AEM Roadshow LATAM (Sep 2025)

Unleashing the power of “Events”
With Event-Driven Integrations

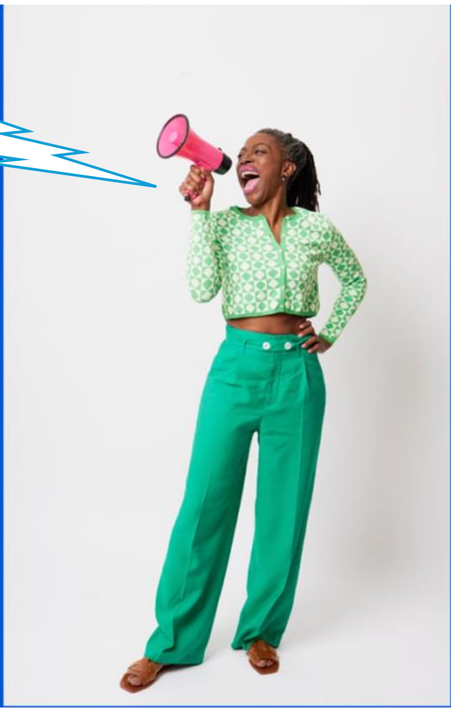


Table of Contents

1 EXECUTIVE SUMMARY 3

THE CHALLENGE/OPPORTUNITY 3
THE BUSINESS NEED? 3
TODAY'S WORKSHOP 3
WHAT YOU'LL WALK AWAY WITH 3

2 KICK OFF & ENVIRONMENT SETUP 3

INTRODUCTION 3
ENVIRONMENT ACCESS/SETUP 4

3 FRAMING THE FIRST BUSINESS SCENARIO 4

BUSINESS REQUIREMENT: 4
HOW WOULD YOU APPROACH THIS? 4
CHALLENGES WITH THIS APPROACH 5

4 INTRODUCING THE BETTER WAY 5

UNDERSTANDING EVENTS & TOPICS 5
UNDERSTANDING AEM QUEUES AND TOPIC TO QUEUE MAPPING 6
HANDS-ON WARM-UP: PLAYING WITH AEM TOPICS & QUEUES 7
THE MODERN APPROACH TO INTEGRATION IS A 3-STEP APPROACH 10
SCENARIO 1: SALESFORCE INTEGRATION (HANDS-ON) 10

5 FRAMING THE 2ND BUSINESS CASE SCENARIO 16

BUSINESS REQUIREMENT: 16
HOW WOULD YOU APPROACH THIS TODAY? 16
CHALLENGES AMPLIFIED: 16

6 THE MODERN APPROACH – EVENT-DRIVEN ARCHITECTURE (EDA) – SEAMLESS EXPANSION 16

SIMPLY "INTEGRATE" 16
BENEFITS REALIZED: 17
SCENARIO 2 – COUPA INTEGRATION HANDS-ON 17
STEP 1: CREATE REQUIRED QUEUES AND SUBSCRIPTIONS FOR COUPA IFLOW 17
STEP 2: CONFIGURE AND DEPLOY BUSINESSPARTNERCHANGE_AEM_COUPA IFLOW 17

7 DEMOS – INTEGRATIONS BEYOND IFLOWS: IF TIME PERMITS 18

8 OPTIONAL: UNDERSTANDING RETRY, ERROR HANDLING, UNBLOCKING MESSAGE PROCESSING – IF TIME PERMITS 18

ADAPTER SETTINGS 18
QUEUE CONFIGURATION FOR RETRY HANDLING 19

9 APPENDIX – CONNECTING TO THE TRY ME UTILITY 22

OBTAIN CONNECTION CREDENTIALS 22

1 Executive Summary

The Challenge/Opportunity

- Most enterprises today rely on **point-to-point (P2P) integrations**.
- These connections work, but:
  - They are hard to extend when new systems are added.
  - They introduce latency and fragility.
  - Every new requirement means rework.

The Business Need?

- Enterprises want integrations that are:
  - **Faster** – real-time, not delayed.
  - **Scalable** – easy to expand to new systems.
  - **Resilient** – no data loss when one system is down.
  - **Future-ready** – supporting cloud, SaaS, and on-prem in one landscape

Today's Workshop

We'll work through real business scenarios together.

- First, you'll explore how integrations are typically built today.
- Then, we'll look at the challenges that approach creates.
- Finally, we'll uncover a modern pattern that changes the game.

What You'll Walk Away With

- A deeper understanding of why enterprises are shifting away from P2P.
- Hands-on experience building **lighter, more flexible integrations**.
- A clear view of how this approach prepares your landscape for the future.

The journey starts with a familiar scenario — and together we'll discover a better way.

2 Kick off & Environment Setup

Introduction

The objective of this hands-on workshop is to guide you through a series of exercises designed to introduce the pattern of modern integration using the **"Micro Integration"** concept. These concepts can be adopted by existing SAP Integration Suite, Cloud Integration customers to extend and complement their current integration approach.

We'll begin with a look at **conventional integration design (point-to-point using iFlows)** — the way many projects have been done in the past. From there, we'll explore **why and how organizations are transitioning from legacy point-to-point integrations to a modern, event-driven approach**.

Through this journey, you'll see how the **EDA micro-integration pattern** can deliver greater agility, scalability, and decoupling — enabling you to connect systems faster and more reliably, without rework or risk to existing integrations.

We'll simulate real business requests and solve them step by step. Along the way, you'll gain hands-on experience with both Integration Suite and Advanced Event Mesh.

Next, before diving into the scenarios, we'll cover what your **environment setup** looks like.

Environment Access/Setup

2.1.1 Tools Required

- **Integration Suite** – You are expected to use your own tenant. You will need this to build iFlows.
- **Advanced Event Mesh Environment** – You are expected to use your own tenant. You will need this to showcase the future of enterprise integrations and the art of possible.
- **Solace Event Feeds** – You can access this here. You will need this to mock S4/ECC systems to stream Business Partner events on your AEM environment.
- **Mock APIs** – Following APIs will be required during the workshop. You will need this to mimic connecting to SaaS platforms like Salesforce, Coupa etc. *The URLs provided below are for information purpose only and have already been configured in the iFlows provided as a package, that will be downloading and importing in subsequent section.*
  https://aemworkshop.free.beeceptor.com/aem/2/sf
  https://aemworkshop.free.beeceptor.com/aem/2/coupa

2.1.2 Package Required

- AEM Roadshow LATAM.zip – You may download this now and we will use it later.

3 Framing the First Business Scenario

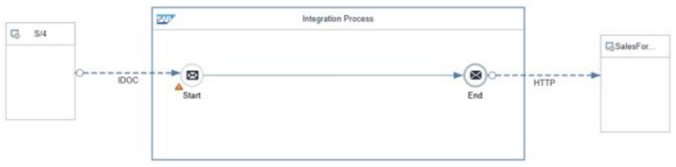
Business Requirement:

Synchronize any updates made to a Business Partner record in SAP S/4HANA with Salesforce (SFDC).

How Would You Approach This?

Most likely, you will take the traditional approach, which is to create P2P integration using an iFlow. The implementation path would look like the following:

1. Use **SAP Integration Suite** to create an **iFlow**.
2. Connect to **SAP S/4HANA** via OData or IDoc adapter to fetch supplier updates.
3. Perform any necessary transformations.
4. Connect to **Salesforce** using the Salesforce adapter or REST API.
5. Push the updated record to Salesforce.



Challenges With This Approach

- Receiver is pre-determined when building the iFlow
- Retry logic will be created manually.
- Difficult to extend when new consumers (e.g., Coupa, Docusign, ServiceNow, data lake, compliance app etc.) need the same event.
- If used, polling mechanisms can create latency and load on the backend.
- If used, scheduled jobs will not deliver the information to the backend in a timely manner leading to out of sync 3rd party systems.

Note: Today however, we are not going to use the P2P approach, instead, we will use event-driven integration approach that will help you build highly decoupled, scalable integrations across hybrid cloud environment.

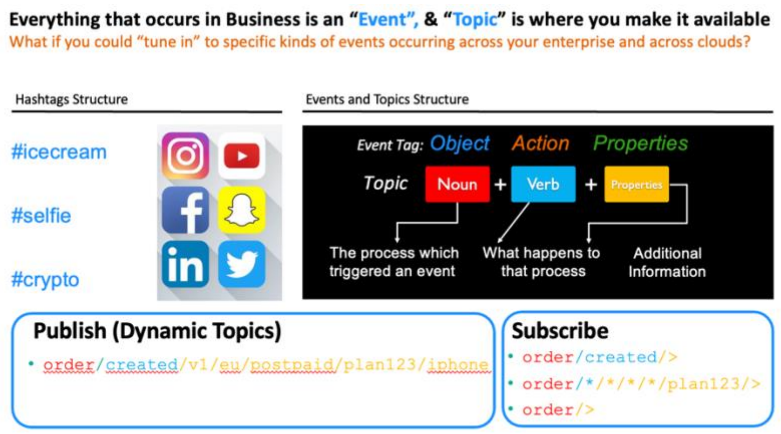
4 Introducing the Better Way

Instead of hardwiring integrations, your SAP ERP systems already natively produce events. So, any changes to the business partner can directly be published on Advanced Event Mesh (AEM) as an event. We will still build an iFlow, but this time it will be like a micro-integration (micro-service) that reacts to an event using "**Topics**" in real-time. With this approach, both S4 Hana and Salesforce systems are **highly decoupled** and doesn't impact each other in the event one of the systems is down. You also get **built-in shock absorber** capability of AEM where sudden deluge of data from S4 Hana will not overwhelm your iFlow. The extra volume of data will simply be buffered on a queue on AEM and AEM will trickle feed records to the iFlow and the speed it can be processed.

Understanding Events & Topics

When we talk about **Events**, think of them as digital signals that something meaningful just happened in your business, like *an order was created*, *a delivery was confirmed*, or *a payment failed*.

