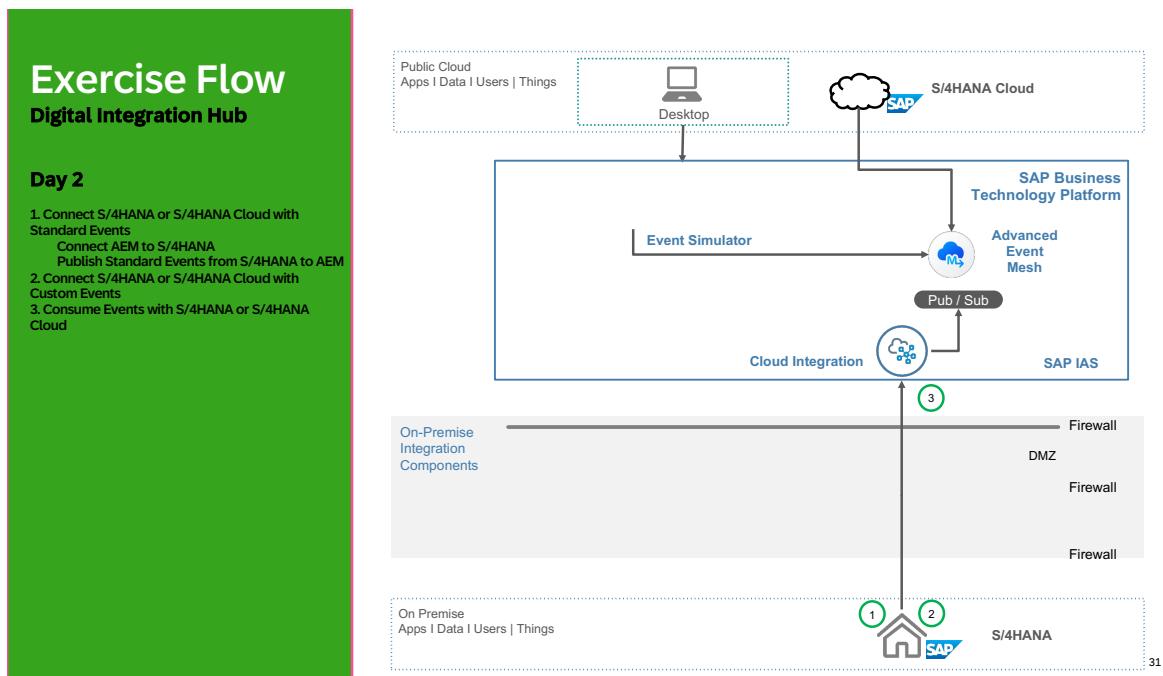


Day 2

1. Connect S/4HANA or S/4HANA Cloud with Standard Events
 - a. Connect AEM to S/4HANA
 - b. Publish Standard Events from S/4HANA to AEM
2. Connect S/4HANA or S/4HANA Cloud with Custom Events
3. Receive Events with S/4HANA or S/4HANA Cloud



Connect S/4HANA or S/4HANA Cloud with Standard Events

In this task we'll be connecting the Advanced Event Mesh to the S/4HANA system and configuring the Business Partner Changed event as a standard event to be sent to the AEM. In doing so we'll be creating two destinations within transaction *SM59*. One for the AEM broker and one for the VMR. While the destination for the AEM broker has to be setup manually, the destination for VMR can be setup automatically via the service key from the BTP.

Connect AEM to S/4HANA

- From the SAP AEM Console, open the Cluster Manager.

Welcome to SAP Integration Suite, advanced event mesh

Subaccount ID: aaa3d8bd-9746-4356-9f6d-4e7c731aa78

Event Portal

- Designer**: Design your event driven architecture components
- Catalog**: Browse or search for events, schemas, and applications across your enterprise
- Runtime Event Manager**: Discover, govern, and visualize the flow of events between your applications in each environment
- KPI Dashboard**: View metrics for your event use in your event driven architecture

Mission Control

- Cluster Manager**: Control the lifecycle of your Event Broker Services (highlighted with a red box)
- Mesh Manager**: Create and manage your event meshes

Insights

- Select your broker

Cluster Manager: Services

Create Service

Filter by service name Only show my services

All services

Broker 100	AEM_CommunitySubregion
AEM_CommunityCentral eks-eu-central-1a-2 Florian.oko@sap.com Running	AEM_CommunitySubregion aks-francecentral n.lorenz@sap.com Running

+

3. Select the “Connect” tab

The screenshot shows the SAP Cluster Manager interface. On the left is a sidebar with icons for Event Portal, Designer, Catalog, Runtime Manager, KPI, Mission Control, Cluster Manager (which is selected), Mesh Manager, and Insights. The main area is titled "Service Details: AEM_CommunityCentral". The "Connect" tab is active, indicated by a red box around its tab header. Below the tabs are three circular progress bars: "Active Connections" (16.0 %), "Guaranteed Messaging Endpoints" (16.0 %), and "Queue Usage" (0.0 %). The "Active Connections" bar has data points: AMQP 5, MQTT 0, SMF 11, REST 0, Web 0. The "Guaranteed Messaging Endpoints" bar has data points: Queues 16, Topic Endpoints 0. The "Queue Usage" bar has data points: Messages Queued 12566, Spool Usage 9 MB. Under "Availability and Versioning", it says "Service State: Running", "Event Broker Service Version: 10.8.1.126-1", "Service ID: 69afkf9w2od", "Created By: Florian.okos@sap.com (Florian.okos@sap.com)", and "Service Creation Time: 25/06/2024, 17:04:24". Under "DMR Cluster", it shows "Hostname: mr-connection-h9tikb3o1b6w.messaging.solace.cloud", "Cluster Name: cluster-eks-eu-central-1a-2-lvcu3jkbetw", "Primary Router Name: 69afkf9w2odprimary", and "Cluster Password: *****". Under "Management Access", it shows "Basic Authentication: Enabled", "LDAP Authentication: Disabled", and "SSO Authentication: Disabled". Under "Distributed Tracing", it says "Distributed Tracing is enabled on this service." and provides a link to "View Tracing Details".

4. Open the “View by” drop down and select “Protocol”

The screenshot shows the same SAP Cluster Manager interface as the previous one, but the "View by" dropdown menu is open, highlighting the "Protocol" option. The menu also includes "Language", which is currently selected. To the right of the dropdown, there is a list of supported client libraries, each with a small icon: C, Python, Go, JavaScript, Node.js, .NET, and Spring.

5. Open “AMQP” and press the “Get Started” Button

The screenshot shows the SAP Cluster Manager interface. In the top navigation bar, the 'Connect' tab is selected. Below it, under 'Service Details: AEM_CommunityCentral', there's a section titled 'Connect Using a Supported Client Library'. This section lists several connection points:

- Solace Messaging
- Solace Web Messaging
- AMQP** (highlighted with a red box)
- MQTT
- REST

For each connection point, there's a table with columns 'Technology', 'Library', and 'Language'. The 'AMQP' row shows:

Technology	Library	Language
Qpid	QPID JMS 1.1	Java
Qpid	QPID JMS 2.0	Java
AMQP	AMQP10 Open Source API	Node.js

Next to the 'AMQP' row, there are three 'Get Started' buttons, all of which are highlighted with red boxes.

6. Copy the “Host URLs”

This screenshot shows the same SAP Cluster Manager interface as the previous one, but with a different connection point selected: 'QPID JMS 1.1'. A modal window has opened for this specific connection point.

The modal has tabs for 'Connection Details' and 'Host URLs'. Under 'Connection Details', there are fields for 'Username' (set to 'solace-cloud-client') and 'Password' (redacted). Under 'Host URLs', the 'Secured AMQP URI' field contains the value: `amqps://mconnection-h91kb3o1bw.messaging.solace.cloud:5671`. This entire field is highlighted with a red box.

7. Open transaction **SM59** and create a new destination

The screenshot shows the SAP Configuration of RFC Connections interface. At the top, there's a toolbar with various icons. Below it is a header bar with the title "SAP Configuration of RFC Connections". Underneath the header, there are two status messages: "Generate RFC Callback Allowlist" and "Activate Non-Empty Allowlists". A note below states "RFC callback check not secure". The main area is a table titled "RFC Connections" with columns for Type, PL Act., and Comment. The rows listed are: ABAP Connections (Type 3), HTTP Connections to External Server (Type G), HTTP Connections to ABAP System (Type H), Internal Connections (Type I), Logical Connections (Type L), TCP/IP Connections (Type T), WebSocket RFC (Type W), and Connections Using ABAP Driver (Type X). A red box highlights the "New" icon (a plus sign inside a square) in the toolbar.

8. Name the destination “AEM_BROKER_CONNECTION” and select Connection Type “G”

The screenshot shows the "Create Destination" dialog box. It has three tabs at the top: a red tab, a grey tab, and a green tab. The green tab is selected. The main area has two sections: "Destination" and "Connection Type". The "Destination" section contains a text input field with the value "AEM_BROKER_CONNECTION". The "Connection Type" section contains a dropdown menu with the value "G HTTP connection to external server". At the bottom right of the dialog are two buttons: a green checkmark button and a red X button.

9. Select Continue

10. Enter “Broker setting for AEM Broker” as description and under Target System Settings enter the previously copied host uri as host. Replace the starting

protocol “amqps” from the host with the protocol “tcps”

SAP RFC Destination AEM_BROKER_CONNECTION

Connection Test 

RFC Destination	AEM_BROKER_CONNECTION	
Connection Type	G HTTP Connection to External Server	
Description	Description	
Description 1	Broker setting for AEM Broker h91kb3o1b6w	
Description 2		
Description 3		

Administration **Technical Settings** Logon & Security Special Options

Target System Settings

Host	tcp://mr-connection-h91kb3o1b6w.messaging.s...	Port	5671
Path Prefix			

HTTP Proxy Options

Global Configuration	
Proxy Host	
Proxy Service	
Proxy User	
Proxy PW Status	is initial
Proxy Password	

11. Select the tab “Logon & Security” and set SSL to Active

SAP RFC Destination AEM_BROKER_CONNECTION

Connection Test

RFC Destination	AEM_BROKER_CONNECTION	
Connection Type	G HTTP Connection to External Server	
Description	Description 1: Broker setting for AEM Broker h91kb3o1b6w Description 2: Description 3:	

Administration Technical Settings **Logon & Security** Special Options

Logon Procedure

Logon with User

Do Not Use a User OAuth Settings
 Basic Authentication

User:
PW Status:

Logon with Ticket

Do Not Send Logon Ticket
 Send Logon Ticket Without Ref. to a Target System
 Send Assertion Ticket for Dedicated Target System

System ID: Client

Logon with MQTT/AMQP

Users:
PW Status:
Password:

Security Options

Status of Secure Protocol

SSL Inactive Active

SSL Client PSE ID: SSL Client (Standard) ▼
 Do Not Use Certificate for Logon

TLS Attributes ▼

12. Hit Save

The Enterprise Event Enablement integration within S/4HANA uses Client Certificate Authentication to connect to AEM. Therefore the next step is to export the necessary certificates from S/4HANA and import them to AEM.

13. Remember the “SSL Client PSE ID”

SAP RFC Destination AEM_BROKER_CONNECTION

Connection Test

RFC Destination	AEM_BROKER_CONNECTION	
Connection Type	G HTTP Connection to External Server	
Description	Description 1: Broker setting for AEM Broker h91kb3o1b6w Description 2: Description 3:	

Administration Technical Settings **Logon & Security** Special Options

Logon Procedure

Logon with User

Do Not Use a User OAuth Settings

Basic Authentication

User:

PW Status:

Logon with Ticket

Do Not Send Logon Ticket

Send Logon Ticket Without Ref. to a Target System

Send Assertion Ticket for Dedicated Target System

System ID: Client

Logon with MQTT/AMQP

Users:

PW Status:

Password:

Security Options

Status of Secure Protocol

SSL: Inactive Active

SSL Client PSE ID:

Do Not Use Certificate for Logon

TLS Attributes

That completes the setup of the first destination. Now lets configure authentication for the Advanced Event Mesh

14. Open the transaction **STRUST**

15. Select the SSL Client node given in the destination setup

16. Select your system own certificate and export it as base64

SAP Trust Manager: Display

The screenshot shows the SAP Trust Manager interface. On the left, a tree view lists various SSL client configurations. In the center, the 'SSL client SSL Client (Standard)' section is selected. Under 'Own Certificate', the subject is listed as 'CN=T33_SSLC_DEFAULT, OU=ISAP-INTERN, OU=SAP Web AS, O=SAP Trust Community, C=DE' (Self-Signed). Below this, a 'Certificate List' table shows several certificates from different issuers, including DigiCert and Thawte. At the bottom of the page, there are two red boxes highlighting the 'Export' and 'Import' buttons.

