

Implementation Guide

– *Track Sales Orders* Template App

Contents

0.Prerequisites	2
1.Environment Configuration in Logistics Business Network, Global Track and Trace Option	2
1.1 Enable Cloud Foundry and Create Space	2
1.2 Maintain Roles and Users	4
1.3 Maintain Technical User	5
1.4 Maintain Destination of GTT Core Engine Base URL.....	8
2.Implement <i>Track Sales Orders</i> Template App	9
2.1 Download Template Code.....	9
2.2 Download and Setup Tools	9
2.3 Deploy Model.....	10
2.4 Package and Deploy.....	12
2.4.1 Login to SAP Cloud Foundry.....	12
2.4.2 Package and Deploy <i>Track Sales Orders</i> Frontend App	12
2.4.3 Package and Deploy SAP Fiori Launchpad for <i>Template Applications</i>	13
2.4.4 Package and Deploy <i>Track Sales Orders</i> Backend Service	20
2.4.5 Check Deployed Apps on SAP Cloud Foundry Cockpit.....	21
2.4.6 Maintain Destination of Forwarding.....	21
2.4.7 Maintain Roles and Users	24
2.4.8 Open <i>Track Sales Orders</i> Template App from SAP Fiori Launchpad	24
3. Implement ERP Extractor	25
4. Other Configuration	25
4.1 Obtain Here Map License.....	25
4.2 Maintain Location Master	26
4.3 Maintain Data Contributors	27
4.4 Integrate with Visibility Provider	27

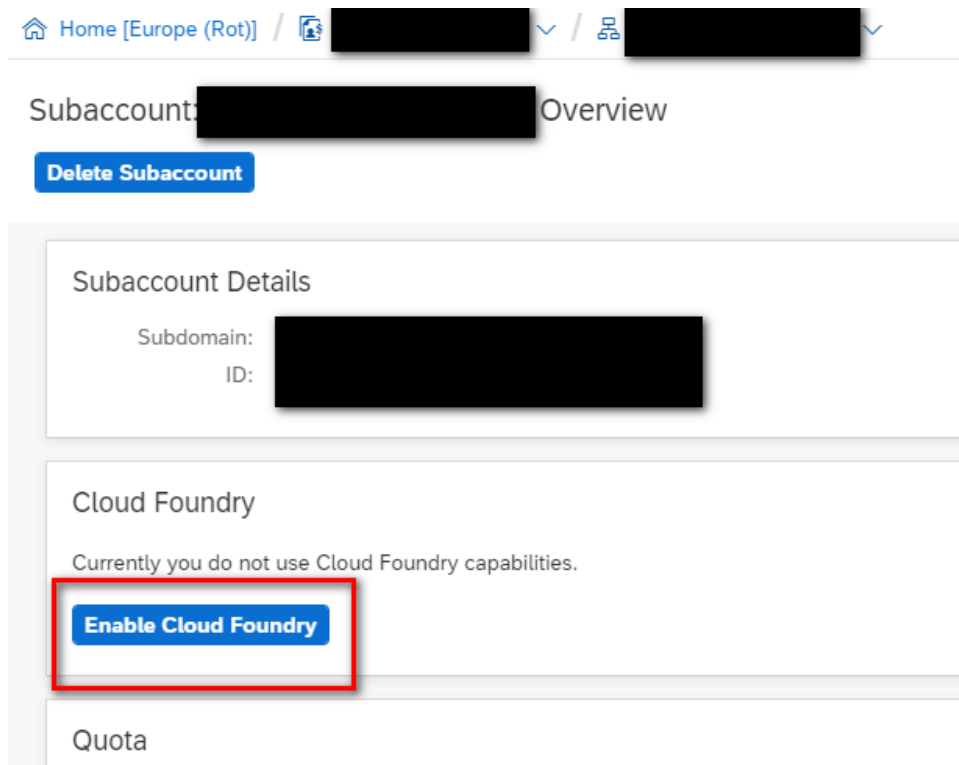
0. Prerequisites

- You have an SAP Cloud Platform global account with the entitlement to the global track and trace option for SAP Logistics Business Network, and the account is assigned 1 portal service quota and 2 GB Application Runtime quota.
You can refer to the 1st Q&A in “*FAQs for Template Code Implementation*” for guidance on how to add quotas.

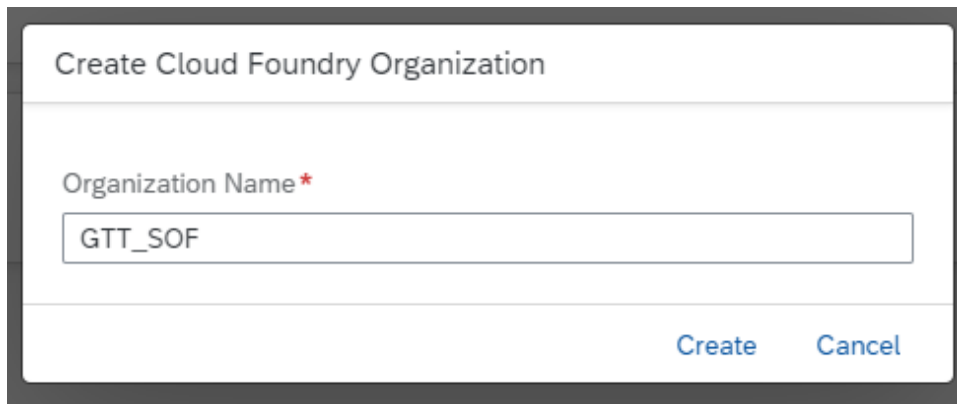
1. Environment Configuration in Logistics Business Network, Global Track and Trace Option

1.1 Enable Cloud Foundry and Create Space

- Click “Enable Cloud Foundry” to enable the development landscape.

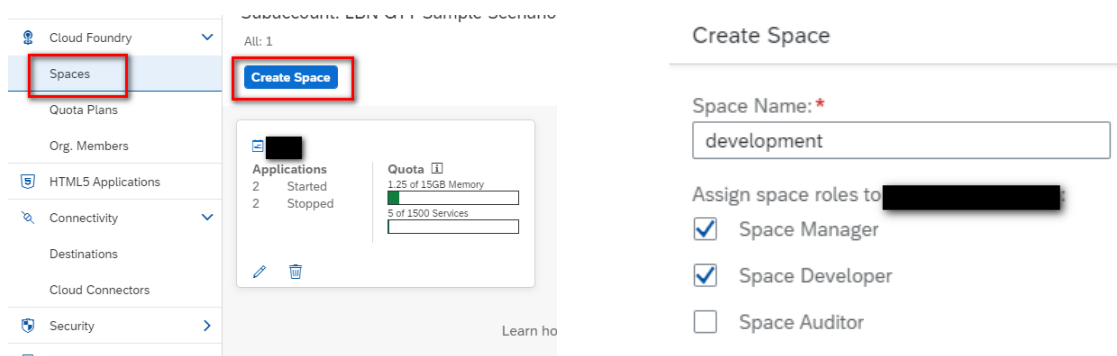


- b. Fill in the Organization Name and click “Create”



A dialog box titled "Create Cloud Foundry Organization". It contains a text input field for "Organization Name" with a red asterisk indicating it is required. The field contains the text "GTT_SOF". At the bottom right, there are two buttons: "Create" and "Cancel".

- c. Go to “Cloud Foundry” -> “Spaces”
Click “Create Space”
Fill in the Space Name and click “Create”



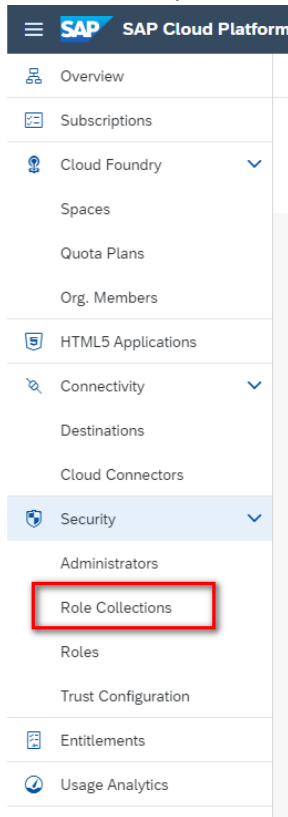
The image shows two parts of the Cloud Foundry interface. On the left is the "Spaces" overview page. The "Spaces" menu item in the left sidebar is highlighted with a red box. The "Create Space" button at the top of the main content area is also highlighted with a red box. Below this button is a table showing the state of spaces. On the right is the "Create Space" dialog box.