Now, every event needs a place to "live" so that others can find it. That's where **Topics** come in. A topic is like the **address or channel** where the event is published. Just as you use hashtags on social media to tag your posts (#icecream, #crypto), we use topics to classify and organize business events.


Everything that occurs in Business is an "Event", & "Topic" is where you make it available
What if you could "tune in" to specific kinds of events occurring across your enterprise and across clouds?
Hashtags Structure
Events and Topics Structure
#icecream
#selfie
#crypto
Event Tag: Object Action Properties
Topic Noun + Verb + Properties
The process which triggered an event
What happens to that process
Additional Information
Publish (Dynamic Topics)
order/created/v1/eu/postpaid/plan123/iphone
Subscribe
order/created/>
order/*/*/*/*/plan123/>
order/>

This approach ensures **loose coupling**: the system creating the event (publisher) does not need to know who is consuming it (subscriber).

Structure of Events & Topics

- **Object (Noun):** The entity involved (e.g., Order, Customer, Invoice).
- **Action (Verb):** The change that occurred (e.g., Created, Updated, Deleted).
- **Properties:** Additional context or attributes (e.g., region, product type, plan ID).

The beauty here is in the **dynamic topic taxonomy**. Instead of publishing to one rigid channel, events can carry rich context directly in the topic path. For example:

A simple event might go to:
order/created
But we can add layers of context dynamically, such as version, region, customer type, or product:
order/**created**/v1/eu/postpaid/plan123/iphone

Why is this powerful?

- A **finance app** might only care about orders which are in created state and are for "EU", it can just subscribe to **order**/**created**/v1/eu/>.
- A **marketing app** might only care about iPhones and subscribe to **order**/*/*/*/*/*/iphone.
- Another system could be broader and simply listen to **order**/**created**/>.

This flexibility means each consumer gets **exactly the events it needs** without the publisher ever having to change its logic or know who the subscribers are.

In short:

- **Publish once** into a structured topic.
- **Let the topic taxonomy do the heavy lifting** so consumers can "tune in" at whatever level of granularity they need.

That's how events and topics work together to enable **scalable, decoupled, and future-proof integrations**.

Understanding AEM Queues and Topic to Queue Mapping

In Advanced Event Mesh (AEM), **topics** are the logical channels where events are published. They organize and classify business events (e.g., order/created/us/priority).

On the consuming side, applications don't connect directly to topics — they connect through **queues**. A queue acts as a durable mailbox that holds events until the consumer is ready to process them. This guarantees reliability (no event loss if the app is offline) and enables load balancing when multiple consumers share the same queue.

The **Topic-to-Queue Mapping** is what links the two worlds:

- You define which topics a queue should subscribe to, often using **wildcards** for flexibility (e.g., a queue might map to order/created/> to capture all order creation events).
- Once mapped, any event published to that topic is automatically delivered into the queue, ready for consumption.

This separation of **publishers (topics)** and **consumers (queues)** is what gives AEM its scalability and decoupling power. Publishers never need to know who the consumers are, and consumers can independently decide which events matter to them.

Hands-On Warm-Up: Playing with AEM Topics & Queues

Before we dive into the Salesforce scenario, let’s do a short warm-up to see AEM Topics and Queues in action.

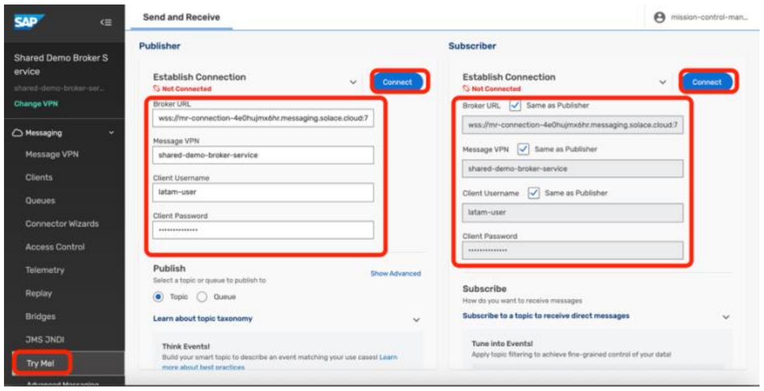
Click here to open broker manager Try-Me UI, which you will use to try publishing events on dynamic topics and create subscriber and subscribe to wild card subscriptions to demonstrate filtering at source.

You will need the following credentials to publish & subscribe events.

Host Name: wss://mr-connection-4e0hujmx6hr.messaging.solace.cloud:7443
Username: latam-user
Password: latam-password

Step 1 – Connect Publisher and Subscriber

Use the broker credentials above and paste it in the Publisher section of the screen. Click “Connect”. The publishing client is now connected, and the credentials are automatically copied over on the Subscriber side as well. Click “Connect” on the subscriber side as well. You subscriber application is now connected.



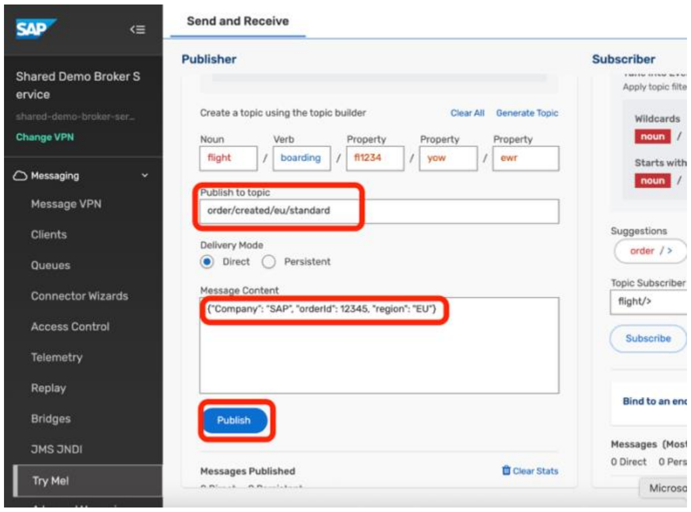
Publisher — Establish Connection — Not Connected
Broker URL: wss://mr-connection-4e0hujmx6hr.messaging.solace.cloud:7
Message VPN: shared-demo-broker-service
Client Username: latam-user
Client Password: ••••••••••••••
Publish — Select a topic or queue to publish to: (●) Topic ( ) Queue
Think Events! Build your smart topic to describe an event matching your use cases! Learn more about best practices

Subscriber — Establish Connection — Not Connected
Broker URL [✓] Same as Publisher: wss://mr-connection-4e0hujmx6hr.messaging.solace.cloud:7
Message VPN [✓] Same as Publisher: shared-demo-broker-service
Client Username [✓] Same as Publisher: latam-user
Client Password: ••••••••••••••
Subscribe — How do you want to receive messages
Tune into Events! Apply topic filtering to achieve fine-grained control of your data!

Step 2 – Publish test messages

1. Open the **Try Me Utility** in your AEM Console.
2. Use the publisher side to publish an event with the
   Topic:
   order/created/eu/standard

   Message Content/Payload example (Please replace the company name with your Company name below):
   {“Company”: “SAP”, "orderId": 12345, "region": "EU"}



Create a topic using the topic builder
Noun: flight / Verb: boarding / Property: fl1234 / Property: yow / Property: ewr
Publish to topic: order/created/eu/standard
Delivery Mode: (●) Direct ( ) Persistent
Message Content: {"Company": "SAP", "orderId": 12345, "region": "EU"}
Messages Published
Apply topic filte
Topic Subscriber: flight/>
Messages (Most
0 Direct 0 Pers

3. Click “Publish”. You would notice that message has been published but not received yet. That’s because, we haven’t subscribed to a required topic on the subscriber side.

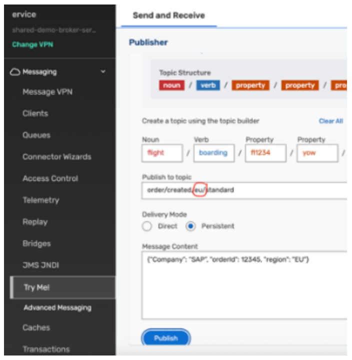
Step 3 – Subscribe to Topic(s)

- Use the subscriber side to subscribe to an event with the Topic:
  order/created/>

- Click Subscribe & then click Publish.
- You would notice that every message you are publishing, is appearing on the subscriber side.
- Now Change the Publish topic to "order/updated/eu/standard". Now when you publish, it wont appear on the subscriber side, why? It's because the subscriber is only interested in order created events and **not** order updated events.

Step 4 – What if the subscriber is offline?

- **Scenario**: An **Order Creation Service** publishes events whenever new orders are created. However, sometimes the **Subscriber Application** may be offline (for maintenance, network issues, or downtime). In such cases, the subscriber still needs to receive all missed events once it comes back online.
- **Objective**: Demonstrate how **Queues** in AEM provide *guaranteed delivery* by storing events until the subscriber is available again.
- **Create following Queues for the Subscriber(s)**
  - **Queue 1 (AllOrdersQ):** Subscribe to order/created/>
    This queue will capture *all* order creation events.
  - **Queue 2 (EuropeOrdersQ):** Subscribe to order/created/eu/>
    This queue will only capture order creation events for the EU region.
- Click "**Disconnect**" on the Subscriber side.
- On the publisher side, change the Publish topic to "order/created/eu/standard". Click "**Publish**" once.



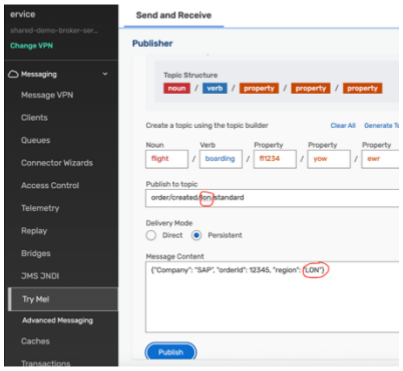
Noun: flight / Verb: boarding / Property: fl1234 / Property: yow

Publish to topic: order/created/eu/standard

Delivery Mode: Direct [✓ Persistent]

Message Content: {"Company": "SAP", "orderId": 12345, "region": "EU"}

- On the publisher side, change the Publish topic to "order/created/lon/standard". Click "**Publish**" once.