Owner	Valid from	Valid to
CN=DigiCert Global Root G2, OU=www.digicert.com, O=DigiCert Inc, C=US	01.08.2013	15.01.2038
CN=DigiCert Global Root CA, OU=www.digicert.com, O=DigiCert Inc, C=US	10.11.2006	10.11.2031
CN=DigiCert High Assurance EV Root CA, OU=www.digicert.com, O=DigiCert Inc, C=US	10.11.2006	10.11.2031
CN=Thawte RSA CA 2018, OU=www.digicert.com, O=DigiCert Inc, C=US	06.11.2017	06.11.2027

17. Remember the common name (CN)

SAP Trust Manager: Display

This screenshot is identical to the previous one, showing the SAP Trust Manager interface. The 'Own Certificate' section highlights the subject 'CN=T33_SSLC_DEFAULT, OU=ISAP-INTERN, OU=SAP Web AS, O=SAP Trust Community, C=DE' (Self-Signed). The 'Certificate List' table and the bottom buttons are also visible. A red box highlights the 'CN' value in the 'Subject' field of the 'Own Certificate' section.

Owner	Valid from	Valid to
CN=DigiCert Global Root G2, OU=www.digicert.com, O=DigiCert Inc, C=US	01.08.2013	15.01.2038
CN=DigiCert Global Root CA, OU=www.digicert.com, O=DigiCert Inc, C=US	10.11.2006	10.11.2031
CN=DigiCert High Assurance EV Root CA, OU=www.digicert.com, O=DigiCert Inc, C=US	10.11.2006	10.11.2031
CN=Thawte RSA CA 2018, OU=www.digicert.com, O=DigiCert Inc, C=US	06.11.2017	06.11.2027

18. In the AEM UI open the “Manage” tab of your broker and press “Authentication”

Service Details: AEM_CommunityCentral

Event Broker Service Settings

Authentication
Enabled

Certificate Authorities
2 Client Certificate Authorities
1 Domain Certificate Authority

Client Profiles
1 Client Profile

Broker Manager Quick Settings

Message VPN

Clients

Queues

Access Control

Bridges

Other Management Tools

SEMP - REST API

Broker Manager - Web Application

SolAdmin - Desktop Application

19. Enable “Client Certificate Authentication”

Service Authentication

Client Authentication

Basic Authentication

Type: Internal Database

Client Certificate Authentication

Validate Certificate Dates

OAuth Provider Authentication

Allow API Provided Username

Username Source: Common Name

LDAP Profile

Save

20. Go back to the “Manage” tab and select Certificate Authorities”

The screenshot shows the SAP Event Broker Service Settings page. The 'Manage' tab is selected. In the 'Event Broker Service Settings' section, there is a box labeled 'Certificate Authorities' containing '2 Client Certificate Authorities' and '1 Domain Certificate Authority'. This box is highlighted with a red border. Below this are sections for 'Broker Manager Quick Settings' (Message VPN, Clients, Queues, Access Control, Bridges) and 'Other Management Tools' (SEMP - REST API, Broker Manager - Web Application, SolAdmin - Desktop Application).

21. Click “Add Client Certificate Authority”, give the certificate authority name “S/4HANA” (You can also give more meaningful names) and upload the previously exported certificate.

The screenshot shows the 'Certificate Authorities' page. It displays two tabs: 'Client Certificate Authorities (2)' and 'Domain Certificate Authorities (1)'. The 'Client Certificate Authorities (2)' tab is selected. A red box highlights the 'Add Client Certificate Authority' button. Below it, a table lists two entries: 'T33' and 'T33PFEILA', each with edit and delete icons. There is also a 'Manage' link at the top right of the table.

22. Save the Certificate Authority

23. go back to the “Manage” tab of the Event Broker and select “Access Control”

The screenshot shows the SAP Cluster Manager interface. The left sidebar has sections like Event Portal, Designer, Catalog, Runtime Manager, KPI, Mission Control, Cluster Manager, Mesh Manager, and Insights. The main area is titled "Service Details: AEM_CommunityCentral". The top navigation bar includes Cluster Manager > Service Details, Status, Connect, Manage (selected), Monitoring, Configuration, Try Me!, Open Broker Manager, and more. Below the navigation is a section titled "Event Broker Service Settings" with three boxes: Authentication (with 2 Client Certificate Authorities and 1 Domain Certificate Authority), Certificate Authorities, and Client Profiles (with 1 Client Profile). Under "Broker Manager Quick Settings", there are icons for Message VPN, Clients, Queues, Access Control (which is highlighted with a red box), and Bridges. A section titled "Other Management Tools" lists SEMP - REST API, Broker Manager - Web Application, and SolAdmin - Desktop Application.

24. Select the tab “Client Usernames” and create a new entry

The screenshot shows the SAP Cluster Manager interface under the Client Authentication tab. The left sidebar includes sections like AEM_CommunityCentral, Change VPN, Messaging (Message VPN, Clients, Queues, Connector Wizards), Access Control (selected), Telemetry, Replay, Bridges, JMS JNDI, Try Me!, Advanced Messaging, Caches, Transactions, System (Cluster), and Clustering. The main area displays a table of Client Usernames. The columns are Client Username, Client Profile, ACL Profile, Enable, Subscription Manager, and Dynamic. The table lists entries such as #client-username, #rdp/BPA RDP, T33_SSLC_DEFAULT, default, email-profile, sc-dt-trace-collector, solace-cloud-client, and t33_Purchaseorderer. A search bar at the top is labeled "Search by name". In the top right corner of the table area, there is a blue button with a plus sign and the text "+ Client Username", which is also highlighted with a red box.

25. Enter the in step 17 mentioned common name (CN) (here T33_SSLC_DEFAULT) as Client Username and hit “Create”

Action	Subscription Manager	Dynamic
Enable	No	No
Yes	No	No
Yes	No	No
No	No	No
Yes	No	No
Yes	No	No

With the authentication configuration for the Advanced Event Mesh let's create the VMR destination within the S/4. Start by going into the BTP Cockpit of your suaccount with the AEM instance in it

26. Open “Instances and Subscriptions” and select the instance of your AEM

27. Press “View Credentials” and copy the service key

28. Go back to SAPGUI and open transaction /IWXB/E/CONFIG.

29. Create a new channel via “Service Key” > “Advanced”.

Status	Advanced	Active	Protocol	Destination	Topic Space	Service Plan	Description
○ Default	AEM_INT	<input checked="" type="checkbox"/>	AMQP	aem-coemeaint-connection	S/4HANA/Events	Advanced	AEM_COE_EMEA_INT_I553641
○ ○○ AEM_COMMUNITY		<input checked="" type="checkbox"/>	AMQP	AEM_COMMUNITY_CONNECT	S/4HANA/Events	Advanced	Connection to AEM
○ ○○ CF_CHANNEL		<input type="checkbox"/>	MQTT_311_WS	Event-Mesh-CF	sap/s4/beh	Default	Event Channel CF
○ ○○ CH_AEM_I517465		<input type="checkbox"/>	AMQP	I517465_AEM	s4/events/demo	Advanced	CH_AEM_I517465
○ ○○ CH_EM_D073297		<input type="checkbox"/>	AMQP	EVENT_MESH_I572275	sap/dev/0002	Default	Test
○ ○○ CH_EM_I572275		<input type="checkbox"/>	AMQP	EVENT_MESH_I572275	sap/dev/0002	Default	Channel for DEMO purposes
○ ○○ COE_EMEA_TECH		<input type="checkbox"/>	AMQP	COE_EMEA_TECH	coe/tech/s4h	Default	Event Mesh for COE_EMEA_TECH
○ ○○ COMMUNITY_EM_I585208		<input checked="" type="checkbox"/>	AMQP	community-em-i585208	community/em/i585208	Default	Event Mesh channel to user instance
○ ○○ EM_BTP_I054810		<input type="checkbox"/>	AMQP	EVENTMESH_AMQP	default/s4dev/events	Default	S4H
○ ○○ EM_BTP_ITA		<input type="checkbox"/>	SCPEMS		sap/s4/beh	Default	sap/s4/beh
○ ○○ EM_CHANNEL_DEMO		<input checked="" type="checkbox"/>	AMQP	FA163E75AABA1EDF84F07CB96708189B	br/sandbox/eventmesh	Default	Demo channel Event Mesh
○ ○○ EM_COM_CHANNEL		<input type="checkbox"/>	MQTT_311_WS	EM_EVENT_SO	sap/s4/beh	Default	em community channel
○ ○○ EMIS_I517465		<input type="checkbox"/>	AMQP	I517465	sap/demo/emis	Default	Channel to connect to Event Mesh in SAP Integration Suite
○ ○○ I054810_AEM		<input checked="" type="checkbox"/>	AMQP	I054810_AEM	default/t33/Events	Advanced	I054810_AEM
○ ○○ I553641_EM		<input checked="" type="checkbox"/>	AMQP	FA163E75AABA1EDF99A4AE4573A65BA0	I553641/test/now	Default	I553641_EM
○ ○○ IB23_D073297_EMCHANNEL		<input checked="" type="checkbox"/>	AMQP	ib23.D073297.eventmesh	ib23/D073297/s4	Default	Integration Bootcamp 23

30. Enter the following data and paste the previously copied service key

Channel: AEM_BROKER

Description: Channel to the AEM Broker

Destination: AEM_BROKER_CONNECTION (from step 8)

Channel: AEM_BROKER

Description: Channel to the AEM Broker

Protocol: AMQP

Destination: AEM_BROKER_CONNECTION

Topic Space: S/4HANA/Events

Preferred Event Mesh Validation Endpoint Name:

Preferred OAuth 2.0 Configuration Name:

Daemon User: I538921

Number of Publish Connections: 1

Service Key:

```
{
  "xsappname": [REDACTED],
  "handshake": {
    "oa2": {
      "clientid": [REDACTED],
      "clientsecret": [REDACTED],
      "granttype": [REDACTED],
      "tokenendpoint": [REDACTED]
    },
    "uri": [REDACTED],
    "serviceinstanceid": [REDACTED]
  }
}
```

31. Press save

With that the connection between the AEM and S/4HANA system should be setup. You can verify the setup by checking the AMQP connections within the AEM dashboard. Simply open “Clients” under the “Manage” tab of the event broker. There should be two

new connections. One for publishing and one for receiving events

The screenshot shows the SAP Solace Clients interface. At the top, there are tabs for Clients Summary, Solace Clients, MQTT, REST, and AMQP. Below the tabs, it says "11 Client". There is a search bar with the placeholder "Search by name". A table lists two clients:

Name	Client Username	Subscriptions	Incoming Message Discards	Outgoing Message Discards	No Subscription Match	Client Address	Slow Subscriber
#amqp/client/341215995a9ea802/0AB4011...	T33_SSCLC_DEFAULT	0	0	0	0	130.214.228.42:29200	No
#amqp/client/3702b5b6d9611711/0AB40116...	T33_SSCLC_DEFAULT	0	0	0	0	130.214.228.42:48566	No

Publish Standard Events from S/4HANA to AEM

To publish a standard event to the AEM you will need to create a topic binding on the channel created in the previous section. In this exercise we will create a binding for the standard [Business Partner Changed](#) event.

1. Open transaction **/IWXB/E/OUTBOUND_CFG**
2. Select your AEM channel and click “Create new topic binding”

The screenshot shows the SAP Outbound Binding Configuration interface. The title bar says "SAP Outbound Binding Configuration". On the left, there is a sidebar with "Channels" expanded, showing "Active Channels" and "Advanced Service Plan". Under "Advanced Service Plan", the "AEM_BROKER" channel is selected. On the right, the main area is titled "Outbound Bindings of Channel AEM_BROKER". It shows a toolbar with icons for New, Edit, Delete, and others. A red box highlights the "New" icon. Below the toolbar, there is a search bar with "Stat...", "Filter...", and "Topic".

3. Enter the topic
“sap/s4/beh/businesspartner/v1/BusinessPartner/Changed/*” and hit save

The screenshot shows the "Create Outbound Binding" dialog box. At the top, there are three radio buttons. The middle one is selected. The title bar says "Create Outbound Binding". Below the title, there is a "Topic" field containing "sap/s4/beh/businesspartner/v1/BusinessPartner/Changed/*". At the bottom right of the dialog, there are "Save" and "Cancel" buttons.

The names of all standard business events can be found in the [SAP Business Accelerator Hub](#). Simply remove the leading “ce/” from the event name for the topic binding.

4. In the AEM UI select the “Manage” tab and open “Queues”

Cluster Manager > Service Details
Service Details: AEM_CommunityCentral Status Connect Manage Monitoring Configuration Try Me! Open Broker Manager ...