Applications		Quota
2	Started	1.25 of 15GB Memory
2	Stopped	5 of 1500 Services

The "Create Space" dialog box has a "Space Name" field with a red asterisk, containing the text "development". Below this is a section "Assign space roles to" followed by a list of roles: "Space Manager" (checked), "Space Developer" (checked), and "Space Auditor" (unchecked). The role names are partially obscured by a black redaction box.

1.2 Maintain Roles and Users

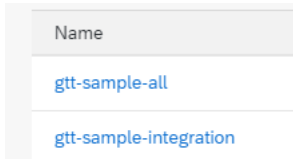
- a. Go to “Security” -> “Role Collections”



- b. Create 2 role collections (You can have user-defined names):

gtt-sample-all

gtt-sample-integration



- Role template “gtt-sample-all” is granted to users who have access to global track and trace option and *Track Sales Orders* template app.
- Role template “gtt-sample-integration” is granted to technical users only who have access to the global track and trace APIs for development purpose.

- c. Go to the role of “gtt-sample-all”, and add the following roles:

Role	Application
MessageLogAdminTemplate	lbn_gtt_core
DataModelAdminTemplate	lbn_gtt_core
LocationProcessTemplate	lbnplatform-loc-broker
BPEditTemplate	lbnplatform-bprepo
BusinessPartnerDiscoveryTemplate	lbnplatform-bprepo

InviteBPPProcessTemplate	lbnplatform-bprepo
ManageInvitationsProcessTemplate	lbnplatform-bprepo
BlockStoreProcessTemplate	lbn-retentionmanagement-sb
ReminderTemplateProcessTemplate	lbn-retentionmanagement-sb

d. Grant "gtt-sample-all" to your **application** users:

- Application users are those who have access to global track and trace option and *Track Sales Orders* template app.

IT Administrator needs to maintain application users in identity provider.

For example, user1@sap.com



The screenshot shows a table titled 'Users (51)' with a search bar. The table has five columns: ID, Identity Provider, E-Mail, First Name, and Last Name. The first three rows of data are visible, all showing 'SAP ID Service' as the identity provider and email addresses ending in '@sap.com'. The first names and last names are partially obscured by black boxes.

ID	Identity Provider	E-Mail	First Name	Last Name
[redacted]	SAP ID Service	[redacted]@sap.com	[redacted]	[redacted]
[redacted]	SAP ID Service	[redacted]@sap.com	[redacted]	[redacted]
[redacted]	SAP ID Service	[redacted]@sap.com	[redacted]	[redacted]

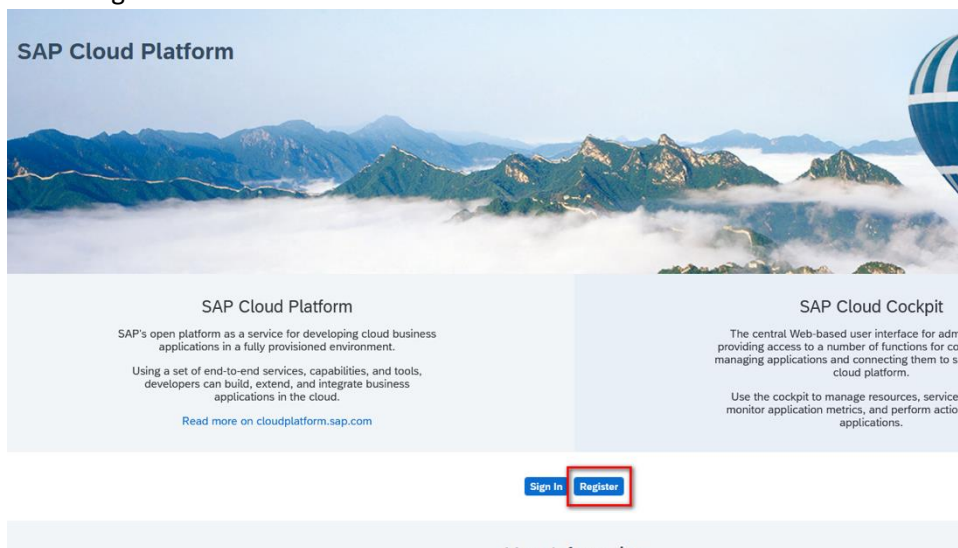
Note: The application user in this step should be different from the technical user created in Step 1.3. A user should **never** be assigned to **both** "gtt-sample-all" and "gtt-sample-integration".

1.3 Maintain Technical User

a. Go to the SAP Cloud Platform to create the technical user account:

<https://account.hana.ondemand.com/#/home/welcome>

Click "Register"



- b. Fill in the mandatory fields:

Tell us about yourself

E-mail address *

First name *

Last name *

Company *

Country/Region
United States

Phone
+1 XXX-XXX-XXXX

Department *

Relationship to SAP *

Create password *

Confirm password *

Must have at least 8 characters.


☐ I have read and understood the Terms and Conditions of [SAP.com](#)

SAP will use any of the data provided hereunder in accordance with the [Privacy Statement](#).

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

[Submit](#)

- c. Go to the mail inbox to activate the account

 **CLOUD IDENTITY**

Hello [redacted]

An account has been created for you. To activate your account for [Administration Console](#) of SAP Cloud Identity, click the link below. You will be taken to a page where you will also set a password for your account.

[Click here to activate](#)

If the link above is not displayed or does not work, copy and paste the link below to the address bar of your browser.

[redacted]

For more information, see [SAP Cloud Identity service documentation](#)

In case of any issues, create a ticket with BC-IAM-IDS component in [SAP Support Portal](#).

- d. As shown in Step 1.2, go to the role of “gtt-sample-integration”, and add the following roles:

Role	Application
EMIDocIntegrationTemplate	lbn_gtt_core
MetadataServiceIntegrationTemplate	lbn_gtt_core
ReadServiceIntegrationTemplate	lbn_gtt_core
WriteServiceIntegrationTemplate	lbn_gtt_core
LocationProcessTemplate	lbnplatform-loc-broker

- e. Grant "gtt-sample-integration" to your **technical** user:

For example, techuser1@techcompany.com

Users (2)

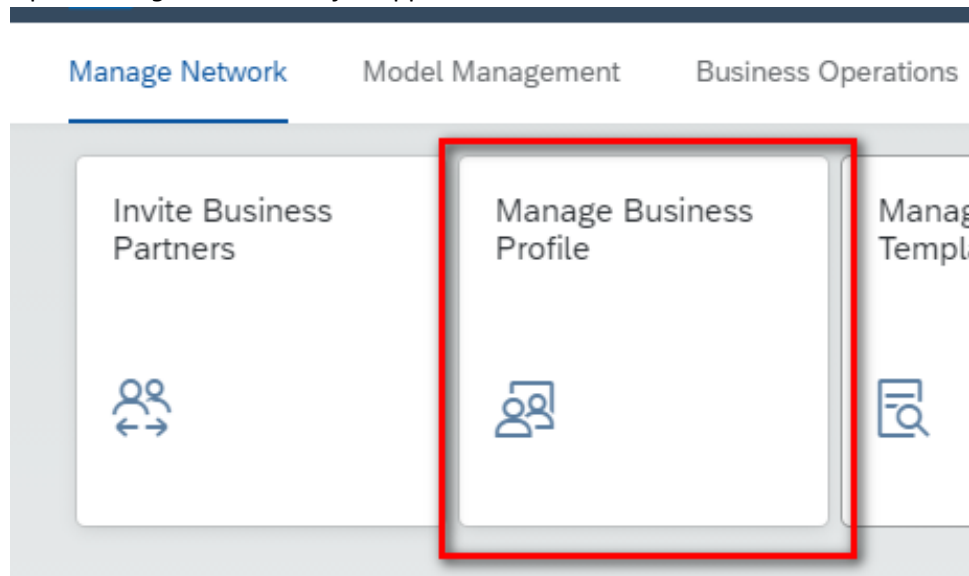
ID	Identity Provider	E-Mail
<input type="text" value="techuser1@techcompany.com"/>	<input type="text" value="Default identity provider"/>	<input type="text" value="techuser1@techcompany.com"/>

Note:

In the "Identity Provider" field, select "Default identity provider".

