1.4 In the popup window, fill in your own service name and security settings and save the service you created

 Web Service Configuration (QW9;170)

Configuration of New Binding for Service Definition '/SCMTMS/TPEVTBNTFI'

1

2

3

3

Service and Binding NameProvider SecuritySOAP ProtocolOperation Settings

Back

Next

Finish

Cancel

Service Information

Service Name:*ZGTT_Event_In

Service Description Text:

Binding Information

New Binding Name:*ZGTT_Event_In



Configuration of New Binding for Service Definition '/SCMTMS/TPEVTBNTFI'



[Back](#) [Next](#) [Finish](#) [Cancel](#)

Transport Guarantee

Transport Level: None

Transport Level Security

- ☒ None (http)
- ☐ SSL (https)

Message Level Security

- ☒ None
- ☐ Symmetric Message Signature and Encryption
- ☐ Asymmetric Message Signature
- ☐ Asymmetric Message Signature and Encryption
- ☐ Secure Conversation
- ☐ Extended Signature and Header Protection

Authentication Settings

Authentication Level: Basic

Authentication Method

- ☐ No Authentication

Transport Channel Authentication

- ☒ User ID/Password
- ☐ X.509 SSL Client Certificate
- ☐ Single Sign On using SAP Assertion Ticket
- ☐ Single Sign On using SPNegotiation

Message Authentication

- ☐ User ID/Password
- ☐ X.509 Certificate
- ☐ Single Sign On using SAML

SAP

Web Service Configuration (QW9;170)

Details of Service Definition: /SCMTMS/TPEVTBNTFI

OverviewConfigurationsClassificationsDetails

Define Services and Bindings

Create ServiceActivateDeactivateDeleteRepublishDisplay as List

Service/Binding	Actions
<input type="checkbox"/> LBN_EVENT_IN	
<input type="checkbox"/> LBN_EVENT_OUT	
<input checked="" type="checkbox"/> ZGTT_EVENT_IN	
<input type="checkbox"/> ZGTT_EVENT_OUT	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	
<input type="checkbox"/>	

SaveEdit

Provider SecurityTransport SettingsIdentifiable Business ContextOperation SettingsAdministrative Information

Transport Binding

Alternative Access URL:

Calculated Access URL:

Calculated Protocol:

HTTP

Make Local Call:

No Call in Local System

State Management Timeout:

0

2.0 Ensure that you have installed SAP Cloud Connector.

<https://help.sap.com/viewer/cca91383641e40ffbe03bdc78f00f681/Cloud/en-US/57ae3d62f63440f7952e57bfcef948d3.html>

2.1 After installing Cloud Connector and starting the Cloud Connector, log on to the cloud connector and perform required configurations to make your Cloud Connector operational.

For details, see <https://help.sap.com/viewer/cca91383641e40ffbe03bdc78f00f681/Cloud/en-US/db9170a7d97610148537d5a84bf79ba2.html>

2.2 After setting up the connection between your cloud platform and your cloud connector, then you need to maintain the connection between your TM system and your cloud connector.

Click “Cloud To On-Premise” and Create “Mapping Virtual to Internal System”


The screenshot shows the SAP Cloud Connector Administration interface. The left sidebar contains a menu with the following items: Connector, Security Status, Alerting, High Availability, Hardware Metrics Monitor, Configuration, and a sub-account section for 'lbn-gtt-samples'. Under the sub-account section, 'Cloud To On-Premise' is highlighted with a red box. The main area is titled 'Cloud To On-Premise' and has tabs for ACCESS CONTROL, COOKIE DOMAINS, APPLICATIONS, and PRINCIPAL PROPAGATION. The 'Mapping Virtual To Internal System' table is displayed with the following data:

Status	Virtual Host	Internal Host	Check Result	Protocol	Back-end Type	Actions
□			Reachable	HTTPS	ABAP System	[+]

A red box highlights the '+' icon in the Actions column. Below this table, there is a section for 'Resources Of tmqw9170:54300' with a table showing URL Path and Access Policy:

Status	URL Path	Access Policy	Actions
□		Path And All Sub-Paths	[+]

2.3 Fill in the virtual URL and internal URL of your TM system

 Virtual host and port cannot be edited

Back-end Type: ABAP System 

Protocol: HTTPS 

Virtual Host:

Virtual Port:


*Internal Host:

*Internal Port:

SAProuter:

Principal Type: None 

SNC Partner Name:

Host In Request Header: Use Virtual Host 

Description:

Check Internal Host: ☐

2.4 Add resource for your mapping internal URL. Fill in the URL path below with Calculated Access URL in the Step “Check Your Service’s Calculated Access URL”.