Event Broker Service Settings

Authentication Enabled

Certificate Authorities 2 Client Certificate Authorities 1 Domain Certificate Authority

Client Profiles 1 Client Profile

Broker Manager Quick Settings

Message VPN Clients Queues Access Control Bridges

Other Management Tools

SEMP - REST API The Solace Element Management Protocol (SEMP) is a REST API that you can use to manage the Event Broker Service. API

Broker Manager - Web Application The Broker Manager is a browser-based administration console that you can use to manage the Event Broker Service. Settings

SolAdmin - Desktop Application SolAdmin is a legacy desktop application that you can use to manage the Event Broker Service. Desktop

5. Create a new Queue and give it the name “Business_Partner”

Create Queue

Queue Name

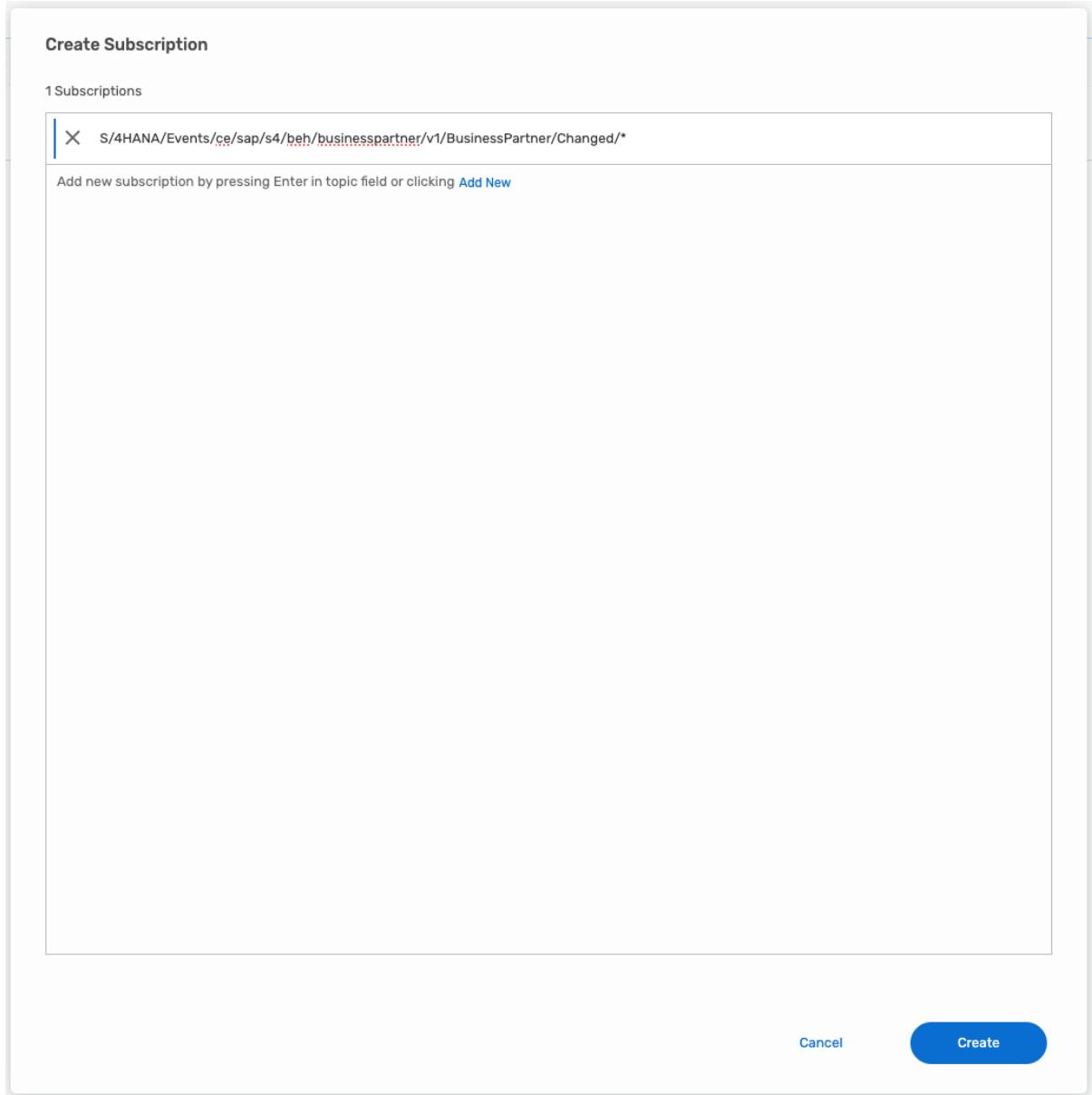
Business_Partner

Cancel Create

6. Select the queue and navigate to “Subscriptions”.

7. Create a new Subscription with topic

“S/4HANA/Events/ce/sap/s4/beh/businesspartner/v1/BusinessPartner/Changed/*”



*Note: Standard events published by S/4HANA will always include the namespace that was given during channel creation in the topic. By default this is “S/4HANA/Events”. “/ce” is this time included to indicate that the event is Cloud.Event.io conform.

Making a change to a business partner within transaction *BP* will now trigger an event that is being bound by the created channel. The Enterprise Event Enablement will then perform a handshake with the VMR/validation service to confirm that the AEM broker is a valid SAP broker and afterwards push the event. Since the created queue is subscribed to the topic of business partner changes the event will be queued. This can be verified by looking at the “Messages Queued” within the queue.

Action	Message ID	Spooled Time	Content Size (B)	Attachment Size (B)	Priority	Undelivered	Redeliveries	DMQ Eligible	Replication State
View	21220	2024-09-06 16:58:12	0	532	Yes	0	Yes	Yes	Not replicated
View	21222	2024-09-06 16:58:12	0	532	Yes	0	Yes	Yes	Not replicated

Connect S/4HANA or S/4HANA Cloud with Custom Events

In this next task, we're going to look at creating custom business events from scratch and publishing them to the Advanced Event Mesh. To be precise we'll be creating a new custom online shop application and defining a items ordered event to indicate that a new order has been placed.

To start lets create the basic online shop application:

1. Create a new package “Z_ONLINE_SHOP_000”. The “000” can be replaced by any dedicated number if multiple users are trying to follow this tutorial. Right click on

“Favorite Packages” and select “New” > “ABAP Package”

✓ T33_800_i538921_en [T33, 800, I538921, EN]

> Local Objects (\$TMP)

> Favorite

> Favorites

> System

> Application

> Release

New

Duplicate Tree...

Show In

Copy

⌘ C

Delete

⌦

Add Package...

Refresh

F5

Coverage As

>

Coverage As

>

Run As

>

Debug As

>

Profile As

>

Expand Tree by...