- f. Go to SAP Fiori Launchpad for global track and trace option

Open *Manage Business Profile* app



- g. Add one Alternate Identifier:

Identifier Type: Technical User

ID: the email address of technical user

Email	Phone Number
<input type="text" value=""/>	<input type="text" value=""/>

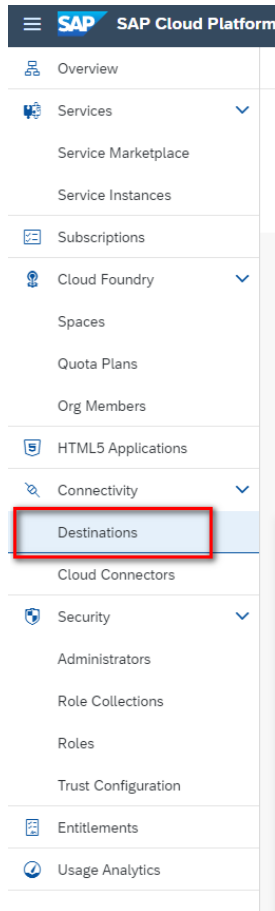
Alternate Identifiers (1)

Identifier Type	ID
<input type="text" value="Technical User"/>	<input type="text" value="...@...com"/>

Note: The technical user in this step should be different from the application user in Step 1.2.4. A user should **never** be assigned to **both** "gtt-sample-all" and "gtt-sample-integration".

1.4 Maintain Destination of GTT Core Engine Base URL

a. Go to “Destinations”

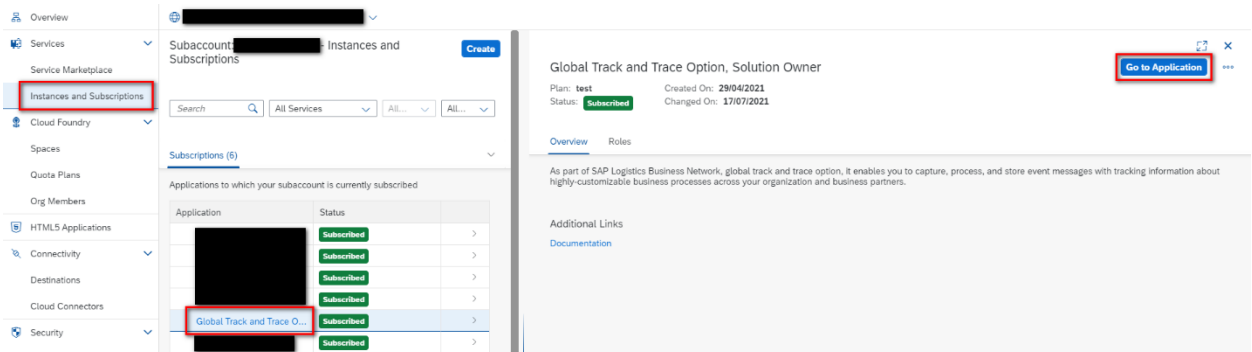


b. Configure Destination of GTT Core Engine Base URL

Name	gtt_core_engine_flp_base_url
Type	HTTP
Description	<optional>
URL	<p>Enter the GTT Fiori Launchpad URL and add “/api” at the end of it. For example: https://xxxxx.gtt-flp-lbnplatform.cfapps.eu10.hana.ondemand.com/api</p> <p>For instructions on how to get the GTT Fiori Launchpad URL, see the note under this table.</p>
Proxy Type	Internet
Authentication	BasicAuthentication
User	Email of Technical User from Step 1.3
Password	Password of Technical User from Step 1.3

Note:

- On your subaccount page, click “Instances and Subscriptions” in the “Service tab” and click “Global Track and Trace Option” application. Click “Go to Application”.



- Copy the GTT Fiori Launchpad URL as shown in the picture:

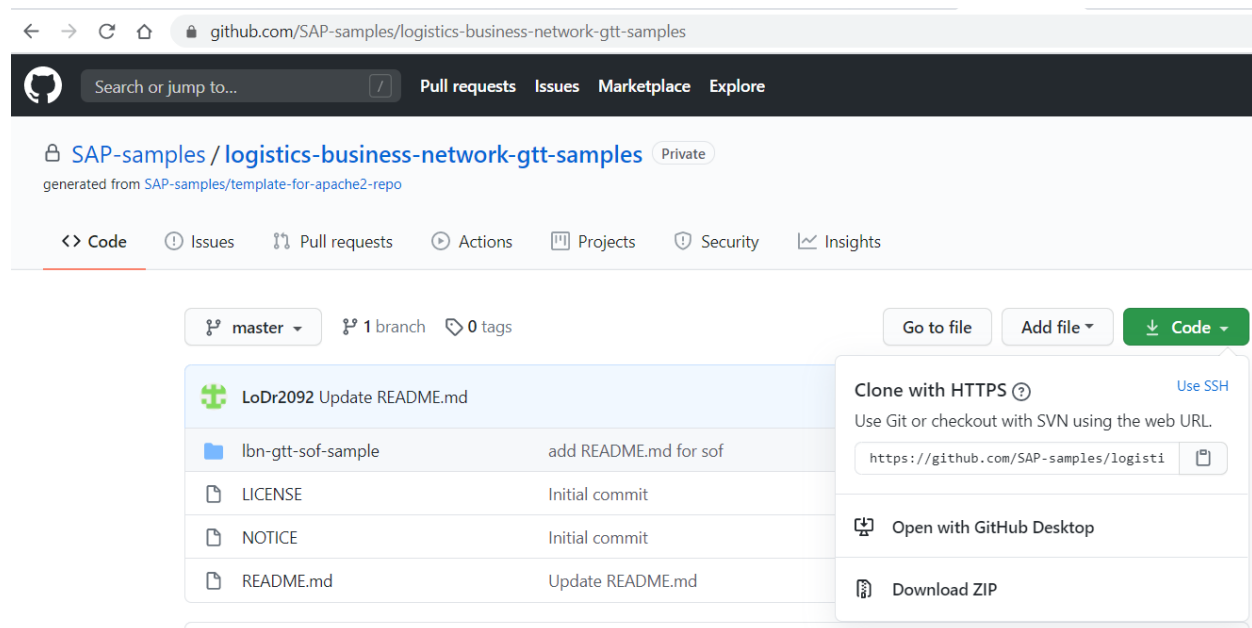
[https://\[redacted\]gtt-flp-lbnplatform.cfapps.eu10.hana.ondemand.com/cp.portal/site#Shell-home](https://[redacted]gtt-flp-lbnplatform.cfapps.eu10.hana.ondemand.com/cp.portal/site#Shell-home)

2. Implement *Track Sales Orders* Template App

2.1 Download Template Code

Go to <https://github.com/SAP-samples/logistics-business-network-gtt-samples>.

Download and extract, or git clone the template code from SAP-samples on GitHub.



2.2 Download and Setup Tools

- Install JDK 8 and Maven

JDK 8: <https://www.oracle.com/java/technologies/javase/javase-jdk8-downloads.html>

Maven: <https://maven.apache.org/download.cgi>

- b. Install NPM, Node JS 12, and grunt-cli
[Node JS 12: https://nodejs.org/dist/latest-v12.x/](https://nodejs.org/dist/latest-v12.x/)

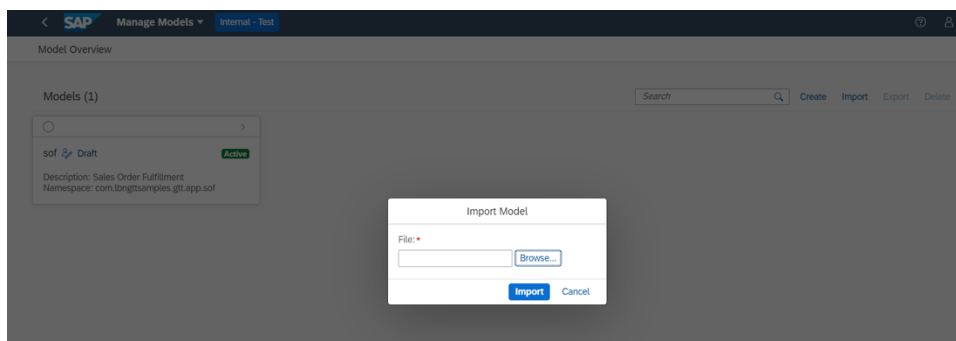
For the instructions on how to install Node JS and NPM, see <https://www.saplearners.com/how-to-install-node-js-and-npm/>

Run the following command after NPM is installed:
`npm config set @sap:registry "https://npm.sap.com" -g`

- c. Install Cloud Foundry Command Line Interface
<https://developers.sap.com/tutorials/cf-download-cli.html>
- d. Install MultiApps CF CLI Plugin
<https://github.com/cloudfoundry-incubator/multiapps-cli-plugin>
- e. Install Cloud MTA Build Tool (MBT)
<https://sap.github.io/cloud-mta-build-tool/download/>
- f. Install MAKE
<https://sap.github.io/cloud-mta-build-tool/makefile/>