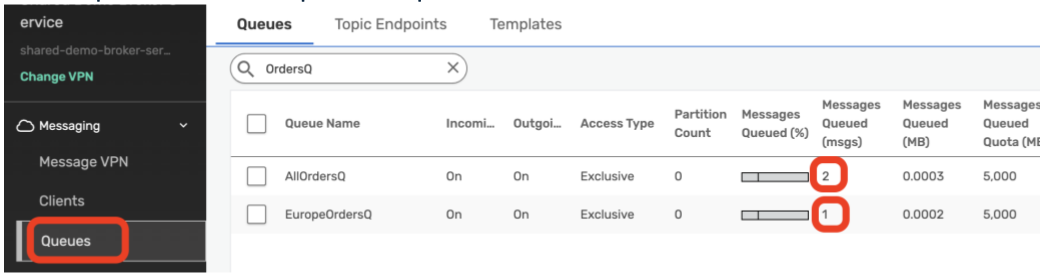
Noun: flight / Verb: boarding / Property: fl1234 / Property: yow / Property: ewr

Publish to topic: order/created/lon/standard

Delivery Mode: Direct [✓ Persistent]

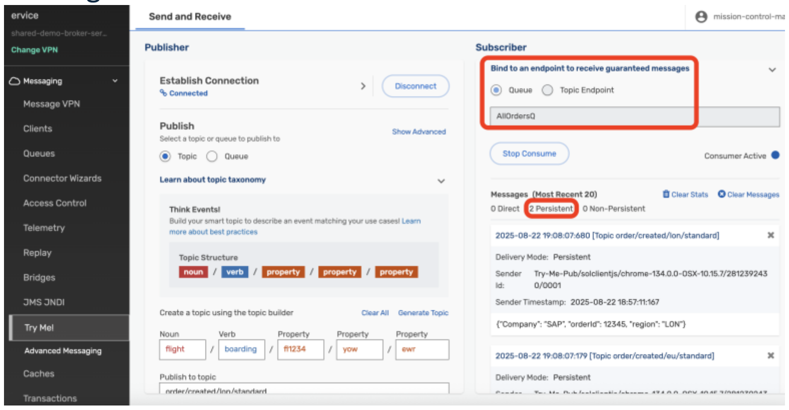
Message Content: {"Company": "SAP", "orderId": 12345, "region": "LON"}

- You would notice that AllOrdersQ has both the events that you published above, while EuropeOrdersQ only has 1 event that you published above. Exactly what the subscription on these queue requests.



	Queue Name	Incomi...	Outgoi...	Access Type	Partition Count	Messages Queued (%)	Messages Queued (msgs)	Messages Queued (MB)	Messages Queued Quota (M
	AllOrdersQ	On	On	Exclusive	0		2	0.0003	5,000
	EuropeOrdersQ	On	On	Exclusive	0		1	0.0002	5,000

- The subscriber, instead of connecting to a topic, can bind to these queues and drain messages.



Publish
Select a topic or queue to publish to
[✓ Topic] Queue

Think Events!
Build your smart topic to describe an event matching your use cases! Learn more about best practices

Topic Structure
noun / verb / property / property / property

Create a topic using the topic builder

Noun		Verb		Property		Property		Property
flight	/	boarding	/	fl1234	/	yow	/	ewr

Publish to topic
order/created/lon/standard

Bind to an endpoint to receive guaranteed messages
[✓ Queue] Topic Endpoint
AllOrdersQ

Consumer Active ●

Messages (Most Recent 20)
0 Direct 2 Persistent 0 Non-Persistent

2025-08-22 19:08:07:680 [Topic order/created/lon/standard]
Delivery Mode: Persistent
Sender Id: Try-Me-Pub/solclientjs/chrome-134.0.0-OSX-10.15.7/281239243 0/0001
Sender Timestamp: 2025-08-22 18:57:11:167
{"Company": "SAP", "orderId": 12345, "region": "LON"}

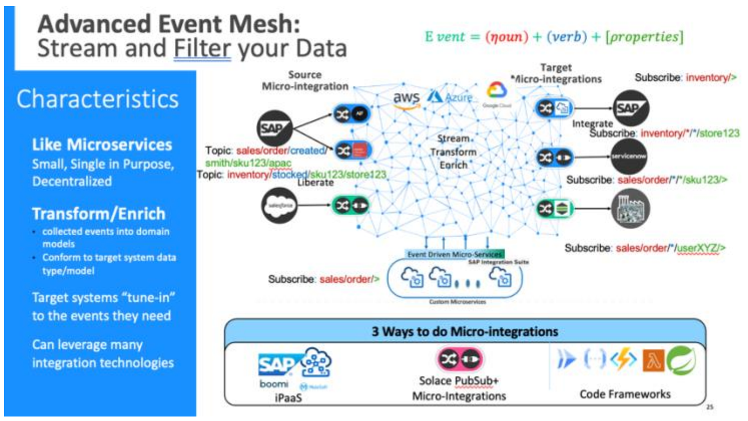
2025-08-22 19:08:07:179 [Topic order/created/eu/standard]
Delivery Mode: Persistent

Key Takeaway

- **Topics = digital "hashtags" for events**
- **Queues = durable mailboxes for subscribers**
- With just a topic structure and subscriptions, AEM ensures the *right events reach the right consumers — no coding required*.

The modern approach to integration is a 3-step approach

1. **Liberate & Integrate,** Liberate your business data by publishing from source systems as "Events" on Dynamic Topics, and integrate it with Destinations.
2. **Stream & Filter** these events for target systems, anywhere they exists. Transform & Enrich these "Events" using micro-integrations (Aka. iFlow)
3. **Democratize** these "Events" by providing self-service discovery, access and governance. **<u>This is out of scope</u>** for this workshop but the capability is available in AEM.



Advanced Event Mesh:
Stream and Filter your Data

Characteristics

Like Microservices
Small, Single in Purpose, Decentralized

Transform/Enrich
- collected events into domain models
- Conform to target system data type/model

Target systems "tune-in" to the events they need

Can leverage many integration technologies

E vent = (noun) + (verb) + [properties]
Source Micro-integration
Target Micro-integrations
Subscribe: inventory/>
Integrate
Subscribe: inventory/*/*/store123
Topic: sales/order/created/smith/sku123/apac
Topic: inventory/stocked/sku123/store123
Liberate
Stream Transform Enrich
Subscribe: sales/order/*/*/sku123/>
Subscribe: sales/order/*/userXYZ/>
Event Driven Micro-Services
SAP Integration Suite
Subscribe: sales/order/>
Custom Microservices
3 Ways to do Micro-integrations
iPaaS
Solace PubSub+ Micro-Integrations
Code Frameworks

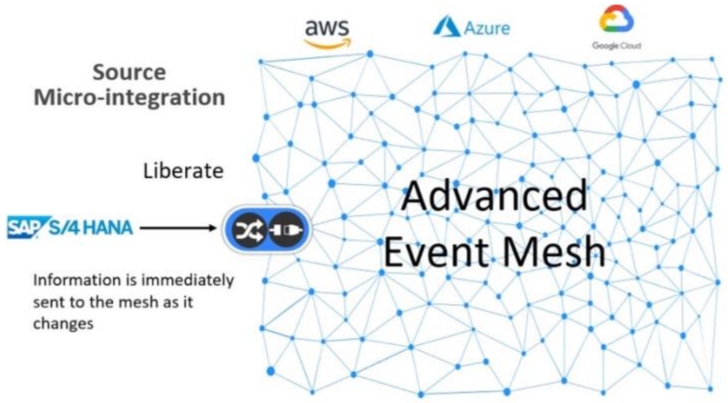
Today , we will be guiding you through the Micro Integration Approach using event-driven integration architecture that will help you build highly decoupled, scalable integrations across hybrid cloud environment.

Scenario 1: Salesforce Integration (Hands-On)

Modern Approach – Event-Driven Integration

Step 1: Liberate your Data

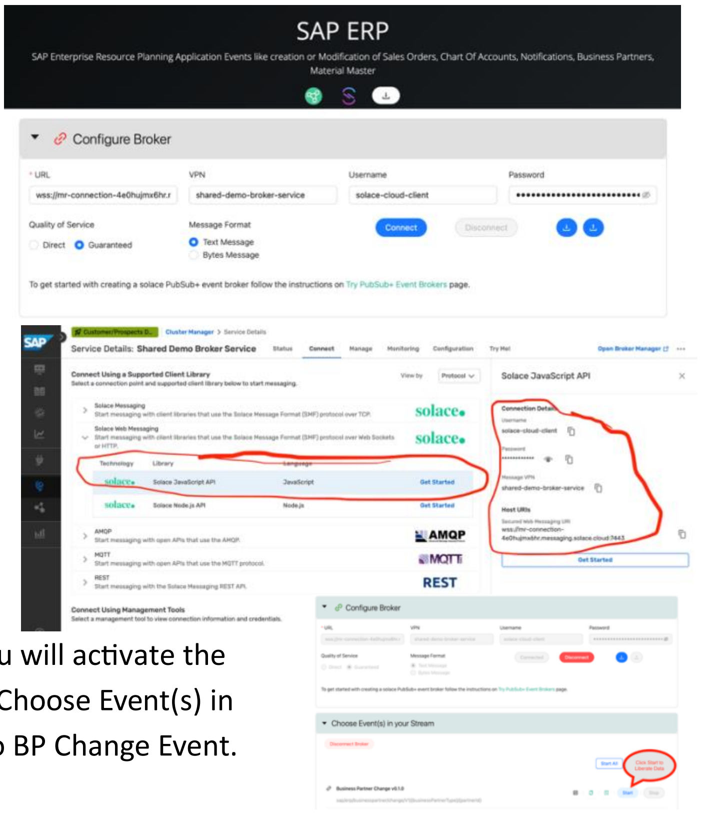
In the environment today, we will be simulating a typical customer setup where we want to have Events that originate from an SAP System like S/4 or ECC published to the Advanced Event Mesh. This allows customers to then react in real time to these events with services like Cloud Integration/iFlow in real time.