Configure Tree... ⌘ I

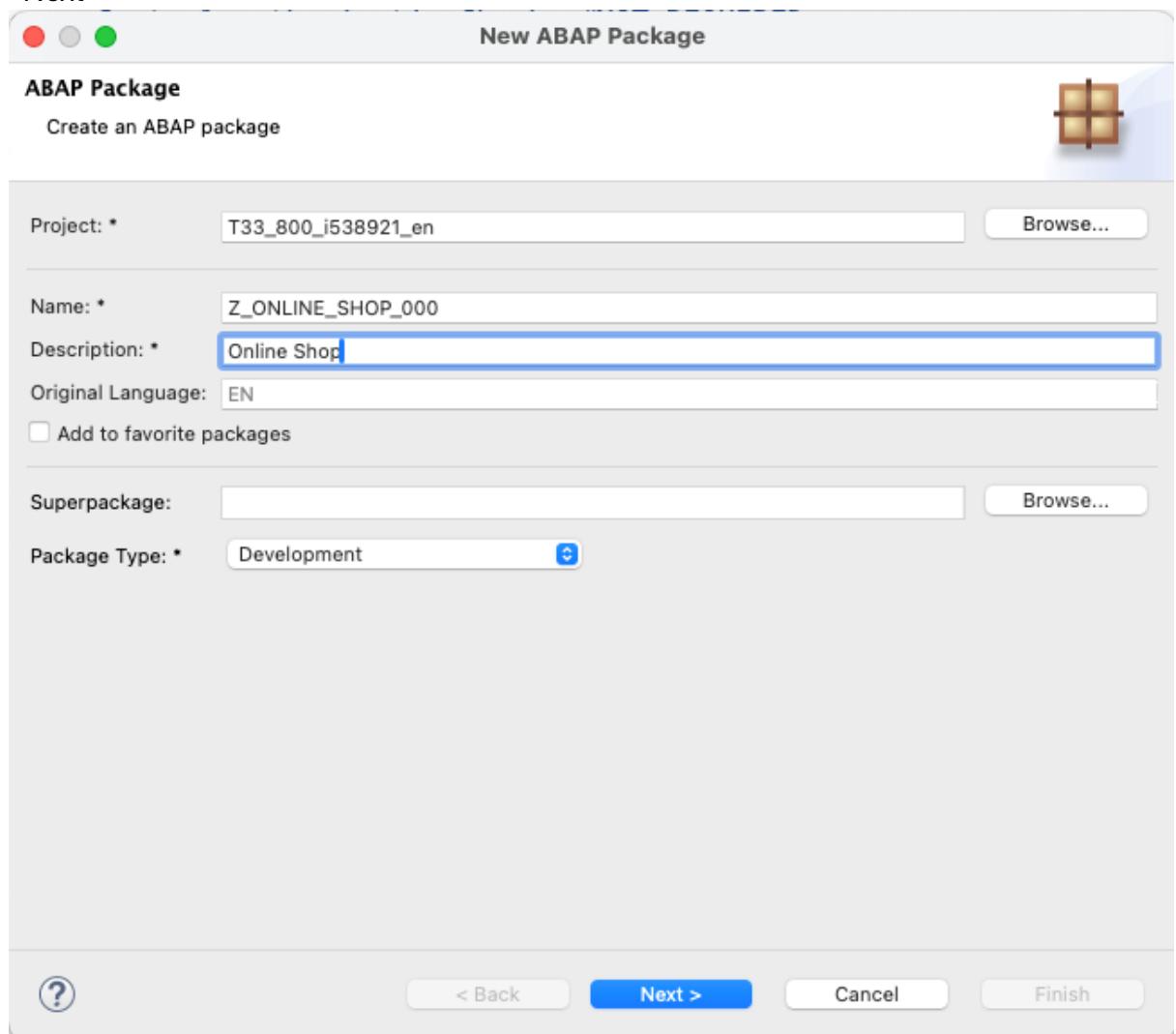
ABAP Class

ABAP Interface

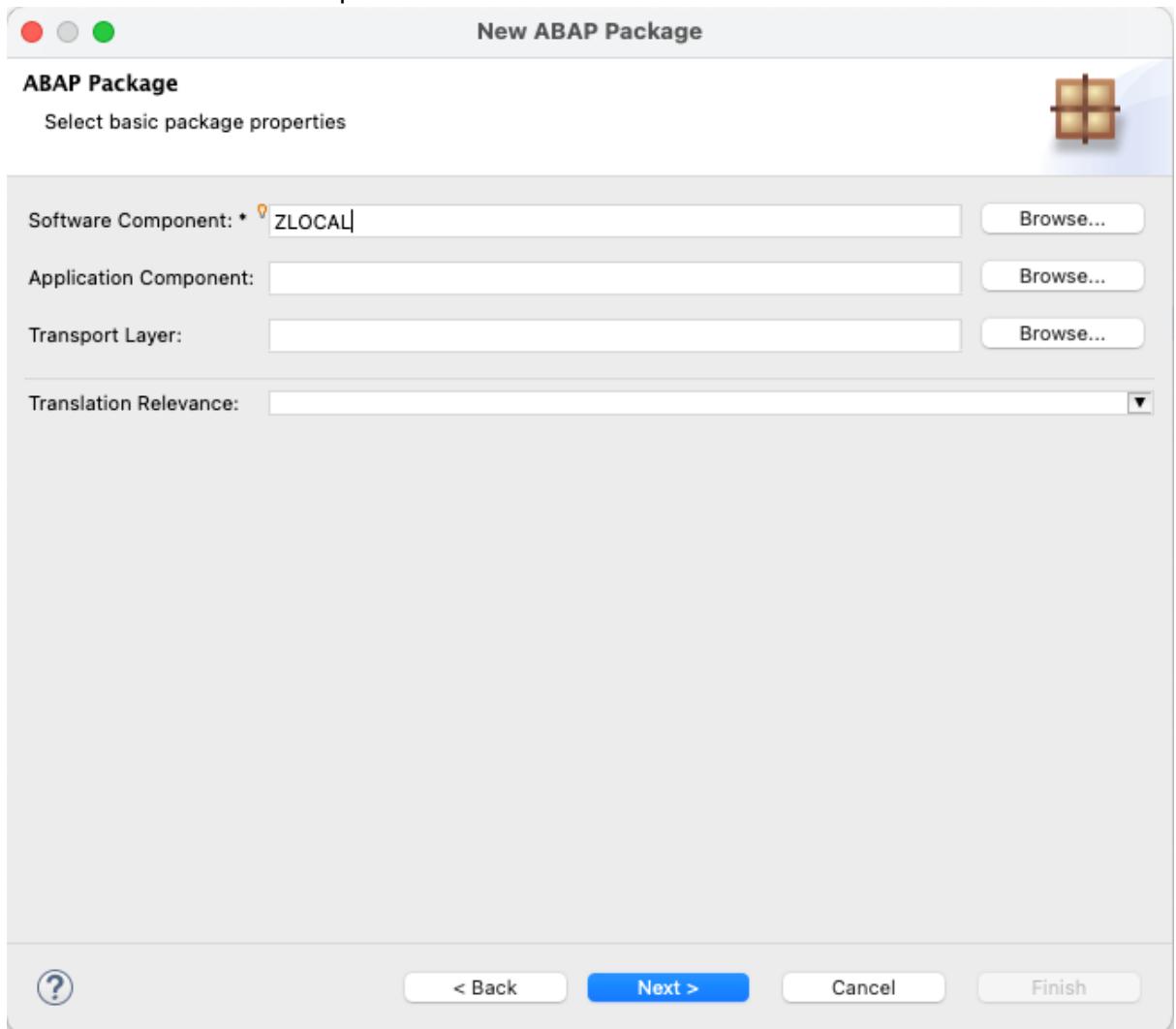
ABAP Package

Other ABAP Repository Object

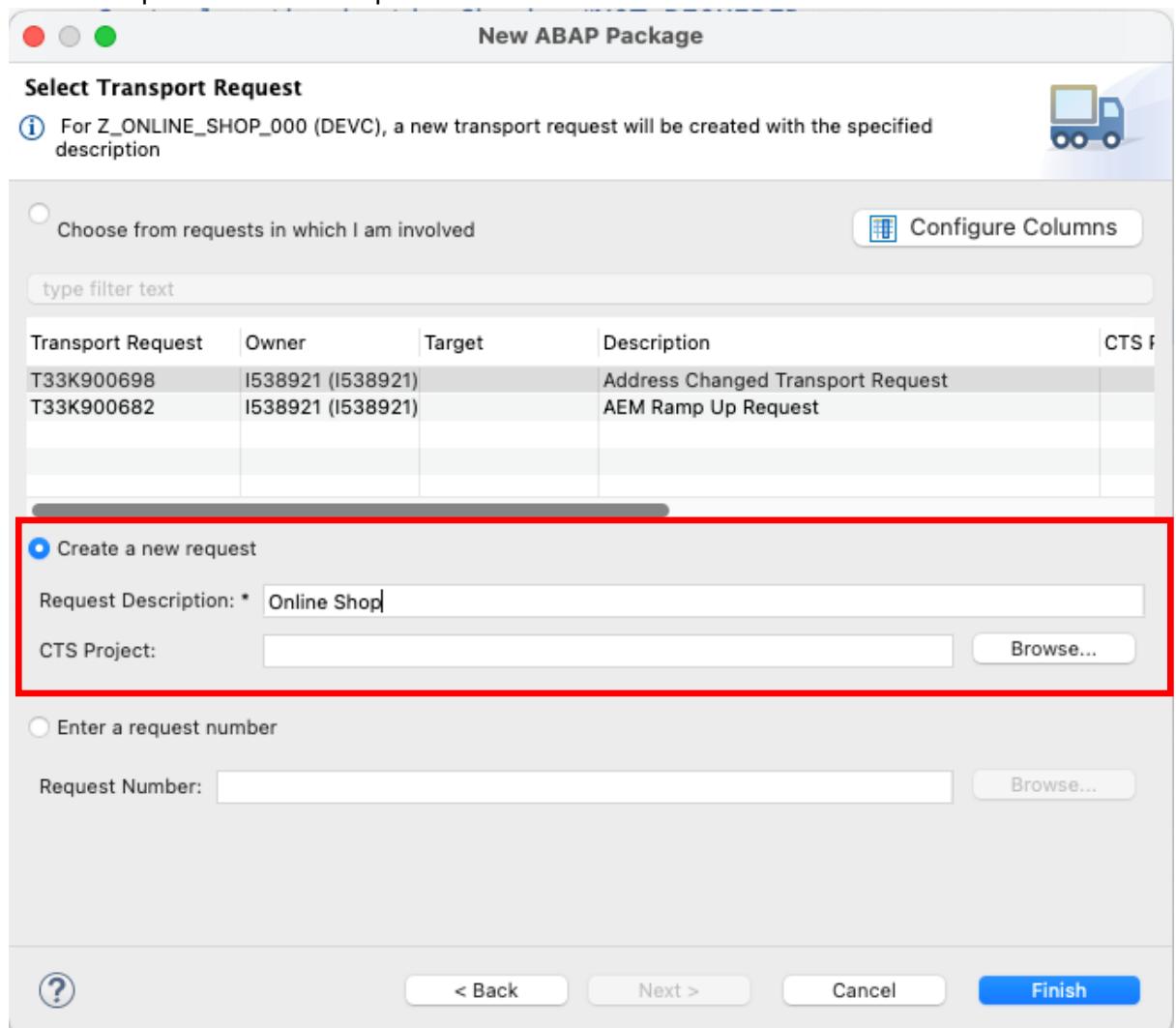
2. Give the name “Z_ONLINE_SHOP_000”, the description “Online Shop” and hit “Next”



3. Select the Software Component “ZLOCAL” and hit “Next”



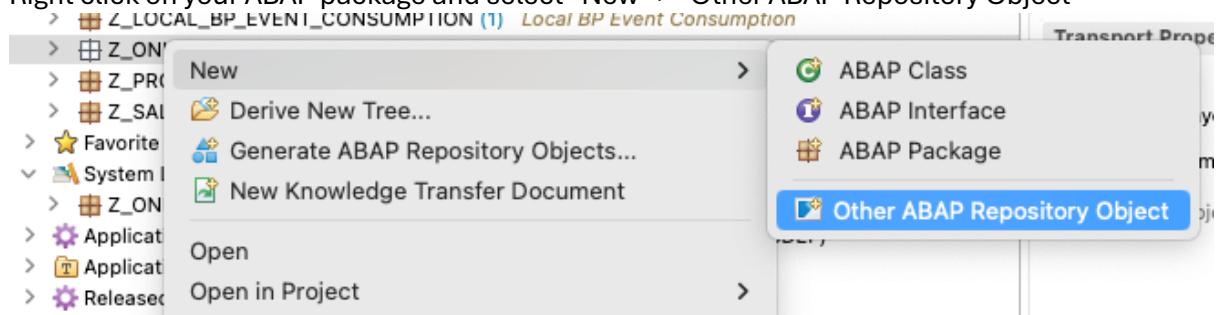
- Create a new transport request by selecting “Create a new request” and giving the description “Online Shop”



- Press “Finish”.

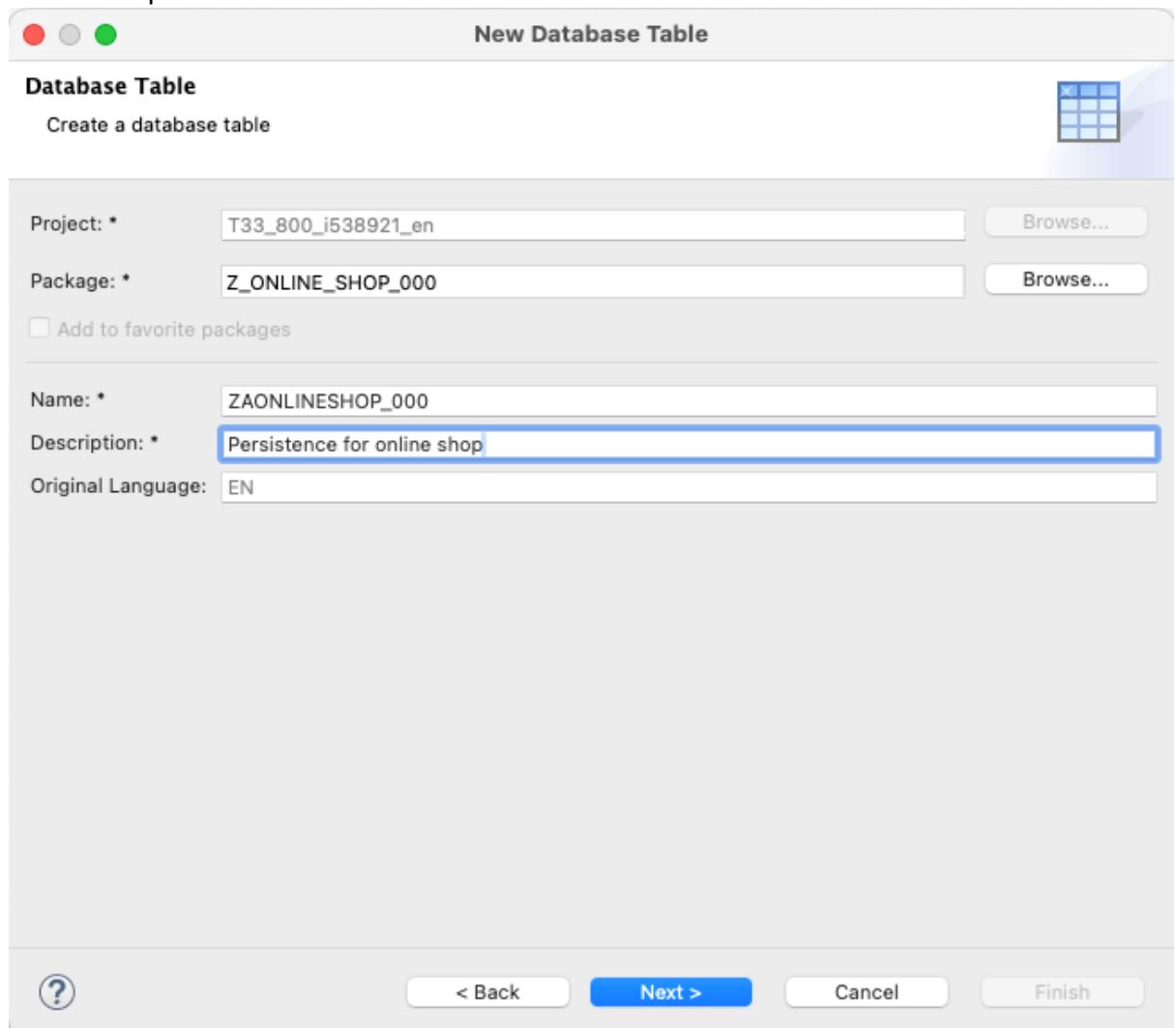
Lets create a new database table

- Right click on your ABAP package and select “New” > “Other ABAP Repository Object”



- Search for “Database Table”, select it and click “Next”.