2.3 Deploy Model

- Go to the folder *model* with downloaded source code. Make sure you have a JSON file *sof-model-YYYYMMDDxx.json* (e.g. *sof-model-2020091001.json*).
- Open *Manage Models* app on SAP Fiori Launchpad for global track and trace option
- Click “Import”
- Click “Browse” to select the *sof-model-YYYYMMDDxx.json* file
- Click “Import” to import the model



- Click “Deploy”
- Write down the model namespace for later use, in this case, “com.lbngttsamples.gtt.app.sof” in the picture below

SAP Model Details Internal - Test

sof Draft Active

Sales Order Fulfillment

Namespace: com.lbngttsamples.gtt.app.sof Correlation Level: 5

Tracked Process Field Type Pool Event Type Pool Code List IDOC Integration Visibility Provider Integration Planned Event Extension Event to Action

Items (6) Create Edit Delete

SalesOrder
Description: Sales Order
Tracking Id Type: SALES_ORDER

SalesOrderItem
Description: Sales Order Item
Tracking Id Type: SALES_ORDER_ITEM

Delivery
Description: Outbound Delivery
Tracking Id Type: OUTBOUND_DELIVERY

DeliveryItem
Description: Outbound Delivery Item
Tracking Id Type: OUTBOUND_DELIVERY_IT

Content: Field Lists

User Model Fields (16)

	Name	Type	DPP	Grant	Readable	Writable	
<input type="checkbox"/>	Sales Order Number	String(10)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	salesOrderNo	String(10)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	Ship-To Party	String(10)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	shipToPartyId	String(10)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	Net Value	Decimal(15,2)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	netValue	Decimal(15,2)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	Currency	String(5)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	currency	String(5)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	Purchase Order Number	String(35)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	customerReference	String(35)			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	Document Creation Date	Date			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	documentDate	Date			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	>
<input type="checkbox"/>	Incoterms						>

- Click "Edit"
- Go to *Event to Action* tab
- Replace all the namespace strings, in this case, "com.lbngttsamples.gtt.app.sof" in *Event to Action* script with the model namespace you wrote down in the previous step
- Click "Save and Deploy" to deploy the model

SAP Model Details Internal - Test

sof Active

Sales Order Fulfillment

Namespace: com.lbngttsamples.gtt.app.sof Correlation Level: 5

Tracked Process Field Type Pool Event Type Pool Code List IDOC Integration Visibility Provider Integration Planned Event Extension **Event to Action**

Help

```

11 //actualEvent.actualBusinessTimezone;
12 //?=> trackedProcess;
13 //?trackedProcess;
14
15 var EXECUTION_STATUS_EVENT_FOR_DELIVERY_ITEM = "com.lbngttsamples.gtt.app.sof.DeliveryItem.DeliveryItemPOD";
16 var EXECUTION_STATUS_EVENT_FOR_SHIPMENT = "com.lbngttsamples.gtt.app.sof.Shipment.POD";
17
18 var obj = toObject("{}");
19 var needForward = false;
20 var emailEnabled = false;
21
22 ?=> eventType;
23 ?actualEvent.eventType;
24 ?=> trackedProcess;
25 ?trackedProcess;
26
27 if (emailEnabled && "com.lbngttsamples.gtt.app.sof.SalesOrder.SalesOrderEvent" == actualEvent.eventType) {
28   var userPlaceholders0 = toObject("{}");
29   var appPlaceholders0 = toObject("{}");
30   var directMailTemp0 = reminderService.generateDirectMailTemplate();
31   var salesOrderGUID0 = trackedProcess.id;
32
33   userPlaceholders0.recipients1 = "johnny.zong@sap.com";
34   userPlaceholders0.salesOrderNo = trackedProcess.salesOrderNo;
35   userPlaceholders0.salesOrderGUID = salesOrderGUID0;
36
37   appPlaceholders0.solutionOwner_BpName = "This is BP name";
38   appPlaceholders0.App_Placeholder_1 = "This is APP Place holder 1";
39
40   directMailTemp0.templateId = "invite";
41   directMailTemp0.recipients = ["jinnming.wang@sap.com", "dexian.li@sap.com", "joe.xia@sap.com"];

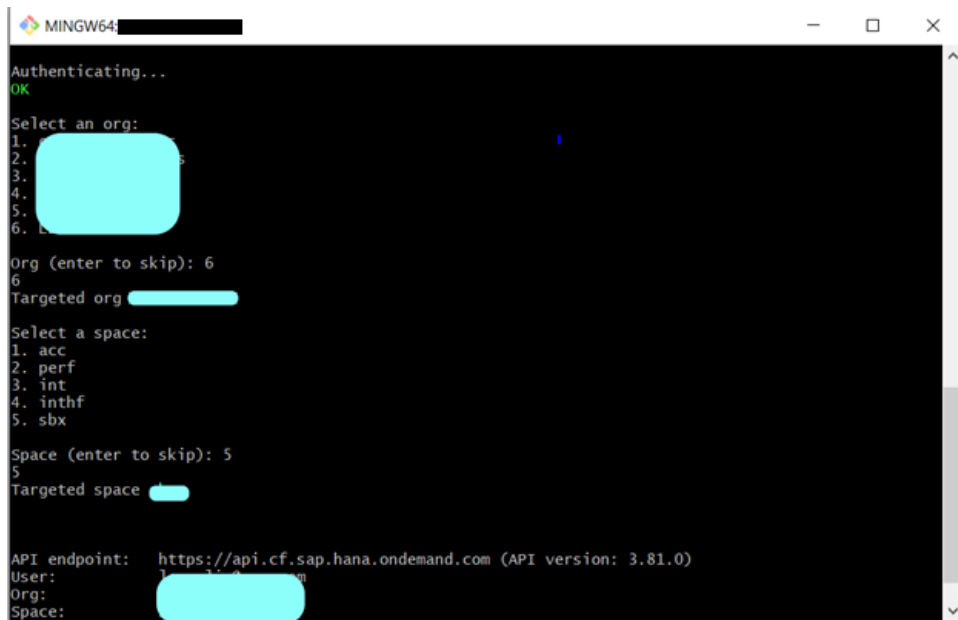
```

Save Save and Deploy Cancel

2.4 Package and Deploy

2.4.1 Login to SAP Cloud Foundry

- Open a command window, *cf login* to SAP Cloud Foundry with your username and password.
- Select the organization and space in which you want to deploy the template apps.

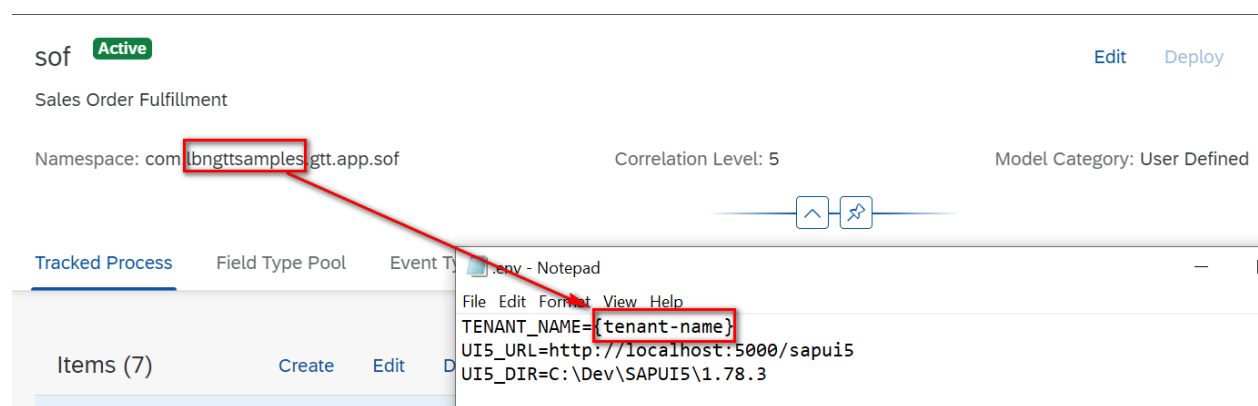


```
MINGW64: [redacted]
Authenticating...
OK
Select an org:
1. [redacted]
2. [redacted]
3. [redacted]
4. [redacted]
5. [redacted]
6. [redacted]
Org (enter to skip): 6
Targeted org [redacted]
Select a space:
1. acc
2. perf
3. int
4. inthf
5. sbx
Space (enter to skip): 5
Targeted space [redacted]
API endpoint: https://api.cf.sap.hana.ondemand.com (API version: 3.81.0)
User: [redacted]
Org: [redacted]
Space: [redacted]
```

2.4.2 Package and Deploy *Track Sales Orders* Frontend App

Go to GTT-V2-Sample-TrackSalesOrders-UI directory.