The pattern that we suggest to customers is the Micro Integration Pattern where interfaces are established without any knowledge of the applications that will consume them. In the diagram below, you can see that the destination for our events is the Advanced Event Mesh(aka AEM). Events like Business Partner Create and Change will be published to the AEM without any knowledge of which or how many applications will be using that event. This makes the integration much easier to implement as you do not require knowledge of the consuming applications.

In our setup for the workshop, we will be using an Event Generator that will mimic "Events" coming out of your ERP system. **<u>We start by heading to the site</u>** here.

You will need Web Messaging credentials of your AEM broker service here. To obtain those credentials, please logon to you AEM Console, click your broker service under "Cluster Manager" tab and go to "Connect Tab and look for "Solace Web Messaging". Click on it and click "Solace JavaScript APIs" and you will see broker credentials for this protocol on right side.


SAP ERP

SAP Enterprise Resource Planning Application Events like creation or Modification of Sales Orders, Chart Of Accounts, Notifications, Business Partners, Material Master

Configure Broker

URL	VPN	Username	Password
wss://mr-connection-4e0hujmx6hr.r	shared-demo-broker-service	solace-cloud-client	••••••••••••••••••••••••

Quality of Service: Direct [✓ Guaranteed]

Message Format: [✓ Text Message] Bytes Message

To get started with creating a solace PubSub+ event broker follow the instructions on Try PubSub+ Event Brokers page.

Cluster Manager > Service Details

Service Details: Shared Demo Broker Service

Connect Using a Supported Client Library
Select a connection point and supported client library below to start messaging.

Solace Messaging
Start messaging with client libraries that use the Solace Message Format (SMF) protocol over TCP.

Solace Web Messaging
Start messaging with client libraries that use the Solace Message Format (SMF) protocol over Web Sockets or HTTP.

Technology	Library	Language	
solace	Solace JavaScript API	JavaScript	Get Started
solace	Solace Node.js API	Node.js	Get Started

AMQP
Start messaging with open APIs that use the AMQP.

MQTT
Start messaging with open APIs that use the MQTT protocol.

REST
Start messaging with the Solace Messaging REST API.

Connect Using Management Tools
Select a management tool to view connection information and credentials.

Solace JavaScript API

Connection Details
Username: solace-cloud-client
Password: ••••••••••••
Message VPN: shared-demo-broker-service

Host URIs
Secured Web Messaging URI: wss://mr-connection-4e0hujmx6hr.messaging.solace.cloud:7443

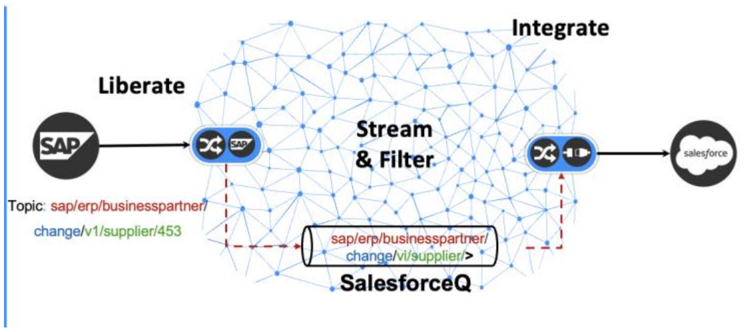
Choose Event(s) in your Stream

Business Partner Change v0.1.0

After you have copied these credentials in the "Configure Broker" screen, hit "Connect" and now for the workshop, you will activate the Business Partner Change events. Under "Choose Event(s) in your Stream" section, click "Start" next to BP Change Event.

Step 2: Stream, Filter and Queue

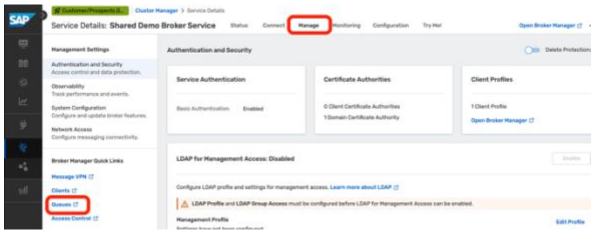
Now that your data from source system is liberated and AEM broker service is streaming these events, for our exercise today, we will need a queue that will subscribe to this event using wild card subscription for receiving filtered data for all business partner related change events as shown in this diagram.


Liberate
Integrate
Stream
& Filter
Topic: sap/erp/businesspartner/change/v1/supplier/453
sap/erp/businesspartner/change/v1/supplier/>
SalesforceQ
salesforce

This queue will be used by our first consumer, which is responsible for delivering events to Salesforce. The queue ensures **guaranteed delivery** of messages even when the target system is temporarily unavailable. For example, if Salesforce is shut down for maintenance or an application that processes new business partner records is offline, events will not be lost. Instead, they will be held safely in the queue and delivered once the system is back online. This is a key benefit of queues in an event-driven architecture.

Follow the steps below to create this queue on your AEM broker service.

Go to [Advanced Event Mesh Console]() **-> Cluster Manager** -> **{your service}** -> **Manage** -> **Queues** - to open the Broker UI

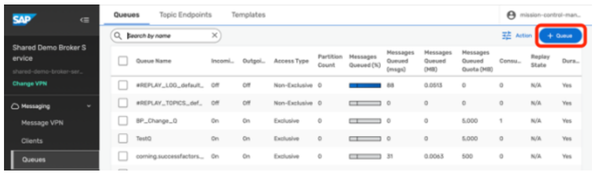


LDAP for Management Access: Disabled

Configure LDAP profile and settings for management access. Learn more about LDAP

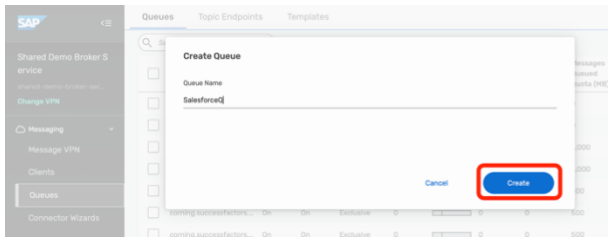
LDAP Profile and LDAP Group Access must be configured before LDAP for Management Access can be enabled.

Click on the "**+ Queue**" button to bring up the create queue dialog.



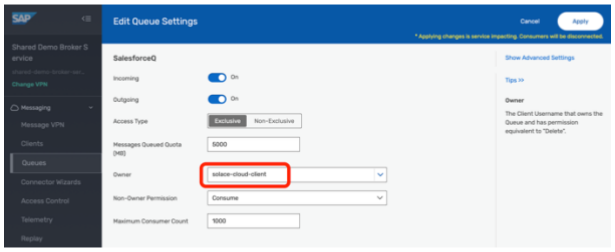
	Queue Name	Incomi...	Outgoi...	Access Type	Partition Count	Messages Queued (%)	Messages Queued (msgs)	Messages Queued (MB)	Messages Queued Quota (MB)	Consu...	Replay State	Dura...
	#REPLAY_LOG_default...	Off	Off	Non-Exclusive	0		88	0.0513	0	0	N/A	Yes
	#REPLAY_TOPICS_def...	Off	Off	Non-Exclusive	0		0	0	0	0	N/A	Yes
	BP_Change_Q	On	On	Exclusive	0		0	0	5,000	1	N/A	Yes
	TestQ	On	On	Exclusive	0		0	0	5,000	0	N/A	Yes
	coming.successfactors...	On	On	Exclusive	0		31	0.0063	500	0	N/A	Yes

Provide the Queue Name



Create the following queue and provide the details as given (copy & paste where appropriate).

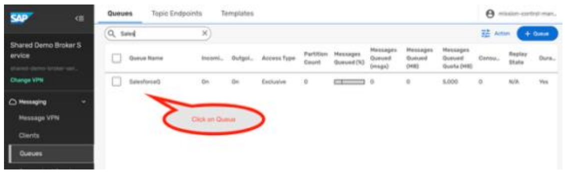
- **Queue Name**: SalesforceQ
- **Queue Owner**: solace-cloud-client


Edit Queue Settings
Cancel
Apply
* Applying changes is service impacting. Consumers will be disconnected.
Show Advanced Settings
Tips >>
Owner
The Client Username that owns the Queue and has permission equivalent to "Delete".

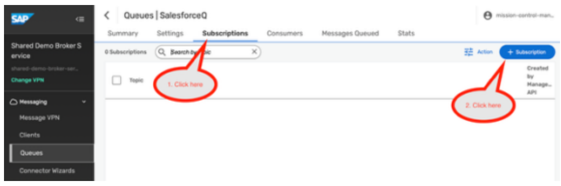
SalesforceQ

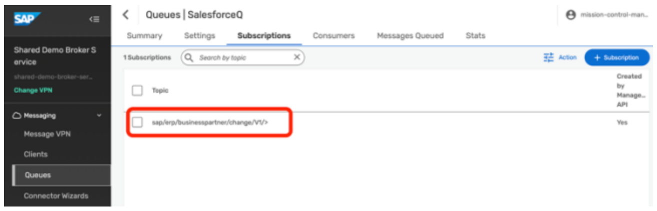
Incoming	On
Outgoing	On
Access Type	[✓ Exclusive] Non-Exclusive
Messages Queued Quota (MB)	5000
Owner	solace-cloud-client
Non-Owner Permission	Consume
Maximum Consumer Count	1000

Once the queue is created, click on the queue name in the list, navigate to the Subscriptions tab and open the subscriptions dialog and add the following subscription:
sap/erp/businesspartner/change/V1/>



	Queue Name	Incomi...	Outgoi...	Access Type	Partition Count	Messages Queued (%)	Messages Queued (msgs)	Messages Queued (MB)	Messages Queued Quota (MB)	Consu...	Replay State	Dura...
	SalesforceQ	On	On	Exclusive	0		0	0	5,000	0	N/A	Yes

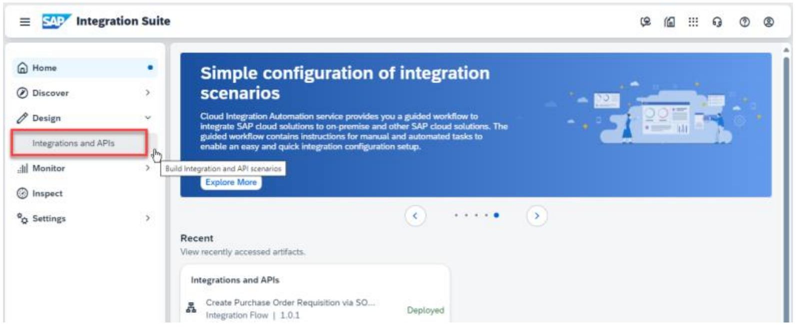




	Topic	Created by Manage... API
	sap/erp/businesspartner/change/V1/>	Yes

Step 3: Integrate

We will use Integration Suite to integrate using micro-integration concepts we discussed earlier. To do this, you will access your SAP Integration Suite service and click on Design section in the left-hand menu. Now click on "Integration and APIs" under Design -



Simple configuration of integration scenarios

Cloud Integration Automation service provides you a guided workflow to integrate SAP cloud solutions to on-premise and other SAP cloud solutions. The guided workflow contains instructions for manual and automated tasks to enable an easy and quick integration configuration setup.