8. Give the name “ZAONLINESHOP_000” and the description “Persistence for online shop”



9. Select “Next”
 10. Select the transport request and click “Finish”
 11. Replace the default code with the code snippet provided below

```

@EndUserText.label : 'Persistence for online shop'
@AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
@AbapCatalog.tableCategory : #TRANSPARENT
@AbapCatalog.deliveryClass : #A
@AbapCatalog.dataMaintenance : #RESTRICTED
define table zaonlineshop_000 {

  key client : abap.clnt not null;
  key order_uuid : sysuuid_x16 not null;
  order_id : abap.char(10) not null;
  ordereditem : abap.char(10) not null;
  deliverydate : abap.dats;
  creationdate : abap.dats;
  local_last_changed : abp_locinst_lastchange_tstmpl;
  last_changed : abp_lastchange_tstmpl;

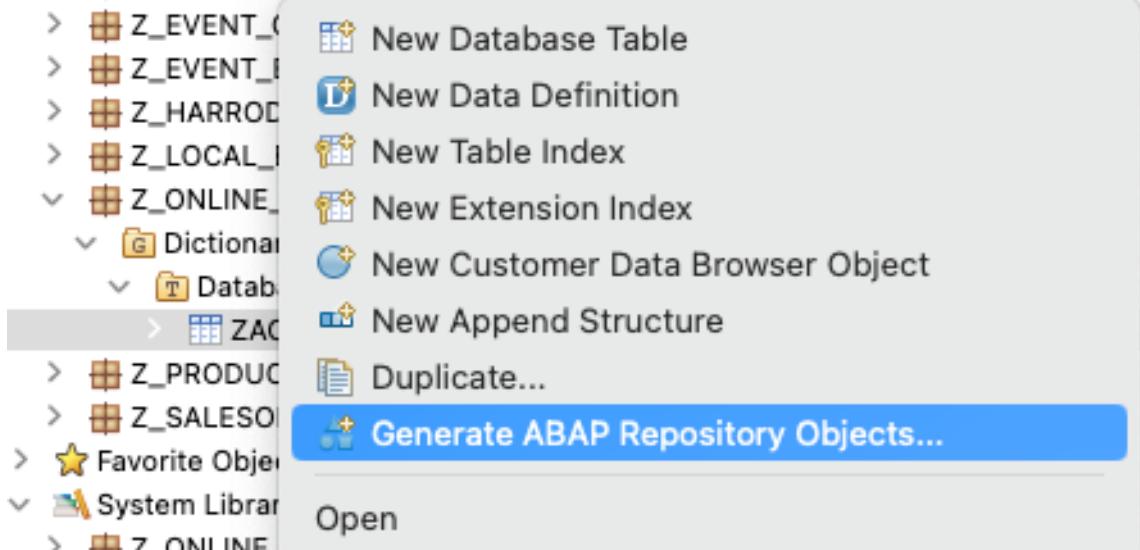
```

}

12. Save and activate the changes

Next lets generate the transactional UI services

13. Right-click your database table and select “Generate ABAP Repository Objects.”



14. Select “Odata UI Service” and hit “Next”

The screenshot shows the SAP Fiori Launchpad interface for selecting a generator. At the top, there's a header with the SAP logo and a search bar labeled "Generate multiple repository objects". Below the header, there are fields for "Project:" (set to "T33_800_i538921_en") and "Referenced Object:" (set to "ZAONLINESHOP_000"). A "Browse..." button is also present. On the left, a tree view shows categories like "ABAP RESTful Application Programming Model" and "Business Configuration Management". Under "ABAP RESTful Application Programming Model", "OData UI Service" is selected and highlighted in blue. To the right of the tree, there's a detailed description of the "OData UI Service". It explains that an OData UI service makes it possible to consume a RAP service with a Fiori Elements UI or other UI clients. It creates all UI-specific information that is annotated in the RAP artifacts in the OData service metadata. A UI service can be previewed with the Fiori Elements App Preview in the service binding artifact. The description continues to say that this generator creates all the development objects that are relevant for a business object and for a RAP service on the basis of one database table. The end result is a full-blown RAP UI service. It lists the objects created: CDS view entity, CDS behavior definition, ABAP implementation class, CDS projection view entity, CDS projection behavior definition, Draft database table, Metadata extension, CDS service definition, and Service binding. At the bottom of the screen, there are navigation buttons: a question mark icon, "< Back", "Next >" (which is highlighted in blue), "Cancel", and "Finish".

15. Enter the previously create package “Z_ONLINE_SHOP_000” and hit “Next”

16. Navigate through the wizard tree and maintain the artefact names provided in the table below, and press “Next”.

RAP Layer	Artefacts	Artifact Names
Busienss Object		
	Data Model	Data Definition Name: ZR_ONLINESHOP_000
		Alias Name: OnlineShop
	Behavior	Implementation Class: ZBP_R_ONLINESHOP_000

		Draft Table Name: ZDONLINESHOP_000
Service Projection (BO Projection)		Name: ZC_ONLINESHOP_000
Business Services		
	Service Definition	Name: ZUI_ONLINESHOP_000
	Service Binding	Name: ZUI_ONLINESHOP_04_000
		Binding Type: OData V4 - UI

17. Press “Finish”

With the RAP objects being created we can preview the application

18. On your service binding ZUI_ONLINESHOP_04_000 hit “Publish”

The screenshot shows the SAP Fiori Launchpad with the title "Service Binding: ZUI_ONLINESHOP_000_O4". The "General Information" section indicates a "Binding Type: OData V4 - UI". The "Services" section shows a table with one row: "Ser Version" (1.0.0) and "AF Service Definition" (Nc ZUI_ONLINESHOP_000). The "Service Version Details" section notes that the local service endpoint is not published and provides a link to "Publish local service endpoint". A red box highlights the "Publish" button in the top right corner of the main content area.

19. Double-click on the entity “OnlineShop” in the “Entity Set and Association” section to open the Fiori elements App Preview.

20. Use your S/4 login credentials to authenticate

The screenshot shows the SAP Fiori Launchpad with the title "Standard". The "Entity Set and Association" section displays a table titled "OnlineShops" with columns: OrderID, Ordereditem, Deliverydate, and Creationdate. A message at the bottom of the table says "No data found. Try adjusting the search or filter parameters." The top right of the screen shows standard SAP Fiori navigation icons.

With the application itself being created lets define an event “ItemIsOrdered” that is triggered when ever a new order is placed. The event should always contain the name of the ordered item.

21. Right click your package Z_ONLINE_SHOP_000 and select “New” > “Other ABAP Repository Object”

22. Search for “Data Definition” and select “Next”
23. Give the name “Z_ITEM_IS_ORDERED” and the description “Payload for ItemIsOrdered event” and hit “Next”

New Data Definition

Data Definition

Create a data definition

D

Project: *	T33_800_i538921_en	Browse...
Package: *	Z_ONLINE_SHOP_000	Browse...
<input type="checkbox"/> Add to favorite packages		
Name: *	Z_ITEM_IS_ORDERED	
Description: *	Payload for ItemIsOrdered event	
Original Language:	EN	
Referenced Object:	Browse...	

?

< Back

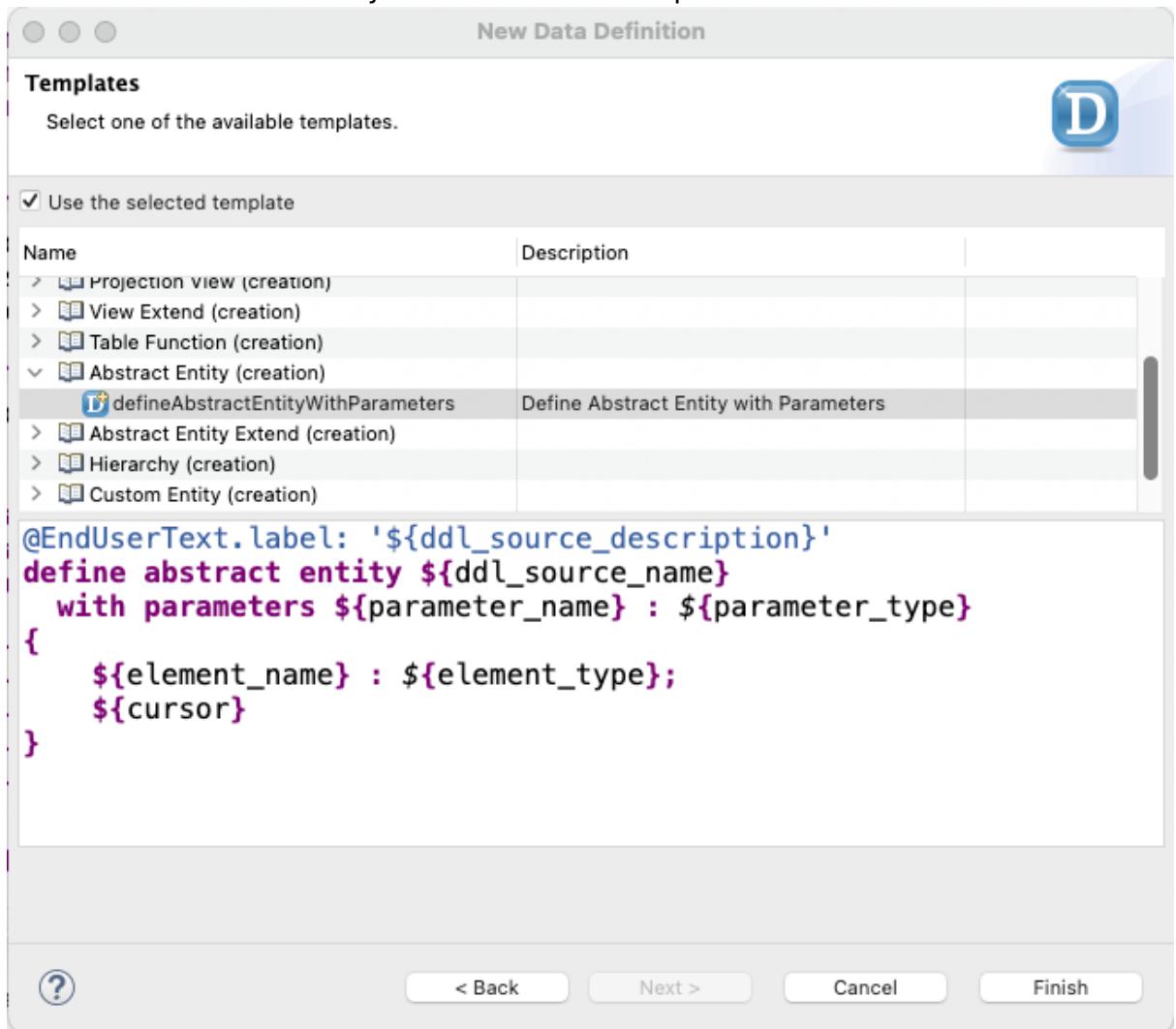
Next >

Cancel

Finish

24. Select a transport request and press “Next”

25. Select “defineAbstractEntityWithParameters” and press “Finish”



26. Replace the code with the below provided code snippet

```
@EndUserText.label: 'Payload for ItemIsOrdered event'
define root abstract entity Z_ITEM_IS_ORDERED
{
  ItemName : abap.char(25);
}
```

27. Save and activate the abstract entity

28. Open the behavior definition ZR_ONLINESHOP_000

29. Enter the line “event ItemIsOrdered parameter Z_ITEM_IS_ORDERED”

```
1 managed implementation in class ZBP_R_ONLINESHOP_000 unique;
2 strict ( 2 );
3 with draft;
4
5 define behavior for ZR_ONLINESHOP_000 alias OnlineShop
6 persistent table zaonlineshop_000
7 draft table ZDONLINESHOP_000
8 etag master LocalLastChanged
9 lock master total etag LastChanged
10 authorization master( global )
11
12 {
13     field ( readonly )
14         OrderUUID,
15         LastChanged,
16         LocalLastChanged;
17
18     field ( numbering : managed )
19         OrderUUID;
20
21
22     create;
23     update;
24     delete;
25
26     event ItemIsOrdered parameter Z_ITEM_IS_ORDERED;
27
28     draft action Edit;
29     draft action Activate optimized;
30     draft action Discard;
31     draft action Resume;
32     draft determine action Prepare;
33
```

With that the data event is defined and can be used. So lets trigger it when ever a new order is created.

30. Add the keywords “with additional save” to the first line of the behavior definition

ZR_ONLINESHOP_000

```
B [T33] ZR_ONLINESHOP_000 X
1 managed with additional save implementation in class ZBP_R_ONLINESHOP_000 unique;
2 strict ( 2 );
3 with draft;
4
5 define behavior for ZR_ONLINESHOP_000 alias OnlineShop
6 persistent table zaonlineshop_000
7 draft table ZDONLINESHOP_000
8 etag master LocalLastChanged
9 lock master total etag LastChanged
10 authorization master( global )
11
12 {
13   field ( readonly )
14   OrderUUID,
15   LastChanged,
16   LocalLastChanged;
17
18   field ( numbering : managed )
19   OrderUUID;
20
21
22   create;
23   update;
24   delete;
25
26   event ItemIsOrdered parameter Z_ITEM_IS_ORDERED;
27
28   draft action Edit;
29   draft action Activate optimized;
30   draft action Discard;
31   draft action Resume;
32   draft determine action Prepare;
33
```

31. Save and activate the behavior definition

32. Place your cursor on the keyword additional (where the warning is displayed) and press

Ctrl + 1 and double click the provided quick assist. This should add the save_modified method to the local implementation class.

```
B [T33] ZR_ONLINESHOP_000 X C [T33] ZBP_R_ONLINESHOP_000
1 managed with additional save implementation in class ZBP_R_ONLINESHOP_000 unique;
2 strict ( 2 );
3 with draft;
4
5 define behavior for
6 persistent table zao
7 draft table ZDONLINE
8 etag master LocalLas
9 lock master total et
10 authorization master
11
12 {
13   field ( readonly )
14   OrderUUID,
15   LastChanged,
16   LocalLastChanged;
17
18   field ( numbering : managed )
```

33. Within the local types of the ABAP Class ZBP_R_ONLINESHOP_000 replace the implementation of the method “save_modified” with the following snippet

METHOD save_modified.