- Open the `./UI/.env` file
- Replace `{tenant-name}` with the middle part of the namespace you got from Step 2.3.

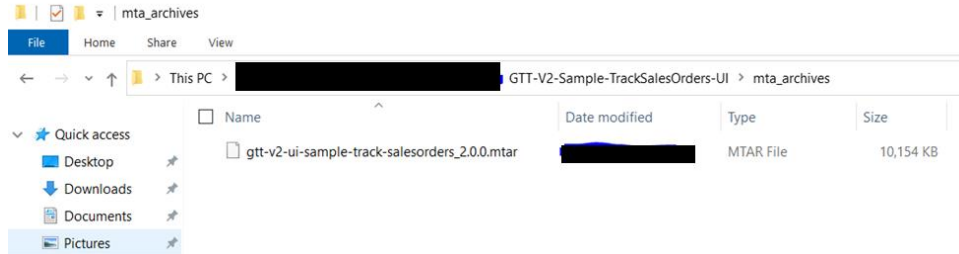


In the command line tool,

- Run the following command to build the MTAR package:

mbt build

A MTAR file `./mta_archives/gtt-v2-ui-sample-track-salesorders_xxx.mtar` is then generated.



- To deploy the MTAR package in Cloud Foundry, run the following command:

```
cf deploy ./mta_archives/gtt-v2-ui-sample-track-salesorders-service_xxx.mta
```

The deployment time depends on your network conditions.

2.4.3 Package and Deploy SAP Fiori Launchpad for *Template Applications*

Go to GTT-V2-Sample-App-FLP directory.

- First ensure that at least 1 portal quota is assigned to the subaccount in which apps are to be deployed.
Please refer to the 1st Q&A in *FAQs for Template Code Implementation* for guidance on how to assign quotas.
 - Open the `mta.yaml` file
 - Replace {unique-identifier} with unique information such as your company name or cloud foundry global account name
- Note:
- {unique-identifier} must only contain lowercase letters, numbers and hyphens. For example, "company-s01".
 - For the length of "route", the maximum limit is 63 characters.

```

1  ---
2  ID: lbn-gtt-sample-app-flp
3  _schema-version: "3.1"
4  parameters:
5    deploy_mode: html5-repo
6    enable-parallel-deployments: true
7    append: "${space}"
8    xsappname: tt_sof_${append}
9  version: 2.3.0
10 modules:
11   - name: lbn-gtt-sample-app-appRouter
12     type: approuter.nodejs
13     path: flp_appRouter
14     parameters:
15       disk-quota: 256M
16       memory: 256M
17       routes:
18         - route: lbn-gtt-samples-app-flp-{unique-identifier} ${domain}
19     properties:
20       SAP_JWT_TRUST_ACL:
21         - clientid: "*"
22         identityzone: "*"
23     destinations:
24       - name: lbn-gtt-sof-api

resources:
  - name: lbn-gtt-sample-app-runtime
    parameters:
      service-plan: app-runtime
      service: html5-apps-repo
    type: org.cloudfoundry.managed-service
  - name: lbn-gtt-sample-portal-resources
    parameters:
      service-plan: standard
      service: portal
    type: org.cloudfoundry.managed-service
  - name: lbn-gtt-sof-api
    properties:
      url: https://gtt-sample-track-salesorders-service-{unique-identifier}.${default-domain}
  - name: lbn-gtt-sst-api
    properties:
      url: https://gtt-sample-track-shipments-service-{unique-identifier}.${default-domain}
  - name: lbn-gtt-pof-api
    properties:
      url: https://gtt-sample-track-purchaseorders-service-{unique-identifier}.${default-domain}
  - name: lbn-gtt-sample-app-uaa
    type: com.sap.xs.uaa-application
    parameters:

```

This FLP project includes 3 template applications which are Track SO Fulfillment, Track Shipments and Track Purchase Orders. Before you build and deploy it, please deploy all the UI projects you need in advance, otherwise the deployment will fail.

If you only need Track SO Fulfillment app, do the following changes:

- Open the *mta.yaml* file
- Delete *lbn-gtt-sst-api* and *lbn-gtt-pof-api* in *lbn-gtt-sample-app-appRouter* and *resources* modules
- Delete *gtt-ui-sample-track-shipments-app-host* and *gtt-ui-sample-track-purchaseorders-app-host* in *flp* and *resources* modules

```
1 ---
2 ID: lbn-gtt-sample-app-flp
3 _schema-version: "3.1"
4 parameters:
5   deploy_mode: html5-repo
6   enable-parallel-deployments: true
7   append: "${space}"
8   xsappname: tt_sof_${append}
9   version: 2.3.0
10 modules:
11   - name: lbn-gtt-sample-app-appRouter
12     type: approuter.nodejs
13     path: flp_appRouter
14     parameters:
15       disk-quota: 256M
16       memory: 256M
17       routes:
18         - route: lbn-gtt-samples-app-flp-{unique-identifier}.${domain}
19     properties:
20       SAP_JWT_TRUST_ACL:
21         - clientid: "*"
22         identityzone: "*"
23     destinations:
24       - name: lbn-gtt-sof-api
25         url: "{lbn-gtt-sof-api/url}"
26         forwardAuthToken: true
27       - name: lbn-gtt-sst-api
28         url: "{lbn-gtt-sst-api/url}"
29         forwardAuthToken: true
30       - name: lbn-gtt-pof-api
31         url: "{lbn-gtt-pof-api/url}"
32         forwardAuthToken: true
33     requires:
34       - name: lbn-gtt-sample-app-usa
35       - name: lbn-gtt-sample-app-runtime
36       - name: lbn-gtt-sample-portal-resources
37       - name: lbn-gtt-sof-api
38       - name: lbn-gtt-sst-api
39       - name: lbn-gtt-pof-api
40   - name: flp
41     type: com.sap.portal.content
42     path: flp
43     parameters:
44       stack: cflinuxfs3
45       memory: 128M
46       buildpack: https://github.com/cloudfoundry/nodejs-buildpack/releases/download/v1.6.39/nodejs-buildpack-cflinuxfs3-v1.6.39.zip
47     requires:
48       - name: lbn-gtt-sample-app-usa
49       - name: lbn-gtt-sample-portal-resources
50       - name: gtt-ui-sample-track-salesorders-app-host
51       - name: gtt-ui-sample-track-shipments-app-host
52       - name: gtt-ui-sample-track-purchaseorders-app-host
```

```

53 resources:
54   - name: lbn-gtt-sample-app-runtime
55     parameters:
56       service-plan: app-runtime
57       service: html5-apps-repo
58     type: org.cloudfoundry.managed-service
59   - name: lbn-gtt-sample-portal-resources
60     parameters:
61       service-plan: standard
62       service: portal
63     type: org.cloudfoundry.managed-service
64   - name: lbn-gtt-sof-api
65     properties:
66       url: https://gtt-sample-track-salesorders-service-{unique-identifier}.{default-domain}
67   - name: lbn-gtt-sst-api
68     properties:
69       url: https://gtt-sample-track-shippments-service-{unique-identifier}.{default-domain}
70   - name: lbn-gtt-pof-api
71     properties:
72       url: https://gtt-sample-track-purchaseorders-service-{unique-identifier}.{default-domain}
73   - name: lbn-gtt-sample-app-uaa
74     type: com.sap.xs.uaa-application
75     parameters:
76       config:
77         xsappname: "${xsappname}"
78         tenant-mode: dedicated
79         oauth2-configuration:
80           system-attributes:
81             - rolecollections
82         description: Security profile of sales order fulfillment application
83         scopes:
84           - name: "$XSAPPNAME.sof.r"
85             description: Allow to access sales order fulfillment application
86           - name: "$XSAPPNAME.pof.r"
87             description: Allow to access purchase order fulfillment application
88           - name: "$XSAPPNAME.sst.r"
89             description: Allow to access standalone shipment tracking application
90           - name: "$XSAPPNAME.sst.w"
91             description: Allow to report events in standalone shipment tracking application
92           - name: uaa.user
93             description: UAA
94         authorities:
95           - "$XSAPPNAME.sof.r"
96           - "$XSAPPNAME.pof.r"
97           - "$XSAPPNAME.sst.r"
98           - "$XSAPPNAME.sst.w"
99           - uaa.user
100         role-templates:
101           - name: SalesOrderFulfillmentAppUserDisplayTemplate
102             description: Track & Trace Sales Order Fulfillment Template App User Read Access
103             scope-references:
104               - "$XSAPPNAME.sof.r"
105               - UAA.UAAP