Recent

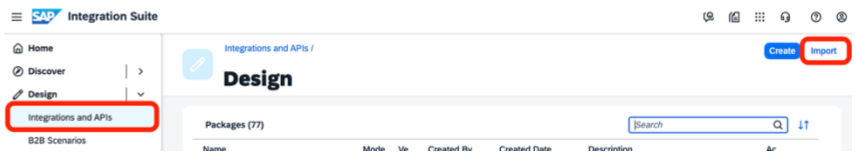
View recently accessed artifacts.

Integrations and APIs

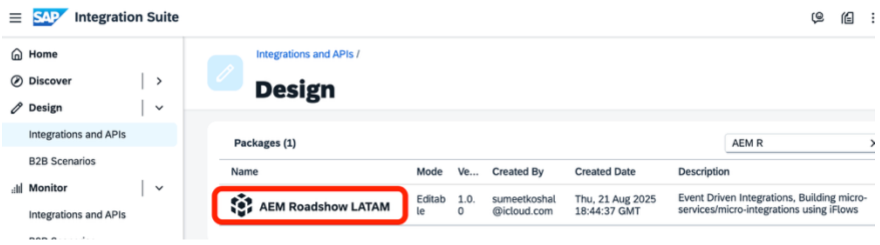
Create Purchase Order Requisition via SO...
Integration Flow | 1.0.1

Import the integration package

We have created integration package for this workshop to save some time. Please use the following link to download the package. Click on Import to import the integration package you just downloaded.



After you click import, please select integration package "AEM Roadshow LATAM.zip" and click "Open". This will import the package successfully.

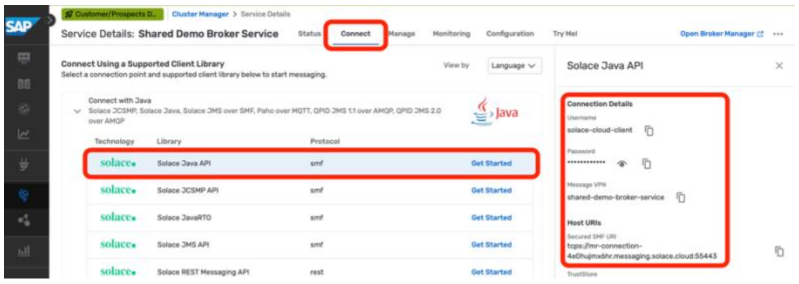


Packages (1)

Name	Mode	Ve...	Created By	Created Date	Description
AEM Roadshow LATAM	Editable	1.0.0	sumeetkoshal@icloud.com	Thu, 21 Aug 2025 18:44:37 GMT	Event Driven Integrations, Building micro-services/micro-integrations using iFlows

Obtain your AEM Broker Service Credentials

Before heading back to Integration Suite, let's head to our [Advanced Event Mesh Console](#) and go to **Cluster Manager** -> **{your service}**. Select the connection point and protocol that you want to use to connect your Integration Suite flows by going to the "Connect" tab, click on "Connect with Java" and then click on "Solace Java API". Make a note of the connectivity details underneath "Solace Java API" (click on the section to open it up). We will need these details in the next steps when configuring your iFlows.



Service Details: Shared Demo Broker Service

Connect Using a Supported Client Library
Select a connection point and supported client library below to start messaging.

Connect with Java
Solace JCSMP, Solace Java, Solace JMS over SMF, Paho over MQTT, QPID JMS 1.1 over AMQP, QPID JMS 2.0 over AMQP

Technology	Library	Protocol	
solace	Solace Java API	smf	Get Started
solace	Solace JCSMP API	smf	Get Started
solace	Solace JavaRTO	smf	Get Started
solace	Solace JMS API	smf	Get Started
solace	Solace REST Messaging API	rest	Get Started

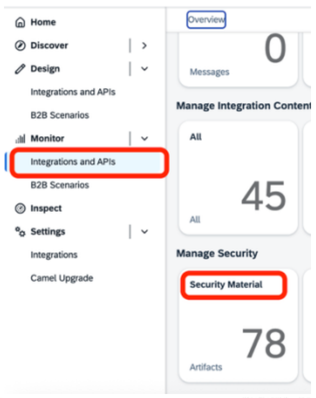
Solace Java API

Connection Details
Username: solace-cloud-client
Password: ••••••••••••
Message VPN: shared-demo-broker-service
Host URIs
Secured SMF URI: tcps://mr-connection-4e0hujmx6hr.messaging.solace.cloud:55443
TrustStore

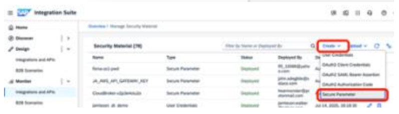
Update your iFlows with your AEM Broker Service Credentials

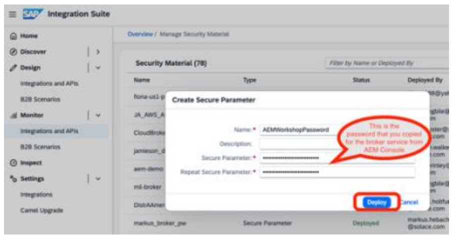
The integration package has 2 iFlows that need to be updated to connect to your AEM broker service. Please do the following steps in order:

- Create SecurityMaterial - Rather than entering the AEM instance password directly on the iFlow and making it visible to everyone with access to the iFlow, we will create a **SecureParameter** which will store the password securely and we then just reference this in our iFlows.
  Go to your Integration Suite **-> Monitor** -> **Manage Security** -> **Security Material**.

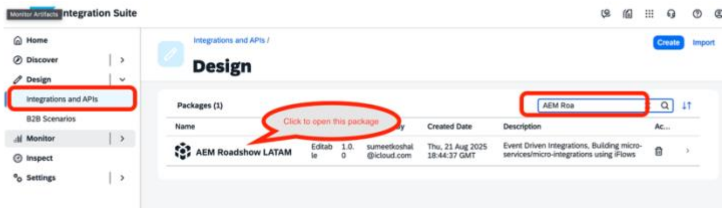


  In here, create security credentials for your AEM broker service.
  Create **SecureParameter** `AEMWorkshopPassword` and store the password for your `solace-cloud-client` application user credentials. Then click "Deploy".

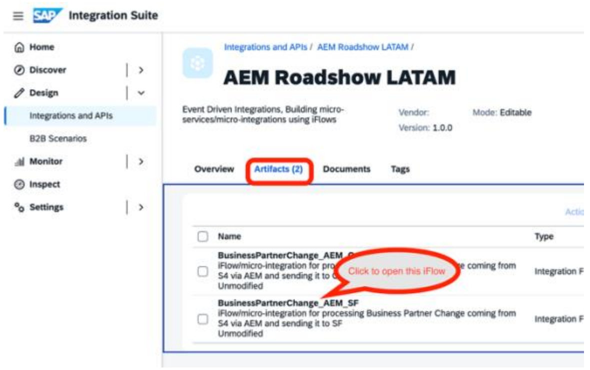



Integration Suite
Overview / Manage Security Material
Security Material (78)
Create Secure Parameter
Name: AEMWorkshopPassword
Description:
Secure Parameter:
Repeat Secure Parameter:
This is the password that you copied for the broker service from AEM Console
Deploy
Cancel

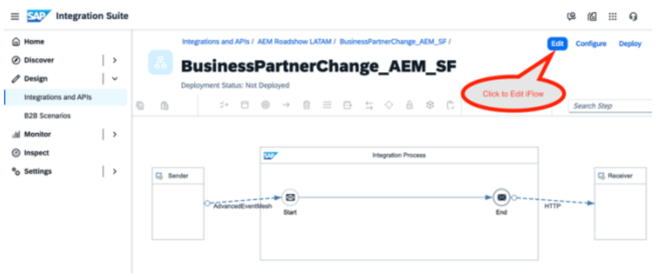
- Update Salesforce iFlow
  In order to update the Salesforce iFlow, in your Integration Suite UI, please click on "**Integrations and APIs**" under **Design**. Look for "**AEM Roadshow LATAM**" package that you have imported previously. Click to open this package.



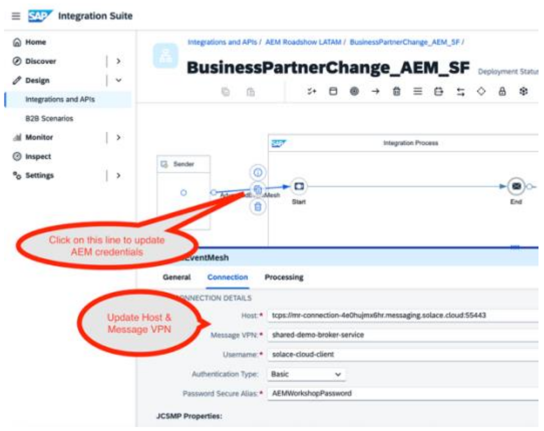
  Once the package is opened, click on "Artifacts", click to open "**BusinessPartnerChange_AEM_SF**" iFlow.