The image shows a screenshot of the SAP Cloud Connector web interface. At the top, there's a header with 'Subaccount: lbn-gtt-samples' and a plus icon. Below it, the main title 'Cloud To On-Premise' is displayed. A navigation bar contains four tabs: 'ACCESS CONTROL', 'COOKIE DOMAINS', 'APPLICATIONS', and 'PRINCIPAL PROPAGATION'. The 'ACCESS CONTROL' tab is active. Under this tab, the section 'Mapping Virtual To Internal System' is visible. Below this section, there are two tables. The first table has columns 'Status' and 'Virtual', with one row showing a green status icon and the value 'tmqw9'. The second table has columns 'Status' and 'URI', with one row showing a green status icon and the value '/sap'. A modal dialog box titled 'Edit Resource' is open in the center. It contains the following fields: '*URL Path:' with a text input field; 'Active:' with a checked checkbox; 'WebSocket:' with an unchecked checkbox; 'Access Policy:' with two radio button options, 'Path Only (Sub-Paths Are Excluded)' and 'Path And All Sub-Paths' (which is selected); and 'Description:' with a large text area. At the bottom of the dialog, there are 'Save' and 'Cancel' buttons. In the background, to the right of the dialog, another table is partially visible with columns 'Back-end Type' and 'Actions', showing a row for 'ABAP System'.

3. Define Destination Service in Your SCP Cockpit

3.1 Go to your subaccount that subscribed GTT in SCP Cockpit and create destination service.

The user and password here are your TM system user and password.

Fill in the URL as:

http://{virtual host in cloud connector}:{virtual port in cloud connector}{Calculated Access URL}

Note down your destination service name.

Destination Configuration

Name: *

TO_QW9_170_STANDARD

Type:

HTTP

Description:

ABAP QW9

URL: *

Proxy Type:

OnPremise

Authentication:

BasicAuthentication

Location ID:

cpi

User: *

Password:

Additional Properties

4. Implement forwardEventToTM Function in Your GTT Model

4.1 In the corresponding model's Event-to-Action script in the Manage Models app, you can refer to the script in the template model below to implement forwardEventToTM function.

Note:

If you only deploy the Track Shipments application, the part starting from “==> sync back event to TM system” in the Event to Action script is enough. The rest of the script is used for Track Sales Orders application. Replace “TO_QW9_170_STANDARD” with the destination name that you maintained in Step 3 and replace “QW9CLNT170” with the logical system of shipment tracked process.

For details, see help.sap.com/gtt and search for *Guide for Model Administrators*.

Help

```
341 ?"==> forwardCurrentTP to gtt_sof_forwarding-act";  
342 forwardCurrentTP("gtt_sof_forwarding-act", false, false, true);  
343 }  
344  
345 ?"==> sync back event to TM system";  
346 ?"==> begin forwardEventToTM";  
347  
348 var eventTypeOfActualEvent = actualEvent.eventType;  
349  
350 if ("com.lbngttsamples.gtt.app.sof.Shipment.Departure" == eventTypeOfActualEvent ||  
351     "com.lbngttsamples.gtt.app.sof.Shipment.Arrival" == eventTypeOfActualEvent ||  
352     "com.lbngttsamples.gtt.app.sof.Shipment.Delay" == eventTypeOfActualEvent ||  
353     "com.lbngttsamples.gtt.app.sof.Shipment.LoadingStart" == eventTypeOfActualEvent ||  
354     "com.lbngttsamples.gtt.app.sof.Shipment.LoadingEnd" == eventTypeOfActualEvent ||  
355     "com.lbngttsamples.gtt.app.sof.Shipment.POD" == eventTypeOfActualEvent ||  
356     "com.lbngttsamples.gtt.app.sof.Shipment.POPU" == eventTypeOfActualEvent ||  
357     "com.lbngttsamples.gtt.app.sof.Shipment.Coupling" == eventTypeOfActualEvent ||  
358     "com.lbngttsamples.gtt.app.sof.Shipment.Decoupling" == eventTypeOfActualEvent ||  
359     "com.lbngttsamples.gtt.app.sof.Shipment.UnloadingStart" == eventTypeOfActualEvent ||  
360     "com.lbngttsamples.gtt.app.sof.Shipment.UnloadingEnd" == eventTypeOfActualEvent) {  
361  
362     var destinationName = "TO_QW9_170_STANDARD";  
363     var reportedBy = null;  
364     var senderPartyId = actualEvent.senderPartyId;  
365     var trackingIdType = actualEvent.trackingIdType;  
366     var partyId = trackedProcess.partyId;  
367     var logicalSystemOfTp = trackedProcess.logicalSystem;  
368  
369     if ("QW9CLNT170" == logicalSystemOfTp && "SHIPMENT_ORDER" == trackingIdType) {  
370         if (actualEvent.containsProperty("reportedBy")) {  
371             reportedBy = actualEvent.reportedBy;  
372         }  
373         if (senderPartyId != partyId || reportedBy != null) {  
374             forwardEventToTM(toObject("{}"), destinationName);  
375         }  
376     }  
}
```