```



```

88     - name: "$XSAPPNAME.sst.r"
89       description: Allow to access standalone shipment tracking application
90     - name: "$XSAPPNAME.sst.w"
91       description: Allow to report events in standalone shipment tracking application
92     - name: uaa.user
93       description: UAA
94   authorities:
95     - "$XSAPPNAME.sof.r"
96     - "$XSAPPNAME.pof.r"
97     - "$XSAPPNAME.sst.r"
98     - "$XSAPPNAME.sst.w"
99     - uaa.user
100   role-templates:
101     - name: SalesOrderFulfillmentAppUserDisplayTemplate
102       description: Track & Trace Sales Order Fulfillment Template App User Read Access
103       scope-references:
104         - "$XSAPPNAME.sof.r"
105         - uaa.user
106     - name: PurchaseOrderFulfillmentAppUserDisplayTemplate
107       description: Track & Trace Purchase Order Fulfillment Template App User Read Access
108       scope-references:
109         - "$XSAPPNAME.pof.r"
110         - uaa.user
111     - name: StandaloneShipmentTrackingAppUserDisplayTemplate
112       description: Track & Trace Standalone Shipment Tracking Template App User Read Access
113       scope-references:
114         - "$XSAPPNAME.sst.r"
115         - uaa.user
116     - name: StandaloneShipmentTrackingAppUserProcessTemplate
117       description: Track & Trace Standalone Shipment Tracking Template App User Write Access
118       scope-references:
119         - "$XSAPPNAME.sst.w"
120         - uaa.user
121   - name: gtt-ui-sample-track-salesorders-app-host
122     type: org.cloudfoundry.existing-service
123   - name: gtt-ui-sample-track-shipments-app-host
124     type: org.cloudfoundry.existing-service
125   - name: gtt-ui-sample-track-purchaseorders-app-host
126     type: org.cloudfoundry.existing-service
127   - name: lbn-gtt-sample-destination
128     type: org.cloudfoundry.managed-service
129   parameters:
130     service-plan: lite
131     service: destination

```

- Open the *flp/portal-site/CommonDataModel.json* file
- Delete viz items *com.sap.gtt.app.sample.sst* and *com.sap.gtt.app.sample.pof* in *payload/catalogs/payload/viz* array
- Delete groups *shipments* and *purchaseOrder* in *payload/groups* array
- Delete orders *shipments* and *purchaseOrder* in *payload/sites/payload/groupsOrder* array

```

1 {
2   "_version": "3.0.0",
3   "identification": {
4     "id": "ee7d38df-c86c-48b1-8f02-c2abc4e18f54-1578021444073",
5     "entityType": "bundle"
6   },
7   "payload": {
8     "catalogs": [
9       {
10        "_version": "3.0.0",
11        "identification": {
12          "id": "defaultCatalogId",
13          "title": "{{title}}",
14          "entityType": "catalog",
15          "i18n": "i18n/defaultCatalogId.properties"
16        },
17        "payload": {
18          "viz": [
19            {
20              "id": "com.sap.gtt.app.sample.sof",
21              "vizId": "SalesOrder-track"
22            },
23            {
24              "id": "com.sap.gtt.app.sample.sst",
25              "vizId": "Shipment-track"
26            },
27            {
28              "id": "com.sap.gtt.app.sample.pof",
29              "vizId": "PurchaseOrder-track"
30            }
31          ]
32        }
33      }
34    ],
35    "groups": [
36      {
37        "_version": "3.0.0",
38        "identification": {
39          "id": "salesOrder",
40          "title": "{{title}}",
41          "entityType": "group",
42          "i18n": "i18n/salesOrder.properties"
43        },
44        "payload": {
45          "viz": [
46            {
47              "id": "com.sap.gtt.app.sample.sof",
48              "appId": "com.sap.gtt.app.sample.sof",
49              "vizId": "SalesOrder-track"
50            }
51          ]
52        }
53      }
54    ]
55  }

```

```

{
  "_version": "3.0.0",
  "identification": {
    "id": "shipment",
    "title": "{{title}}",
    "entityType": "group",
    "i18n": "i18n/shipment.properties"
  },
  "payload": {
    "viz": [
      {
        "id": "com.sap.gtt.app.sample.sst",
        "appId": "com.sap.gtt.app.sample.sst",
        "vizId": "Shipment-track"
      }
    ]
  },
  {
    "_version": "3.0.0",
    "identification": {
      "id": "purchaseOrder",
      "title": "{{title}}",
      "entityType": "group",
      "i18n": "i18n/purchaseOrder.properties"
    },
    "payload": {
      "viz": [
        {
          "id": "com.sap.gtt.app.sample.pof",
          "appId": "com.sap.gtt.app.sample.pof",
          "vizId": "PurchaseOrder-track"
        }
      ]
    }
  }
},
"sites": [
  {
    "_version": "3.0.0",
    "identification": {
      "id": "a3eee102-5181-4222-9924-8b8e90985f07-1578021444073",
      "entityType": "site",
      "title": "SAP Fiori launchpad site on Cloud Foundry",
      "description": "SAP Fiori launchpad site on Cloud Foundry, deployed from SAP Web IDE"
    },
    "payload": {
      "config": {
        "ucxshellConfie": {

```

```

90     ],
91     "sites": [
92         {
93             "_version": "3.0.0",
94             "identification": {
95                 "id": "a3eee102-5181-4222-9924-8b8e90985f07-1578021444073",
96                 "entityType": "site",
97                 "title": "SAP Fiori launchpad site on Cloud Foundry",
98                 "description": "SAP Fiori launchpad site on Cloud Foundry, deployed from SAP Web IDE"
99             },
100             "payload": {
101                 "config": {
102                     "ushellConfig": {
103                         "renderers": {
104                             "fiori2": {
105                                 "componentData": {
106                                     "config": {
107                                         "applications": {
108                                             "Shell-home": {}
109                                         },
110                                         "enableHelp": false,
111                                         "enableSearch": false,
112                                         "sessionTimeoutIntervalInMinutes": 19,
113                                         "sessionTimeoutReminderInMinutes": 3
114                                     }
115                                 }
116                             }
117                         }
118                     }
119                 },
120                 "groupsOrder": [
121                     "salesOrder",
122                     "shipment",
123                     "purchaseOrder"
124                 ],

```

In the command line tool,

- Run the following command to build the MTAR package:

mbt build

A MTAR file *./mta_archives/lbn-gtt-sample-app-flp_xxx.mtar* is then generated.

- Deploy the MTAR package in Cloud Foundry by running the following command:

cf deploy ./mta_archives/lbn-gtt-sample-app-flp_xxx.mtar

2.4.4 Package and Deploy *Track Sales Orders* Backend Service

Replace User/Password, Here Map License Key and Host Configuration

- Open the file `mta.yaml` under the folder “GTT-V2-Sample-TrackSalesOrders-Service”:

```
1  ---
2  _schema-version: '3.1'
3  ID: gtt-v2-sample-track-salesorders-service
4  version: 2.3.0
5  modules:
6  - name: gtt-sample-track-salesorders-service
7    type: java
8    path: srv/
9    properties:
10     SPRING_PROFILES_ACTIVE: cloud
11     DESTINATION_GTT: gtt_core_engine_flp_base_url
12     HERE_MAP_KEY: {here-map-license-key}
13    parameters:
14     buildpack: java_buildpack
15     health-check-timeout: 180
16     host: gtt-sample-track-salesorders-service-{unique-identifier}
17    build-parameters:
18     ignore:
19     - "*.md"
20     build-result: target/*.jar
21     timeout: 10m
22     requires:
23     - name: lbn-gtt-sample-app-uaa
24
25  resources:
26  - name: lbn-gtt-sample-app-uaa
27    type: org.cloudfoundry.existing-service
28
29
```

- Replace `{here-map-license-key}` with the license key that you can get from step 4.1.
Note: This step is optional, you could ignore this to proceed to verify if the application could be built and pushed successfully. After the actual license key is obtained, you could come back and replace it and execute this step again to build and push the application to cloud so that the map could be shown.
- Replace `{unique-identifier}` with the unique information such as your company name or cloud foundry global account name.
Note:
 - `{unique-identifier}` must only contain lowercase letters, numbers and hyphens. For example, “company-s01”.
 - For the length of “host”, the maximum limit is 63 characters.
- Open the file `./srv/src/main/java/com/sap/gtt/v2/sample/sof/constant/Constants.java`
 - Replace “com.lbngttsamples.gtt.app.sof” with the model namespace you got from Step 2.3

```

1 package com.sap.gtt.v2.sample.sof.constant;
2
3 public class Constants {
4     public static final String GTT_MODEL_NAMESPACE = "com.lbngttsamples.gtt.app.sof";
5     public static final String GTT_MODEL_NAMESPACE_WRITE_SERVICE = GTT_MODEL_NAMESPACE + ".sofWriteService";
6     public static final String MODEL_NAMESPACE = GTT_MODEL_NAMESPACE + ".sofService";
7
8     public static final String URL_SPLITTER = "/";
9     public static final String ENTITY_CONTAINER_NAME = "EntityContainer";
10    public static final String SAP_ICON_LEAD = "sap-icon://lead";
11    public static final String ID = "id";
12
13    public static final String EVENT_STATUS_PLANNED = "PLANNED";
14
15    public static final String PROCESS_STATUS_SUPPRESSED = "SUPPRESSED";

```

- Run the following command to build the MTAR package:

mbt build

A MTAR file *./mta_archives/gtt-v2-sample-track-salesorders-service_xxx.mtar* is then generated.