The iFlow uses the ***SAP AEM Adapter*** to read events from the queue on your AEM instance. Within the iFlow, an outgoing call will be made to Salesforce. We will show you the message getting to the Salesforce system and the successful response.

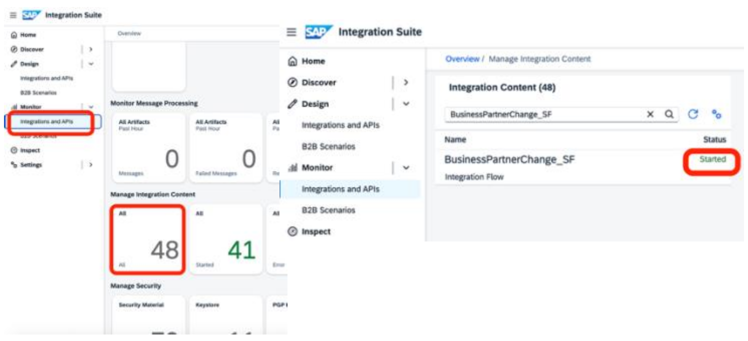


The iFlow has been preconfigured with our AEM credentials and mock Salesforce REST endpoint. While we are planning to use the mock Salesforce service, you must update configuration of AEM credentials to point to your AEM broker service. Click on "Edit", and click on the line from Sender adapter, where it says "AdvancedEventMesh". Below the iFlow,you will see AEM configuration tabs. Click on "Connection" tab and replace the host and message VPN name with the credentials of you broker that you saved in previous step. Under "Processing" you will see that the iFlow is connecting to SalesforceQ, that we previously created. **Click Save, then Deploy.**


Integration Suite
Home
Discover
Design
Integrations and APIs
B2B Scenarios
Monitor
Inspect
Settings
Integrations and APIs / AEM Roadshow LATAM / BusinessPartnerChange_AEM_SF /
BusinessPartnerChange_AEM_SF
Deployment Status
Sender
Integration Process
Start
End
Click on this line to update AEM credentials
Update Host & Message VPN
General
Connection
Processing

CONNECTION DETAILS
Host: * tcps://mr-connection-4e0hujmx6hr.messaging.solace.cloud:55443
Message VPN: * shared-demo-broker-service
Username: * solace-cloud-client
Authentication Type: Basic
Password Secure Alias: * AEMWorkshopPassword
JCSMP Properties:

To verify if the iFlow is deployed successfully, click on "**Integrations and APIs**" and click on "**ALL**" under "**Manage Integration Content**". The staus of iFlow "BusinessPartnerChange_SF" should show "**Started**".



***Note**: Please let us know what your status is. If its **Started**, we should be able to see the logs of the mock service to see end to end flow. If you see Error, still let us know so that we can help you debug and resolve.*

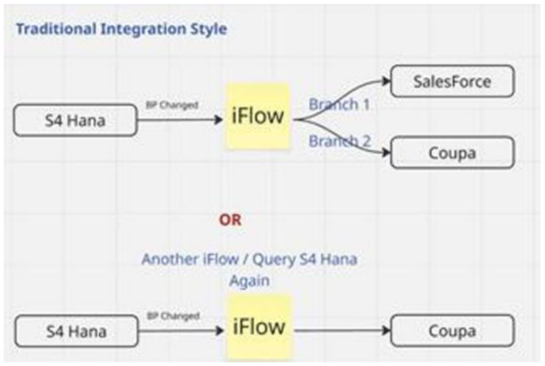
5 Framing The 2nd Business case scenario

Business Requirement:

In addition to updating Salesforce, the business now requires that any changes made to a Business Partner (Supplier) record in SAP S/4HANA are also sent to **Coupa**, to ensure supplier profiles are accurately reflected in procurement workflows.

How Would You Approach This Today?

Most likely, you will take the traditional approach again which is to using a branch or creating another P2P iFlow. The implementation path would look like the following:



1. You may extend your existing iFlow to also call **Coupa's Supplier API**, or
2. Build a **second iFlow** solely for Coupa

Challenges Amplified:

- Increased payload transformation complexity
- Error handling across systems becomes harder to manage
- Deployments take longer with every new consumer
- Adding a third or fourth downstream system (e.g., Data Lake, SAP Analytics Cloud, etc.) would require further duplication or chaining

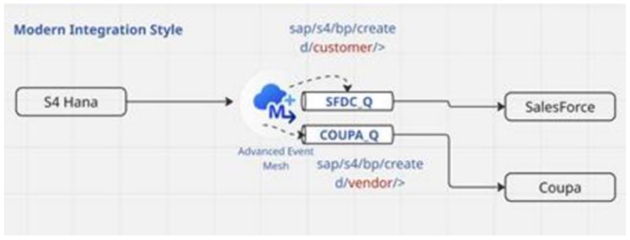
Note: Today however, we are not going to use the P2P approach, instead, we will use event-driven integration approach that will help you add a new consumer without impacting existing producer/consumer application.

6 The Modern Approach – Event-Driven Architecture (EDA) – Seamless Expansion

Simply "Integrate"

Since we already "Liberated" the data from S4 Hana previously, and it is already available via AEM, we will simply create a new iFlow (pre-created for this workshop) and let it connect to its own queue called "**CoupaQ**" (please follow the steps to create this queue as described for creating SalesforceQ in previous section of this handout).

1. No changes are needed in the source system (S/4HANA continues publishing Business Partner events).
2. Coupa integration becomes just another subscriber.
3. You simply build a second event-consumer iFlow in Integration Suite for Coupa.



Benefits Realized:

1. Business logic for each consumer is encapsulated independently
2. No change to existing applications (producer/consumer(s)).
3. No regression risk to Salesforce flow
4. Future systems (e.g., Compliance Monitoring, Risk Engines) can subscribe instantly without any changes to the event producer

Scenario 2 – Coupa Integration Hands-On

Step 1: Create required queues and subscriptions for Coupa iFlow

Instead of repeating the same instructions for creating queue, please create the queue with the Queue Details mentioned below queue by following the instructions "Scenario 1: Salesforce Integration (Hands-On) => Step 2",
Queue details:
• Name: CoupaQ
• Owner: solace-cloud-client
Once the queue is created, click on the queue name in the list, navigate to the Subscriptions tab and open the subscriptions dialog.
sap/erp/businesspartner/change/V1/>

Step 2: Configure and Deploy BusinessPartnerChange_AEM_Coupa iFlow

Instead of repeating the same instructions for configuring and deploying an existing iFlow, please follow the steps from "Scenario 1: Salesforce Integration (Hands-On) => Step 3=>Update Salesforce iFlow". This time, remember to select the iFlow called **"BusinessPartnerChange_AEM_Coupa".**

After deploying this iFlow, you will notice that your **Coupa iFlow** also processes the message. That's because both Salesforce and Coupa are subscribing to the **same topic** — which means the same published event is delivered to multiple consumers automatically. This illustrates the power of decoupling: publish once and let AEM distribute to all interested systems.

7 Demos – Integrations beyond iFlows: If Time permits

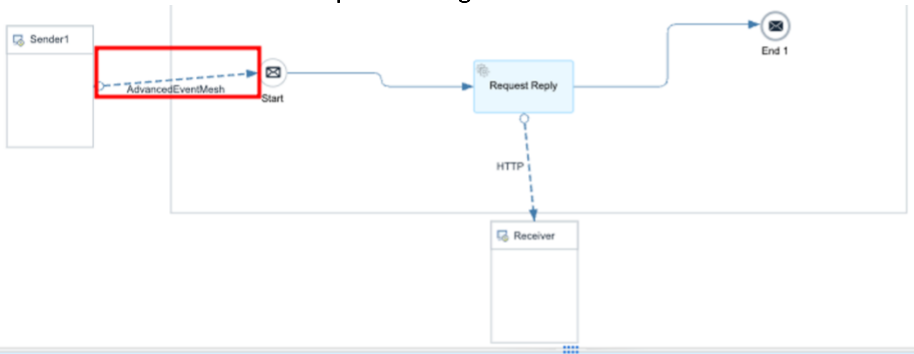
8 Optional: Understanding Retry, Error handling, unblocking message processing – If time permits

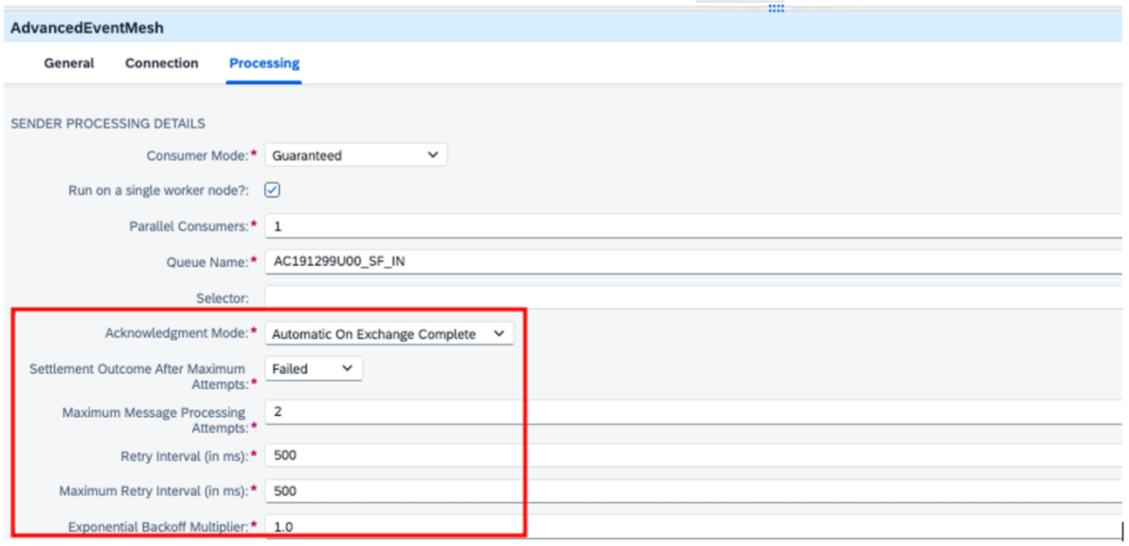
This exercise will demonstrate the benefits you get by adopting a publish/subscribe methodology even if you only have 1 downstream subscriber.