IF create-onlineshop IS NOT INITIAL.

```

RAISE ENTITY EVENT ZR_ONLINESHOP_000~ItemIsOrdered
FROM VALUE #( FOR online_shop IN create-onlineshop ( %key      = online_shop-
%key
                %param-ItemName = online_shop-Ordereditem
            )).
ENDIF.
ENDMETHOD.

```

34. The code should look something like this

```

1 CLASS lsc_zr_onlineshop_000 DEFINITION INHERITING FROM cl_abap_behavior_saver.
2   PROTECTED SECTION.
3     METHODS save_modified REDEFINITION.
4   ENDCLASS.
5
6 CLASS lsc_zr_onlineshop_000 IMPLEMENTATION.
7
8   METHOD save_modified.
9     IF create-onlineshop IS NOT INITIAL.
10       RAISE ENTITY EVENT ZR_ONLINESHOP_000~ItemIsOrdered
11         FROM VALUE #( FOR online_shop IN create-onlineshop ( %key      = online_shop-%key
12           %param-ItemName = online_shop-Ordereditem
13         )).
14       ENDIF.
15     ENDMETHOD.
16
17   ENDCLASS.
18
19 CLASS LHC_ONLINESHOP DEFINITION INHERITING FROM CL_ABAP_BEHAVIOR_HANDLER.
20   PRIVATE SECTION.
21     METHODS:
22       GET_GLOBAL_AUTHORIZATIONS FOR GLOBAL AUTHORIZATION
23         IMPORTING
24
25
26

```

35. Save and activate

With that the event is triggered whenever a new order is placed. However it is not yet sent to the Advanced Event Mesh. For that we need to create an event binding and a topic binding on the AEM channel.

36. Right click your ABAP package and select “New” > “Other ABAP Repository Object”
37. Search for “Event Binding”, select it and press “Next”

38. Give the name “Z_EB_ONLINE_SHOP” and the description “Event Binding for Online Shop ItemIsOrdered”

Event Binding

Create Event Binding

Project: * T33_800_i538921_en Browse...

Package: * Z_ONLINE_SHOP_000 Browse...

Add to favorite packages

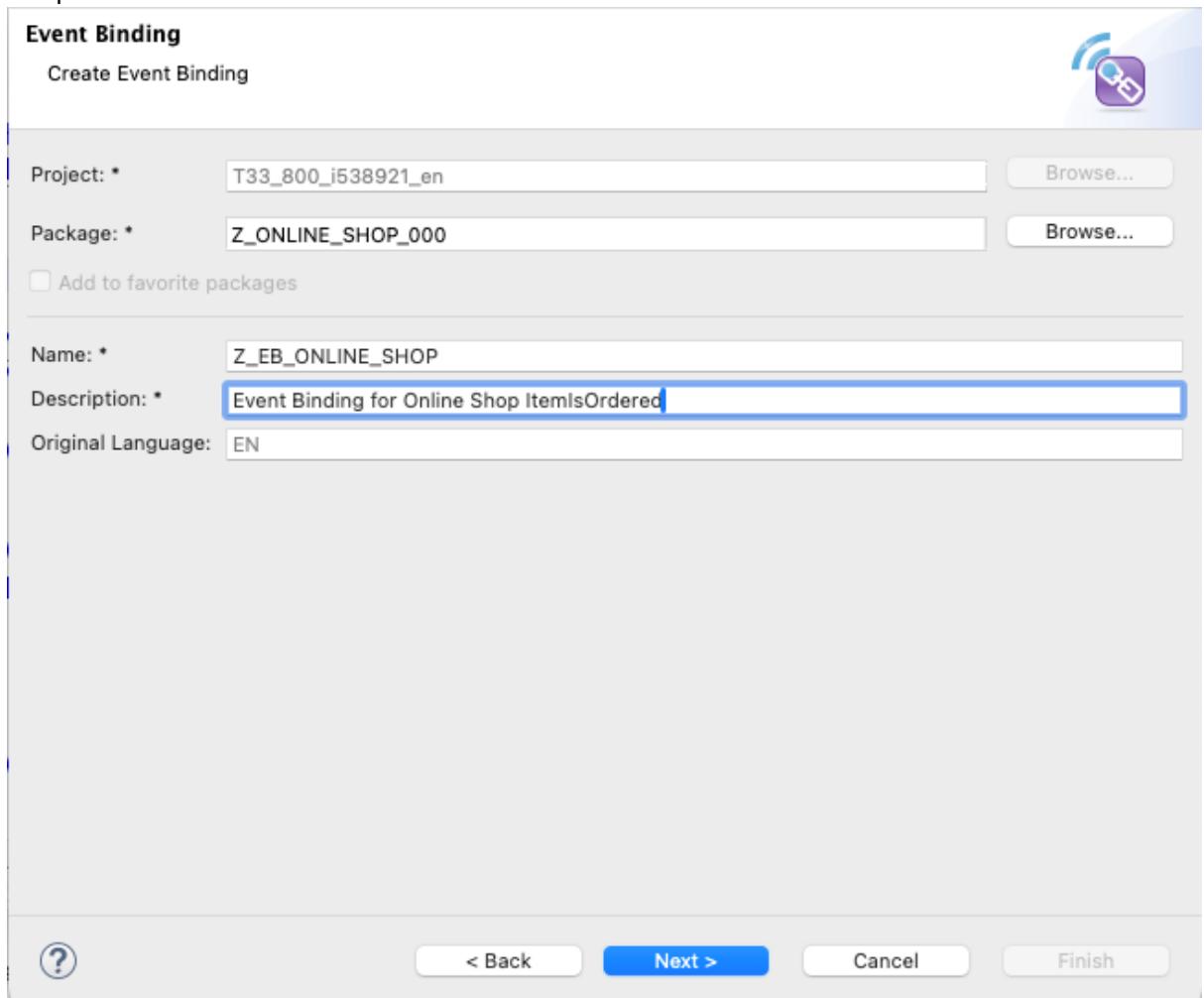
Name: * Z_EB_ONLINE_SHOP

Description: * Event Binding for Online Shop ItemIsOrdered

Original Language: EN

?

< Back Next > Cancel Finish



39. Select your transport request and press “Finish”
40. In the event binding you can define the topic und which the event is later published. Enter the following values:
- Type Namespace: zcustom
SAP Object Type: OnlineShop
Operation: created
41. Press “Add..”

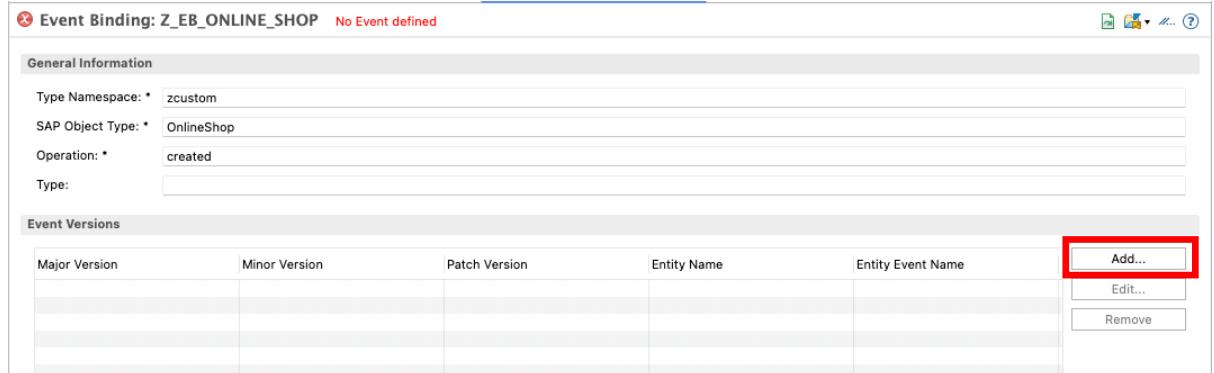
Event Binding: Z_EB_ONLINE_SHOP No Event defined

General Information

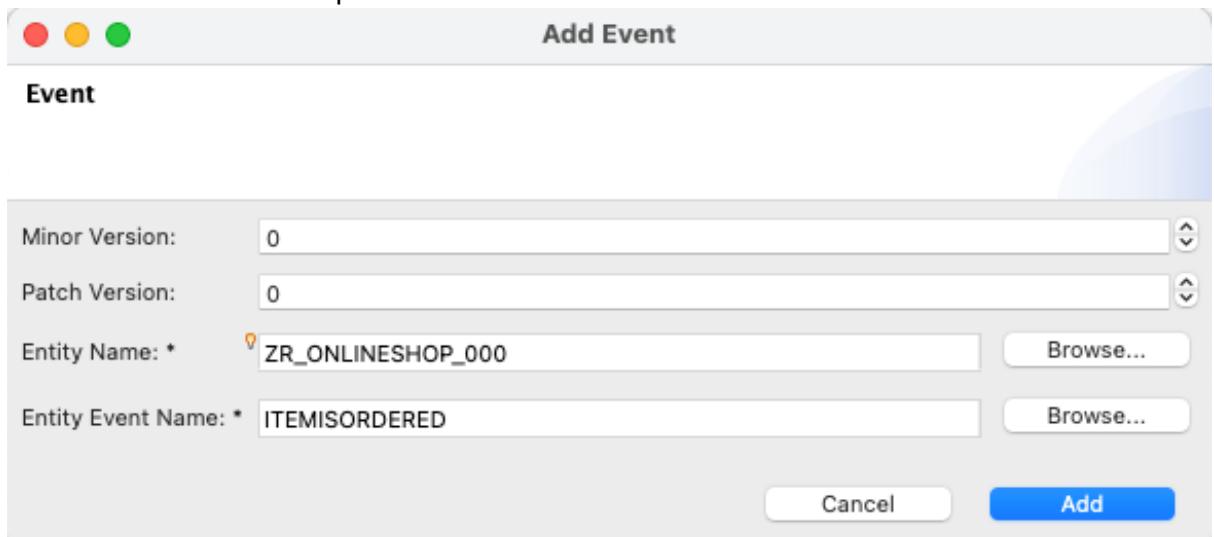
Type Namespace: * zcustom
SAP Object Type: * OnlineShop
Operation: * created
Type:

Event Versions

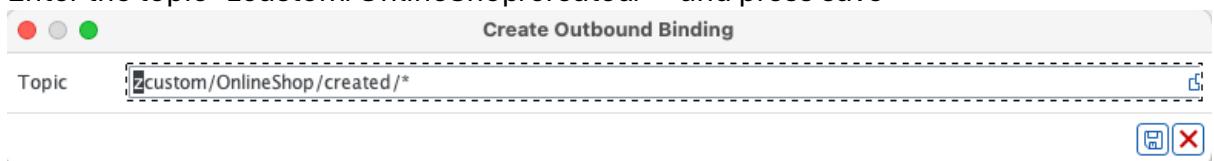
Major Version	Minor Version	Patch Version	Entity Name	Entity Event Name	Add...
					Add...
					Edit...
					Remove



42. Enter the Entity Name “ZR_ONLINESHOP_000” and the Entity Event Name “ITEMISORDERED” and press “Add”

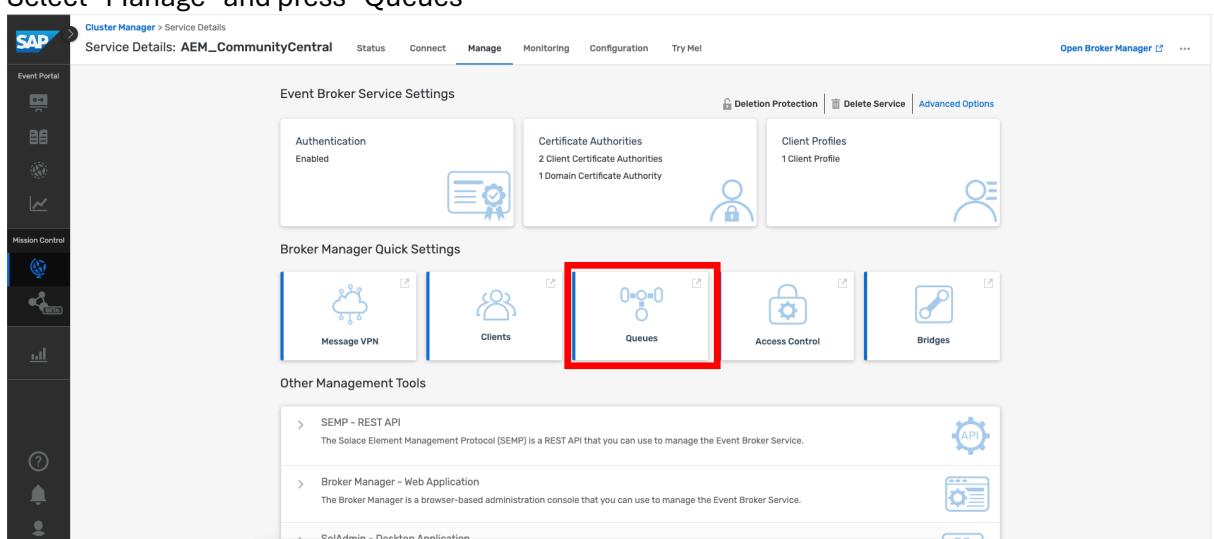


43. Activate the event binding.
 44. Next go into transaction “/IWXB/E/OUTBOUND_CFG” and select the channel to your AEM that has been previously set up
 45. Press “Create new topic binding”
 46. Enter the topic “zcustom/OnlineShop/created/*” and press save



With that the event is automatically send to the AEM where it can be consumed.