- Deploy the MTAR package in Cloud Foundry by running the following command:

cf deploy ./mta_archives/gtt-v2-sample-track-salesorders-service_xxx.mtar

2.4.5 Check Deployed Apps on SAP Cloud Foundry Cockpit

After operations from Step 2.4.1 to Step 2.4.3, check the deployed apps on SAP Cloud Platform Cockpit.

The screenshot shows the SAP Cloud Platform Cockpit interface. The left sidebar contains navigation links for Applications, Services, SAP HANA Cloud, SAP HANA, Database Systems, Tenant Databases, Service Requests, Portal, Routes, Security Groups, Events, and Members. The main area displays a table of applications in the 'Space: [redacted] Applications'.

Requested State	Name	Instances	Disk Quota	Memory
Started	gtt-sample-track-salesorders-service	1/1	1024 MB	1024 MB
Stopped	gtt-ui-sample-track-salesorders-ui_deployer	0/1	1024 MB	256 MB
Started	lbn-gtt-sample-app-appRouter	1/1	256 MB	256 MB

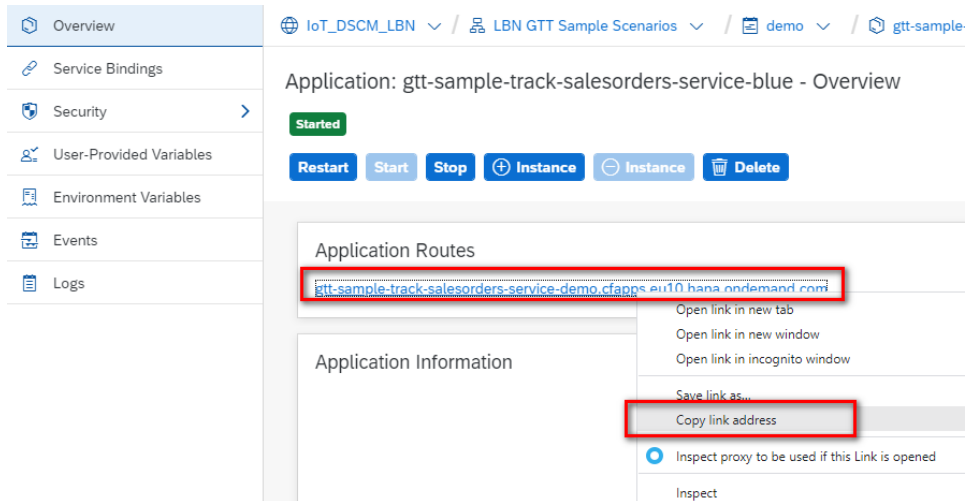
2.4.6 Maintain Destination of Forwarding

- Obtain the UAA Token
- Go to the *Track Sales Orders* backend application that you have deployed

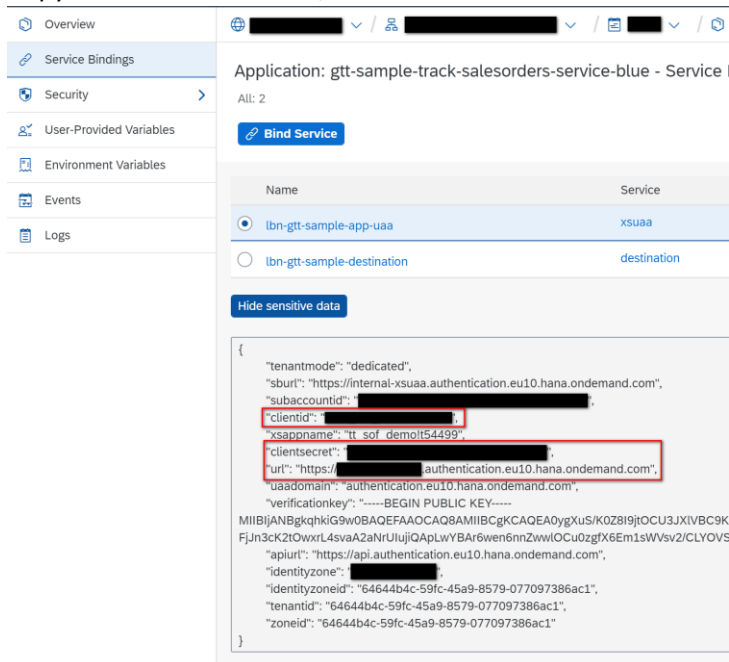
The screenshot shows a single application entry from the table above, with the state 'Started' highlighted in a green box.

Started	gtt-sample-track-salesorders-service-green	1/1
---------	--	-----

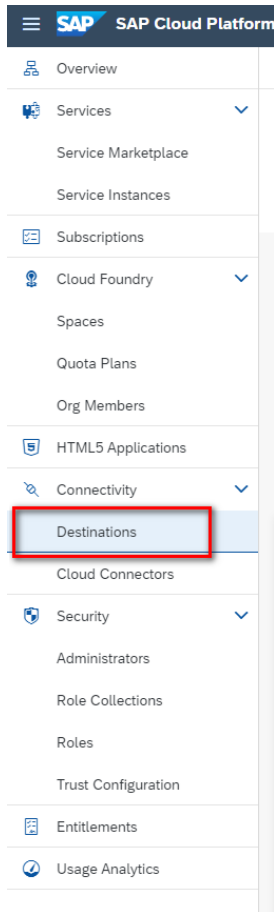
- Copy the **link address** of application routes



- Find the environment of UAA
- Copy the value of **clientid** ,**clientsecret** and the url



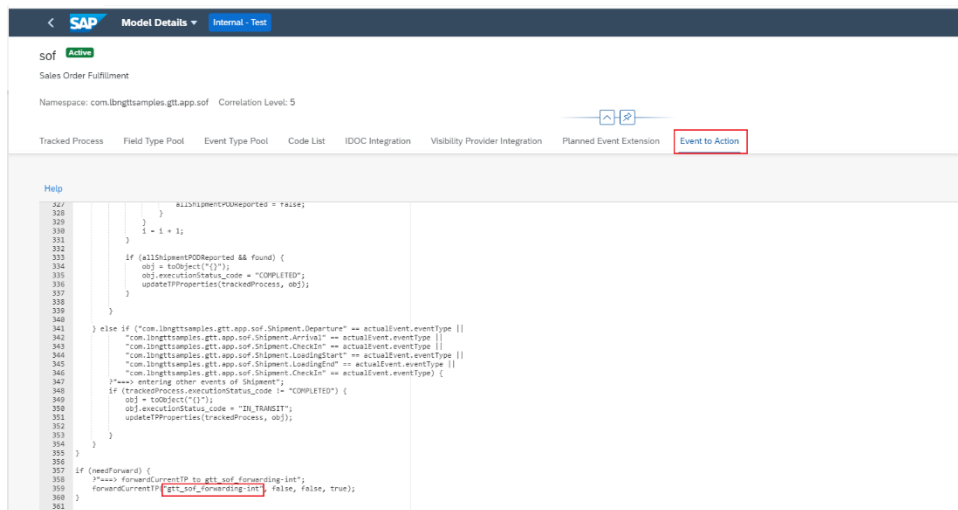
- Go to “Destinations”



c. Configure Destination of Event to Action Forwarding

Name	e.g. gtt_sof_forwarding-int
Type	HTTP
Description	<optional>
URL	<p>Enter the link address of application routes copied in the previous step, and add <code>"/sap/logistics/gtt/sample/sof/rest/v1/forward"</code> at the end of it.</p> <p>For example: https://gtt-sample-track-salesorders-service.cfapps.eu10.hana.ondemand.com/sap/logistics/gtt/sample/sof/rest/v1/forward</p>
Proxy Type	Internet
Authentication	OAuth2ClientCredentials
Client ID	Copied in the previous step
Client Secret	Copied in the previous step
Token Service URL	<p>Enter the URL copied in previous step and add <code>"/oauth/token"</code> at the end of the it.</p> <p>For example: https://lbn-gtt-samples.authentication.eu10.hana.ondemand.com/oauth/token</p>

Note: You need to update the destination name in *Event to Action* script of *Track Sales Orders* model if you change the name from “gtt_sof_forwarding-int” to others.