In this scenario, the downstream application becomes unavailable (we will arrange this). You will make some configuration changes to your queues to implement error/retry-handling and ensure that undeliverable messages do not block processing. All this without writing any code!

Adapter settings

Let's take a look at some of the relevant AEM adapter settings that control this behaviour.




AdvancedEventMesh

General | Connection | Processing

SENDER PROCESSING DETAILS

Consumer Mode:*	Guaranteed
Run on a single worker node?:	✓
Parallel Consumers:*	1
Queue Name:*	AC191299U00_SF_IN
Selector:	
Acknowledgment Mode:*	Automatic On Exchange Complete
Settlement Outcome After Maximum Attempts:*	Failed
Maximum Message Processing Attempts:*	2
Retry Interval (in ms):*	500
Maximum Retry Interval (in ms):*	500
Exponential Backoff Multiplier:*	1.0

Let's look at these settings one by one:

1. **Acknowledgement Mode: "Automatic on Exchange Complete"**

The most important setting when it comes to not accidentally acknowledging and therefore removing a message from the broker's queue. This setting tells the flow/AEM adapter to only acknowledge (ack) the message after the flow has successfully completed processing the message. If any error in the processing occurs, the AEM adapter will instead send a negative acknowledgment back (nack) to tell the broker to keep the message and retry it, because it couldn't be successfully processed by the flow. The alternative is to immediately ack the message when it's received, which will always result in the message being removed from the queue even if the flow fails to successfully process the message. (!!)

2. **Settlement Outcome After Maximum Attempts: "Failed"**

This setting controls the nack type and behaviour, we have two options here:

a) **Failed**, which will nack the message back to the broker and lets the broker check the retry count of the message to trigger retries based on the queue settings and only sending messages to DMQ when the retry count on the message has exceeded the max retry settings on the queue.
N.B. With the Failed setting, values of the Maximum Redelivery Count on the queue **and** the max. message processing attempts on the adapter are taken into account.

b) **Rejected**, which will nack the message telling the broker to immediately move the message to DMQ when the AEM adapter settings (Maximum Message Processing Attempts) are exceeded irrespective of queue settings.
N.B. With the Rejected setting, only the value for max. message processing attempts on the adapter is taken into account.

3. **Max. Message Processing Attempts: 2**

Controls how often we want to retry a message inside the iFlow before we "give up" and pass it back to the broker.

4. **Retry interval, Max Retry Interval and Exponential Backoff Multiplier**

These are all settings that control how quickly we want to retry and whether we want to incrementally increase our retry delay with each failure. A good retry delay value prevents the iFlow from repeatedly retrying a message within a few milli-seconds and gives some time for transient error situations to clear before we retry.
Note that the error handling and retry settings go hand-in-hand with the DMQ and retry settings on the input queue for this flow (queue retry settings multiply with the internal retry settings in the iFlow, e.g. if the iFlow tries 2 times internally every time we pass it a message and the broker is configured to retry the same message 3 times to the broker, then we might get 8 executions before the message is actually stopped being processed and moved to the DMQ [(1 initial attempt + 3 times retry) * 2 times retry inside the iFlow = 8 processing attempts]):

Queue configuration for retry handling

8.1.1 Create a Dead Message Queue(DMQ)

Using the instructions from **Error! Reference source not found.**, create a DMQ as follows:

1. AC191299Unn_SF_DMQ queue
• Name: AC191299Unn_SF_DMQ
• Owner: solace-cloud-client

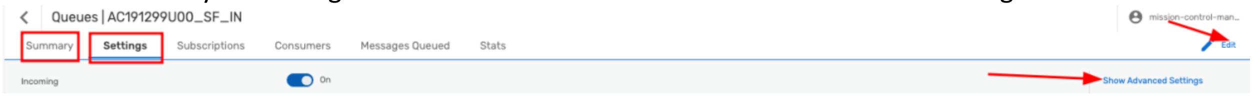
A DMQ is just like any other queue but it is used to hold all messages that have not been successfully processed in a DMQ-enabled iFlow. Not all messages are eligible for dead message processing. There are 4 pre-requisites:

1) A DMQ must exist and be associated with the input queue
2) The DMQ eligibility must be set by the publisher on the incoming message
3) TTL turned on
4) Retry settings configured on the AEM adapter.

8.1.2 Amend the Input queue as follows

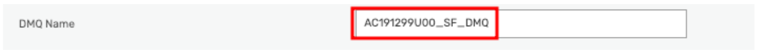
Go to Advanced Event Mesh Console **-> Cluster Manager** -> **{your service}** -> **Manage** -> **Queues** - to open the Broker UI **-> Click on the relevant queue** (e.g. AC191299Unn)

Click on Summary or Settings then click on Edit. Next click on Show Advanced Settings.

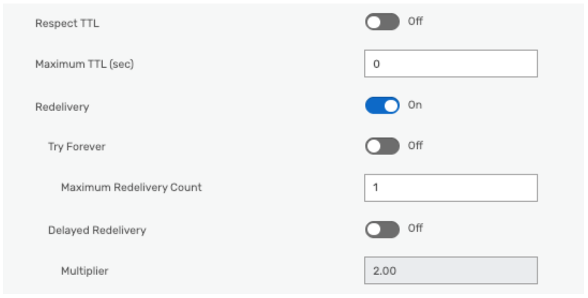


Add the following configurations

- DMQ Name: AC191299Unn_SF_DMQ
- Redelivery: enabled
- Try Forever: disabled
- Maximum Redelivery Count: 2



DMQ Name	AC191299U00_SF_DMQ



Respect TTL	Off
Maximum TTL (sec)	0
Redelivery	On
Try Forever	Off
Maximum Redelivery Count	1
Delayed Redelivery	Off
Multiplier	2.00

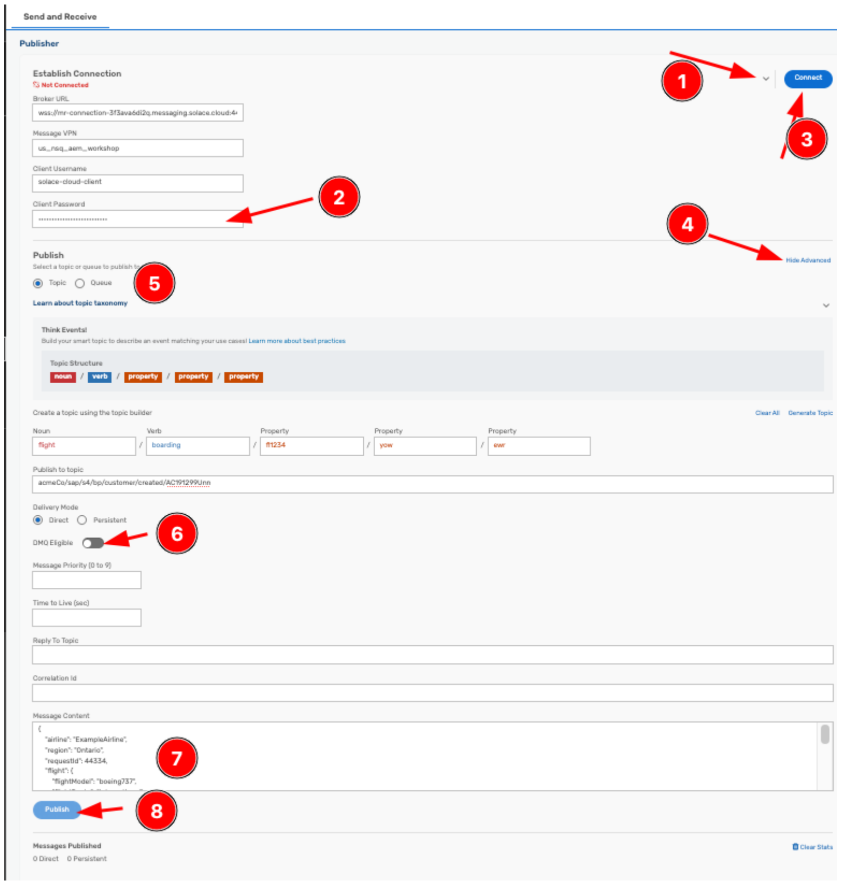
Now that all the required retry configuration is in place, you will send a message and observe the retry behaviour. How many retries do you expect when you publish a message on the input queue for the salesforce iFlow and a problem is encountered within the flow?

8.1.3 Publish a message to the input queue for the salesforce iFlow

As mentioned in 8.1.1 above, the **publisher must set DMQ eligibility** when publishing the message if you want a message to be a candidate for DMQ processing and handling. Carefully follow the instructions below on how to activate this setting in Try Me!

Follow the instructions from **Error! Reference source not found.** to access the Try Me! utility to publish to the iFlow.

See the screenshot below and the following instructions on how to access the DMQ settings in Try Me!



Send and Receive

Publisher

Establish Connection

Not Connected

Broker URL: wss://mr-connection-3f3ava6di2q.messaging.solace.cloud:4

Message VPN: us_nsq_aem_workshop

Client Username: solace-cloud-client

Client Password: ••••••••••••••••••••••••

Publish

Select a topic or queue to publish to

(● Topic) ○ Queue

Learn about topic taxonomy

Think Events!