47. Go into the AEM UI
 48. Under Cluster Manager select your broker
 49. Select “Manage” and press “Queues”



50. Create a new queue and give it the name “Online Shop”
 51. Go into the queue and navigate to “Subscriptions”

52. Add a new subscription with the name
“S/4HANA/Events/ce/zcustom/OnlineShop/created/v1”

Create Subscription

1 Subscriptions

X S/4HANA/Events/ce/zcustom/OnlineShop/created/v1

Add new subscription by pressing Enter in topic field or clicking Add New

Cancel Create

With that the event `ItemsOrdered` will be triggered when ever a new order is created and send to the AEM where it is consumed by the created queue.

Let's test that.

53. Within eclipse open the service binding “ZUI_ONLINESHOP_04_000 and double click the entity “OnlineShop” within the Entity Set and Association section. This should open the Fiori Preview.
54. If required log in with your S/4 credentials
55. Press create in the Fiori application

Standard ▾

Editing Status:
All

Go Adapt Filters (1)

OnlineShops

OrderID	Ordereditem	Deliverydate	Creationdate
To start, set the relevant filters and choose "Go".			

Create Delete |

56. Give the OrderID “1” and the Ordereditem “Item 1” and press “Create”

The screenshot shows the SAP Fiori OnlineShop application interface. At the top, there are four input fields: 'OrderID' (containing '1'), 'Ordereditem' (containing 'Item 1'), 'Deliverydate' (set to 'e.g. Dec 31, 2024'), and 'Creationdate' (set to 'e.g. Dec 31, 2024'). Below these fields is a large empty area for the order details. At the bottom right of the screen, there are three buttons: 'Draft updated' (disabled), 'Create' (highlighted in blue), and 'Discard Draft'.

57. With that there should be a messaged queued within the AEM

The screenshot shows the SAP AEM Community Central Queues interface. The left sidebar includes 'AEM_CommunityCentral', 'Change VPN', 'Messaging' (selected), 'Clients', 'Queues' (highlighted in blue), 'Connector Wizards', and 'Access Control'. The main area is titled 'Queues | Online Shop' and shows a table for 'Messages Queued'. The table has columns: Message ID, Spooled Time, Content Size (B), Attachment Size (B), Priority, Undelivered, Redeliveries, DMO Eligible, and Replication State. One message is listed: '55245' with a spooled time of '2024-09-22 23:04:35', content size of '0', attachment size of '528', priority 'Yes', undelivered '0', redeliveries 'Yes', DMO Eligible 'Yes', and replication state 'Not replicated'. There is also a 'Message ID' column header with a dropdown arrow.

Message ID	Spooled Time	Content Size (B)	Attachment Size (B)	Priority	Undelivered	Redeliveries	DMO Eligible	Replication State
55245	2024-09-22 23:04:35	0	528	Yes	0	Yes	Yes	Not replicated

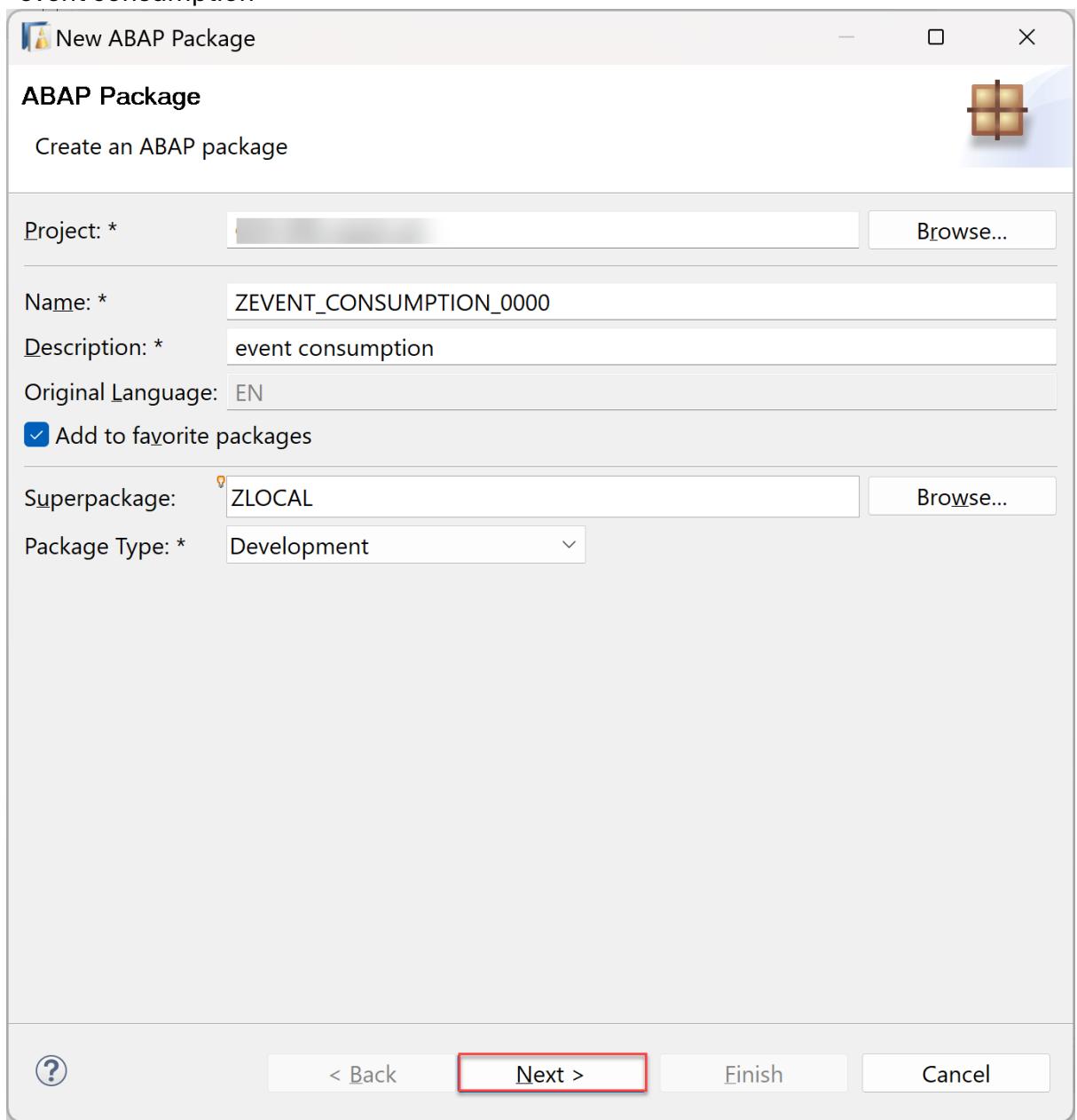
Receive events with S/4HANA or S/4HANA Cloud

Besides sending events to the Advanced Event Mesh it is also possible to receive events from it. To demonstrate that we will configure the system to receive the Business Partner Changed event that is send to the AEM as per the first step. For that we'll be creating a consumption model in ABAP Development Tools, configure an inbound binding and configure a subscription.

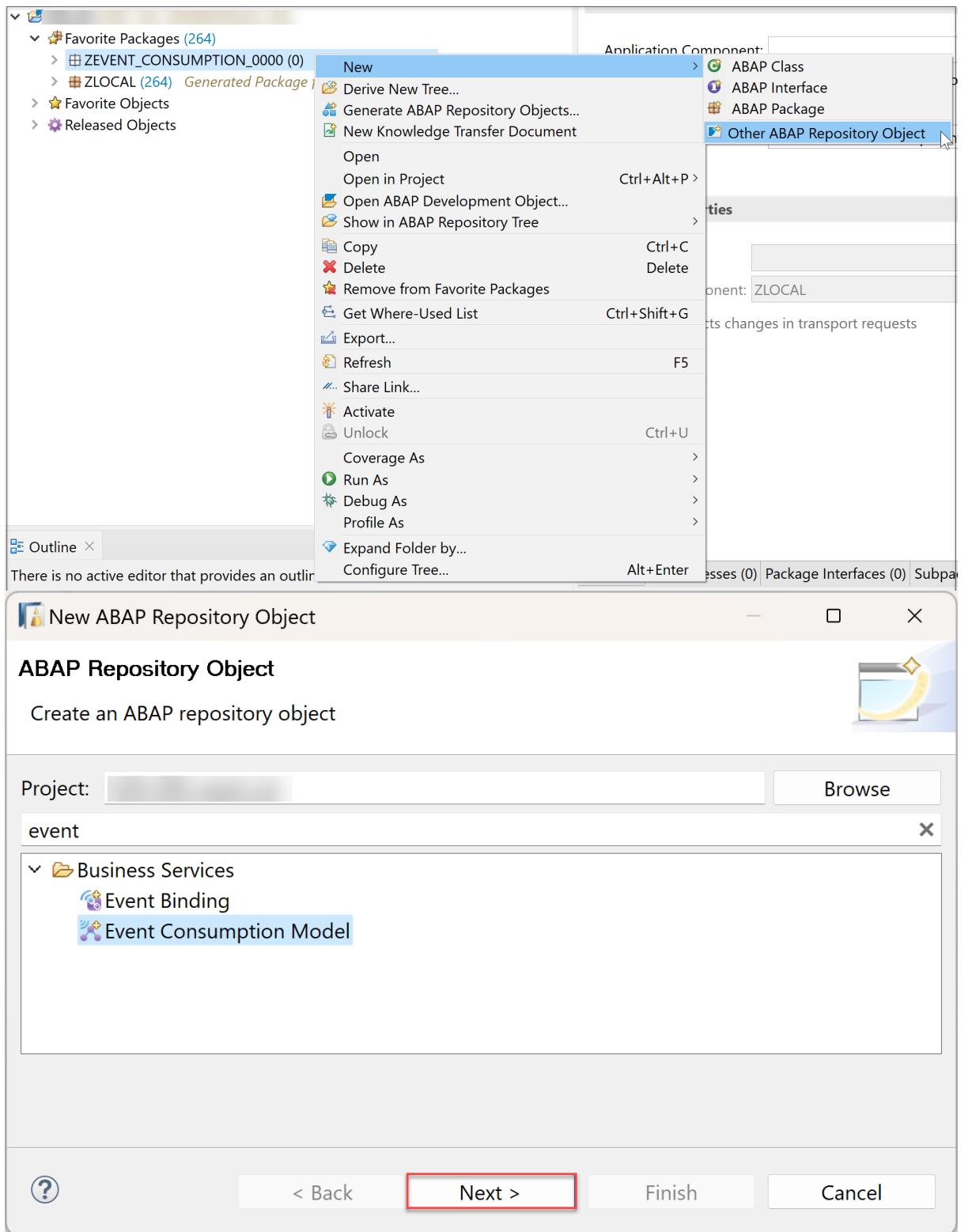
1. Go to the event specification of [Business Partner Events](#) within the Business Accelerator Hub and download the JSON file.

The screenshot shows the SAP Business Accelerator Hub interface. At the top, there's a navigation bar with links for Explore, Resources, Discover Integrations, and Partner with Us. Below the navigation is a blue header bar with the title "Business Partner Events" and a subtitle "Informs a remote system about created and changed business partners in an SAP S/4HANA system". The main content area has tabs for Overview, Event References, Event Consumption, and Documents. The Overview tab is selected. On the left, there's an "Introduction" section with a brief description of what a business partner is and a list of available events: "Business partner changed" and "Business partner created". To the right, there's a summary card with details like Status (ACTIVE), Version (1.0.0), Last Modified (28 Feb 2024), and Type (EVENT). Below the introduction, there's a "Event Resources" section with a "Event Specifications" heading. Under this heading, there are two options: "JSON" and "YAML", each with a download icon. A red box highlights the "JSON" option. At the bottom right of this section is a "Chat (Beta)" button. A vertical "FEEDBACK" button is located on the far right edge of the page.

2. Create a new ABAP Package “ZEVENT_CONSUMPTION_0000” with description “event consumption”



3. Right-click your package and choose **New > Other ABAP Repository Object > Business Services > Event Consumption Model** and click **Next** to launch the creation wizard.