2.4.7 Maintain Roles and Users

As shown in Step 1.2, go to the role of “gtt-sample-all”, and add the following role:

Role	Application
SalesOrderFulfillmentAppUserDisplayTemplate	tt_sof_{space}

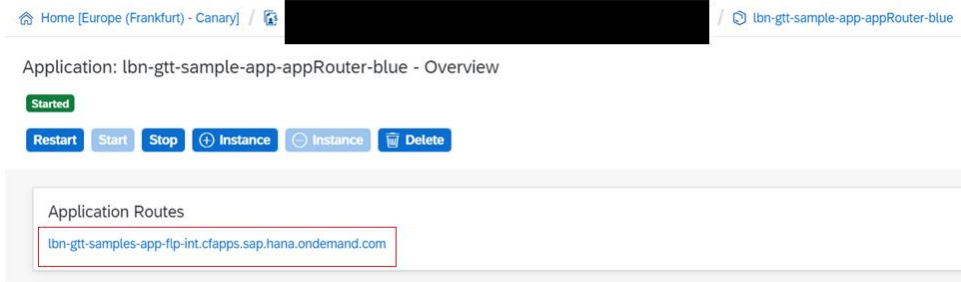
Note: the role of “SalesOrderFulfillmentAppDisplayTemplate” is used to access the *Track SO Fulfillment* application.

If you need to access other template applications such as *Track Shipments* or *Track PO Fulfillment*, add the following roles:

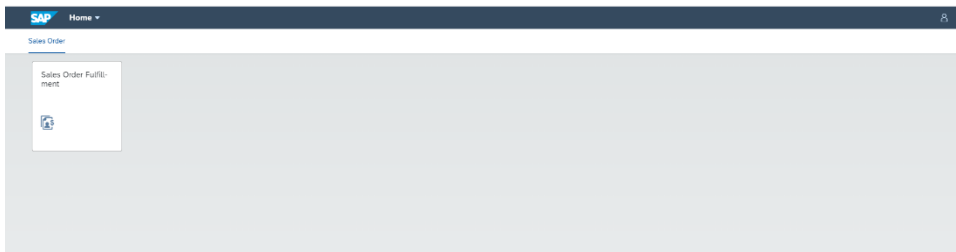
Role	Application
StandaloneShipmentTrackingAppUserDisplayTemplate	tt_sof_{space}
StandaloneShipmentTrackingAppUserProcessTemplate	tt_sof_{space}
PurchaseOrderFulfillmentAppUserDisplayTemplate	tt_sof_{space}

2.4.8 Open *Track Sales Orders* Template App from SAP Fiori Launchpad

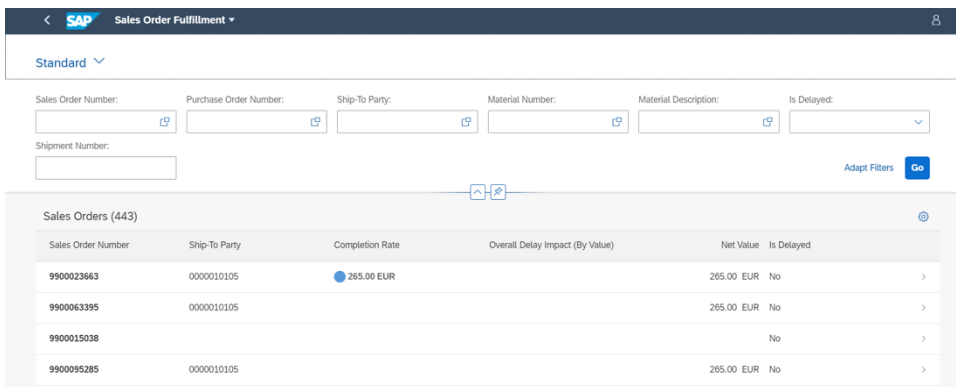
- Obtain the Application Routes from SAP Cloud Foundry Cockpit



- Click the link of Application Routes
The system browser then navigates to the SAP Fiori Launchpad of *Track Sales Orders*



- Open the *Track Sales Orders* template app



3. Implement ERP Extractor

For ERP extractor implementation, please refer to the following document:

02_Extractor_Creation_Guide-SOF.pdf

4. Other Configuration

4.1 Obtain Here Map License

To have the map view, log the incident with the URL of Fiori Launchpad of template apps in SAP support system under component "SBN-LBN-GTT-APP". And follow the 2 steps below when you create an incident:

- 1) Add logon credentials in the incident to the secure area so that we can provide you the key in a safe way. You can refer to SAP Note 1773689 for reference:
<https://launchpad.support.sap.com/#/notes/0001773689>
- 2) Provide us with the application routes which you get in step 2.4.8 “Obtain the Application Routes from SAP Cloud Foundry Cockpit”

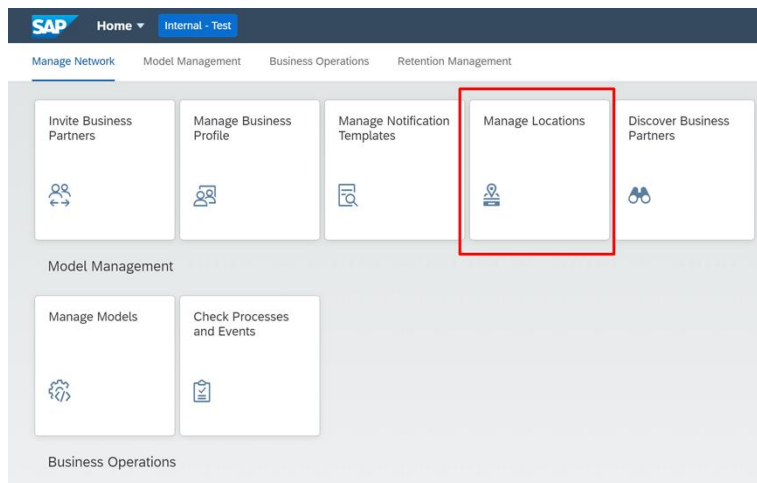
After you get the key, you can use it in Step 2.4.4 to re-deploy the backend service.

For quick validation, you can also use the tile service of other map services.

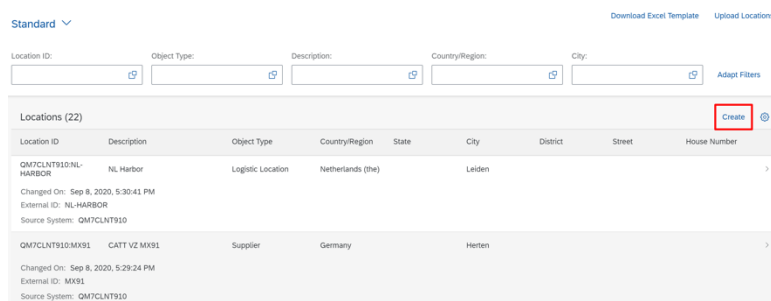
4.2 Maintain Location Master

To show stops and planned events in the map, you need to maintain location master data with valid geocoordinates in *Manage Locations* app.

- a. Open *Manage Locations* app in your SAP Fiori Launchpad for global track and trace option



- b. Click “Create” to create a new location



- c. Maintain location information with valid geocoordinates

To integrate with ERP, you need to keep the Object Type, Source System, and External ID location consistent with ERP location information.

To know the details, please refer to the following document:

< SAP Manage Locations > Internal - Test

Create Location

GENERAL ALTERNATIVE LOCATION IDS

Object Type: *
Logistic Location

Source System/External ID *
Source System External ID

Description: *

Time Zone:

Country/Region:

State:

City:

District:

Street/House Number:

Zip Code:

Longitude: * (East) Example: 000.00000000

Latitude: * (North) Example: 00.00000000

4.3 Maintain Data Contributors

To receive actual events from other business partners, you need to invite them to join your business network. For more, search for *Administration Guide for Version 2* at help.sap.com/gtt and see the chapter “Onboarding and User Management (Data Contributors)”.

4.4 Integrate with Visibility Provider

To integrate with visibility provider, log your incident in SAP Support system with the component “SBN-LBN-GTT-APP”.