4. Fill the fields and upload the JSON Event Specification file you saved before.
Description: Event consumption model

Namespace/Prefix/Identifier: “Z” and “BP_Events_0000”

New Event Consumption Model

Event Consumption Model

Create Event Consumption Model

Project: * T33_800_i538921_en

Package: * Z_EVENT_CONSUMPTION

Add to favorite packages

Name: ZBPEVENTS0000 0001

Description: * Event Consumption Model

Original Language: EN

Namespace / Prefix / Identifier: *

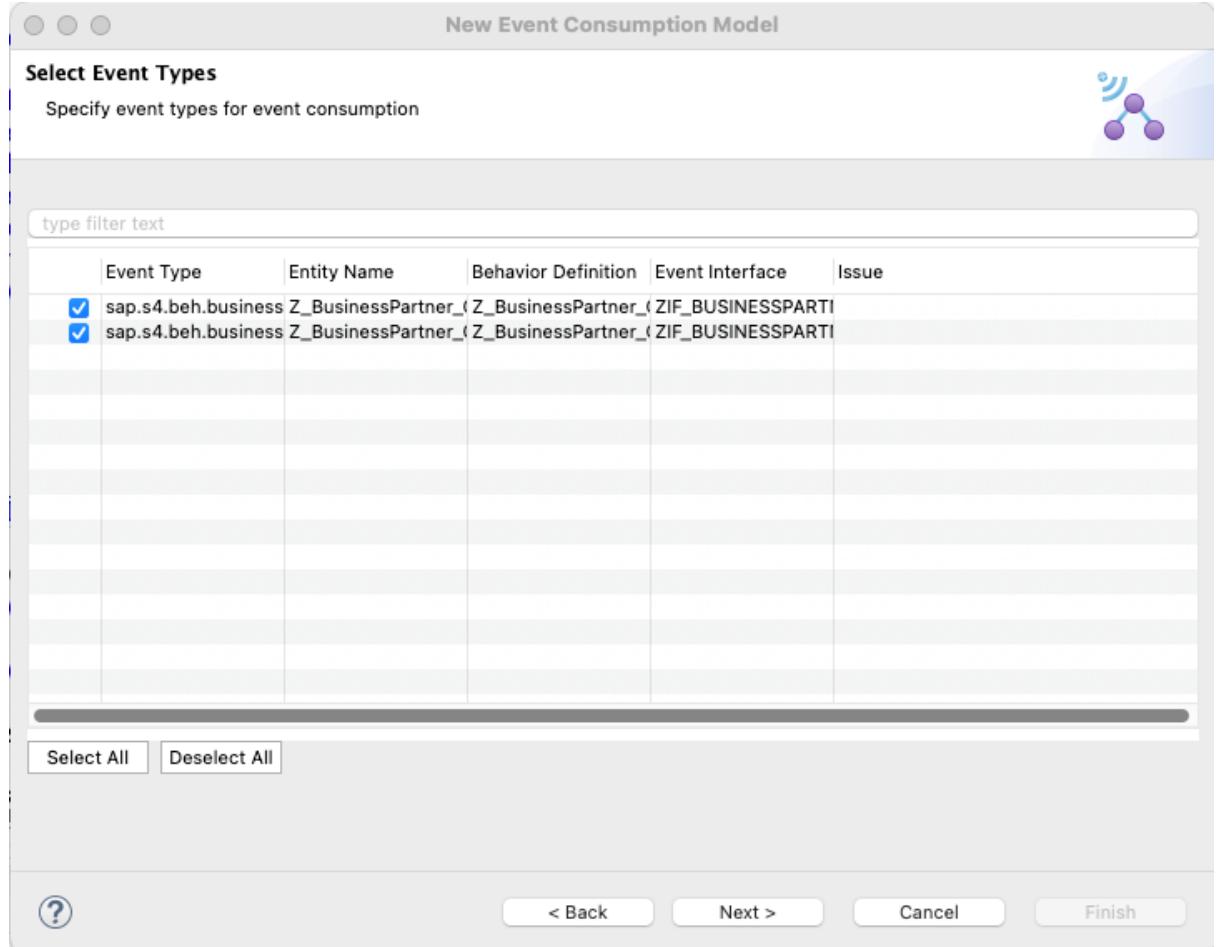
<No namespace in package> / /

Event Specification File: *

Namespace and Prefix for objects to be generated:

Z

5. Select both events and press “Next”



6. Press “Next” to the consumer Artifacts and “Next” to the ABAP Artifact Generation List
7. Select a Transport Request and click “Finish”
8. Save and activate your event consumption model

With the event consumption model created we can go into SAP GUI and configure the inbound binding

9. Open transaction /IWXB/E/INBOUND_CFG

10. Select your AEM channel and press “Create new topic binding”

The screenshot shows the SAP Inbound Binding Configuration interface. The title bar reads "SAP Inbound Binding Configuration". The left sidebar lists "Channels" under "Active Channels" and "Inactive Channels". Under "Active Channels", there is a section for "Default Service Plan" containing items like "COE_EMEA_TECH", "COMMUNITY_EM_1585208", "EM_CHANNEL_DEMO", "IS53641_EM", "IB23_D073297_EMCHANNEL", "IB24_J745704_EMCHANNEL", and "Advanced Service Plan" which includes "AEM_BROKER", "AEM_COE_EMEA_INT", "AEM_COMMUNITY", and "IO54810_AEM". The main area is titled "Inbound Bindings of Channel AEM_BROKER" and contains tabs for "Status" and "Maintained by Topic". Below these tabs, there is a section for "Consumers" with similar tabs for "Status" and "Repository". The top toolbar has several icons, and a red box highlights the "Create New Topic Binding" icon.

11. Enter “sap/s4/beh/businesspartner/v1/BusinessPartner/Changed/v1” or use the search help

The screenshot shows the "Create Inbound Binding" dialog. The title bar says "Create Inbound Binding". The "Topic" field contains the value "sap/s4/beh/businesspartner/v1/BusinessPartner/Changed/v1". On the right side, there are two blue buttons: one with a right arrow and one with a red X.

12. Switch to the next page

13. Choose your consumer, “ZBPEVENTS0000” and click “Create Destination”

The screenshot shows the "Create Inbound Binding" dialog again. The title bar says "Create Inbound Binding". The "Topic" field contains the value "sap/s4/beh/businesspartner/v1/BusinessPartner/Changed/v1". Below it, a section titled "Consumers to be generated" shows a table with two rows:

Val...	Consumer	Versi...	Destinat...
	ZBPEVENTS0000	1	
	ZEVENTSBUSINESSPARTNER	1	

A red box highlights the "Create Destination" button in the toolbar above the table. Both rows in the table are also highlighted with red boxes.

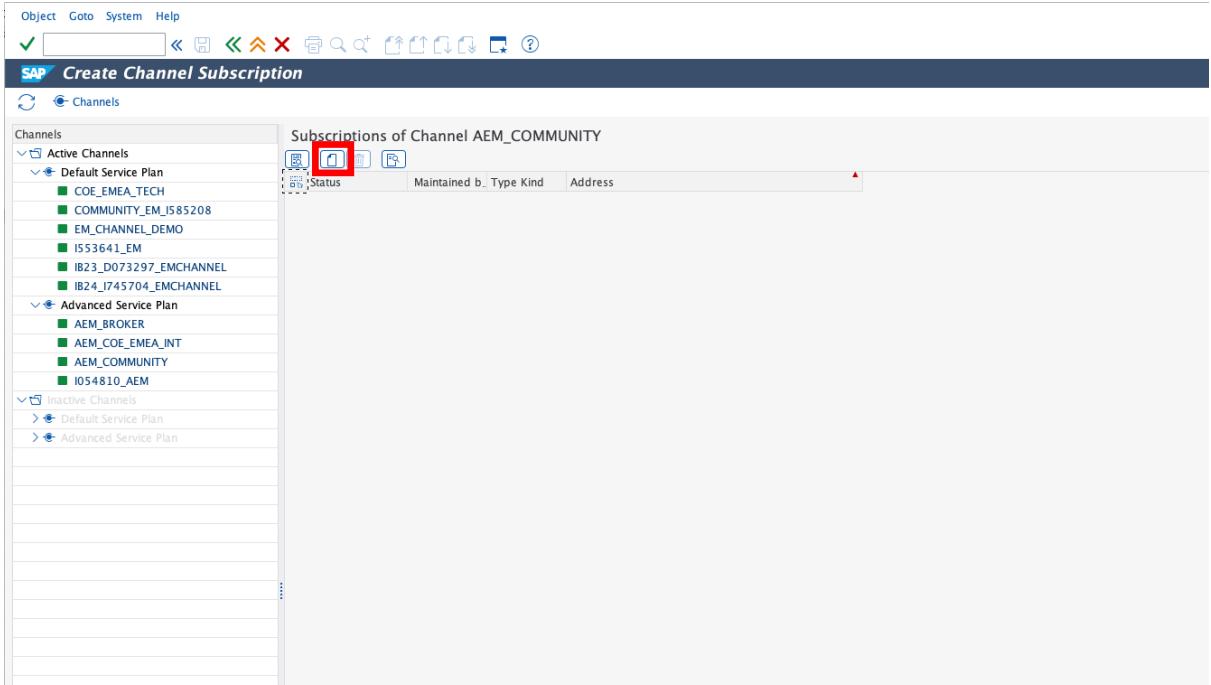
14. Enter your user name in the Create Destination dialog and press save

15. Press save on the Create Inbound Binding dialog

With the Inbound Binding configured we can configure the subscription next

16. Open transaction “/IWXB/E/SUBSCRIPTION”

17. Click “Create new subscription”



18. Enter the name of the queue you created. If you followed the steps of this tutorial it should be “Business_Partner”.

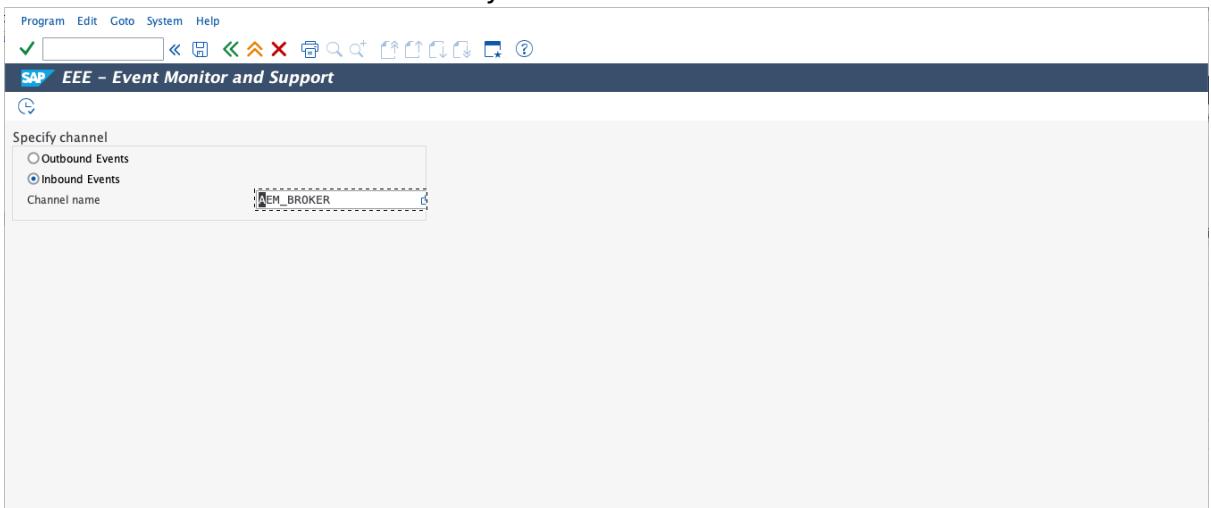


19. Save the subscription

With the subscription being configured events that are send to the AEM queue should be consumed by the S/4 system. Let's test that

20. Go into transaction BP, select a business partner, edit it and save the changes. If the outbound binding and the queue has been properly configured as shown in this tutorial, this should trigger a Business Partner Changed event that should be send to the AEM. As a subscriber to the queue the event should then be received again by the S/4 system.
21. To check if the event has been properly received open transaction /IWXBE/EVENT_MONITOR.

22. Select Inbound Events and choose your AEM channel



23. Choose execute and view the received events

1 Inbound Events of Channel AEM_BROKER -- Received										
		Received <-> Consumed	Event ID	Topic	Arrival Time	Payload Size [k...]	Status	Retries	Repository ID	Consumer ID
			FA163E75AA8A1EDF9E9954D67F45B...	sap/s4/beh/businesspartner/v1/Busine...	22.09.2024 10:08...	0,27	In Process	1	DEFAULT	ZBPEVENTS0000