Build your smart topic to describe an event matching your use cases! Learn more about best practices

Topic Structure

noun / verb / property / property / property

Create a topic using the topic builder

Noun	Verb	Property	Property	Property
flight	boarding	fl1234	yow	ewr

Publish to topic: acmeCo/sap/s4/bp/customer/created/AC191299Unn

Delivery Mode: (● Direct) ○ Persistent

DMQ Eligible

Message Priority (0 to 9)

Time to Live (sec)

Reply To Topic

Correlation Id

Message Content

```
{
  "airline": "ExampleAirline",
  "region": "Ontario",
  "requestId": 44334,
  "flight": {
    "flightModel": "boeing737",
```

Messages Published

0 Direct 0 Persistent

I. Click on the drop-down arrow to reveal the connection credentials dialogue

II. Key in the AEM password in the Client Password field

III. Click on Connect

IV. Click on show Advanced

V. Click on the Topic radio button

VI. Toggle the DMQ eligible button to on

VII. Paste the message content from **Error! Reference source not found.** in the message content window

VIII. Click on Publish

Check the processing logs to get more details on the failure. How many retries do you expect to see based on the settings configured on the queue and AEM adapter?

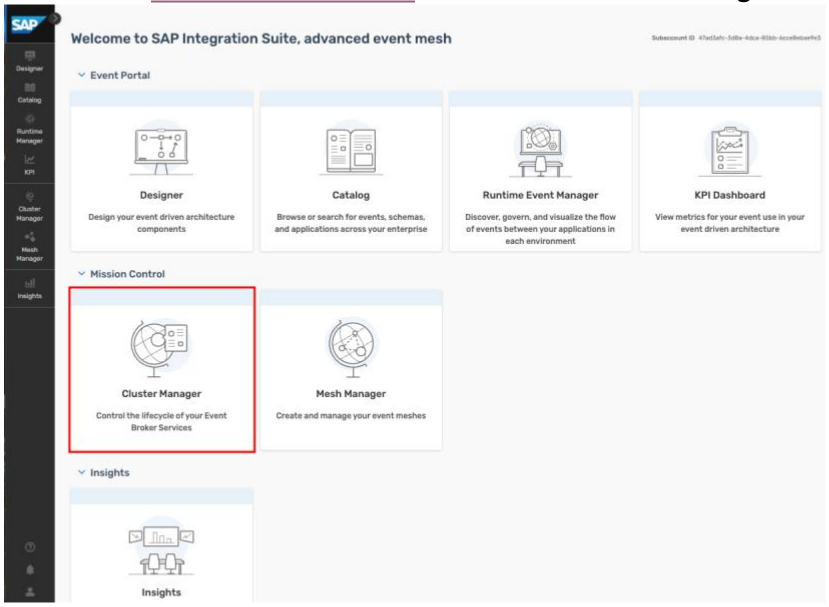
Go to Integration Suite **-> Monitor -> Integrations and APIs -> Monitor Message Processing**

9 APPENDIX – CONNECTING TO THE TRY ME UTILITY

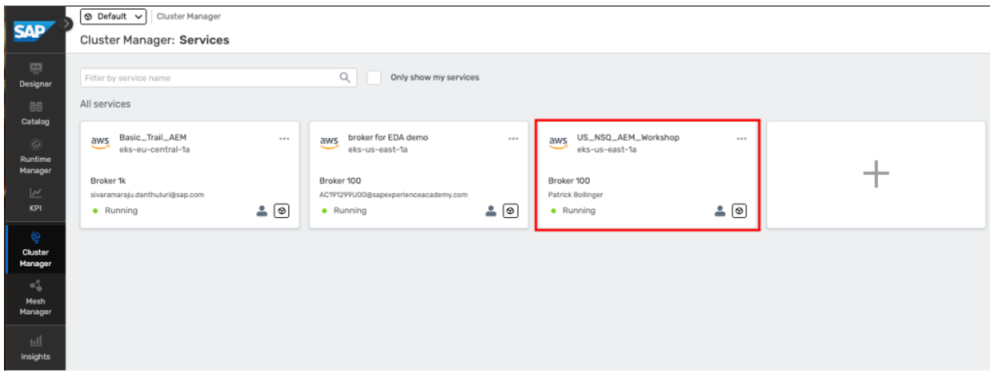
Obtain connection credentials

The Try Me! utility is a JavaScript application for quickly getting up and running with testing smart topics. The application uses Web Sockets and so you'll need to select the appropriate connection credentials.

Use the link Advanced Event Mesh and click on **Cluster Manager**

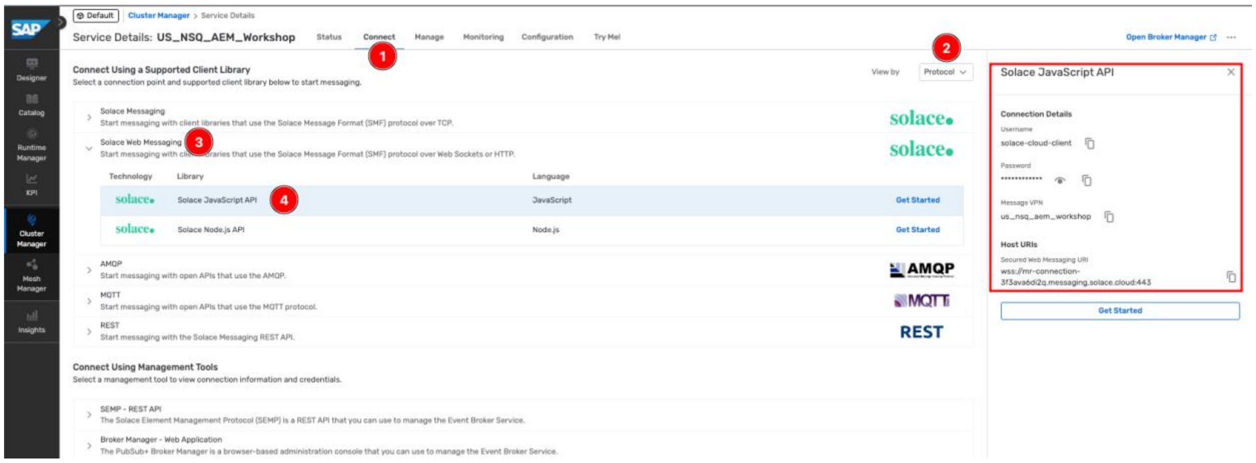

SAP
Designer
Catalog
Runtime Manager
KPI
Cluster Manager
Mesh Manager
Insights
Welcome to SAP Integration Suite, advanced event mesh
Subaccount ID 47ad3afc-3d8a-4dca-85bb-6cce8ebae9e3
Event Portal
Designer
Design your event driven architecture components
Catalog
Browse or search for events, schemas, 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
Mesh Manager
Create and manage your event meshes
Insights
Insights

Next, select the broker that is being used for this workshop: '**US_NSQ_AEM_Workshop**'



After selecting the AEM instance above,
1 - click on the "Connect" tab
2 - order by protocol
3 - click on Solace Web Messaging.

Make a note of the connectivity details underneath "Solace JavaScript API" (click on "Solace JavaScript API" to reveal the connection credentials).



Service Details: US_NSQ_AEM_Workshop

Connect Using a Supported Client Library
Select a connection point and supported client library below to start messaging.

Solace Messaging
Start messaging with client libraries that use the Solace Message Format (SMF) protocol over TCP.

Solace Web Messaging
Start messaging with client libraries that use the Solace Message Format (SMF) protocol over Web Sockets or HTTP.

Technology	Library	Language	
solace	Solace JavaScript API	JavaScript	Get Started
solace	Solace Node.js API	Node.js	Get Started

AMQP
Start messaging with open APIs that use the AMQP.

MQTT
Start messaging with open APIs that use the MQTT protocol.

REST
Start messaging with the Solace Messaging REST API.

Connect Using Management Tools
Select a management tool to view connection information and credentials.

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

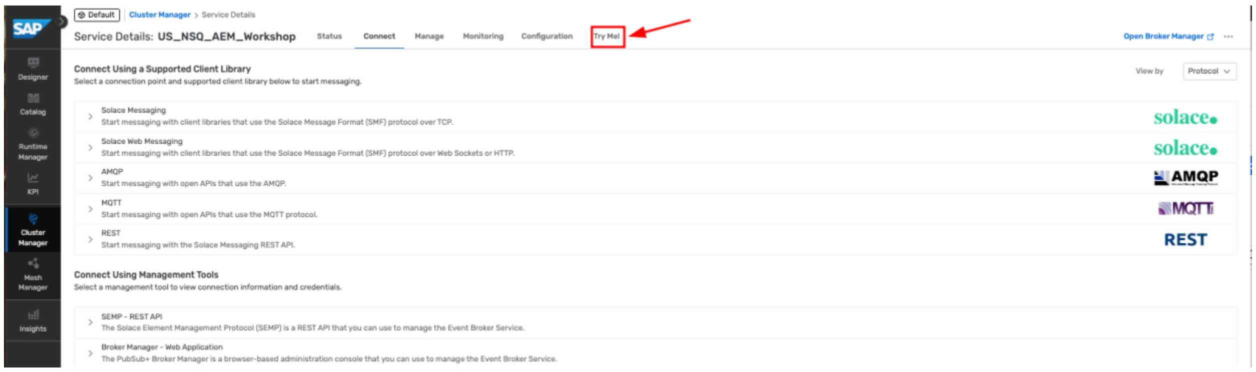
Broker Manager - Web Application
The PubSub+ Broker Manager is a browser-based administration console that you can use to manage the Event Broker Service.

Solace JavaScript API

Connection Details
Username
solace-cloud-client
Password
••••••••••••
Message VPN
us_nsq_aem_workshop

Host URIs
Secured Web Messaging URI
wss://mr-connection-3f3ava6di2q.messaging.solace.cloud:443

You will use these credentials shortly within the Try Me Tab. Select the Try me Tab and proceed with “Open Broker Manager”.



Service Details: US_NSQ_AEM_Workshop

Connect Using a Supported Client Library
Select a connection point and supported client library below to start messaging.

Solace Messaging
Start messaging with client libraries that use the Solace Message Format (SMF) protocol over TCP.

Solace Web Messaging
Start messaging with client libraries that use the Solace Message Format (SMF) protocol over Web Sockets or HTTP.

AMQP
Start messaging with open APIs that use the AMQP.

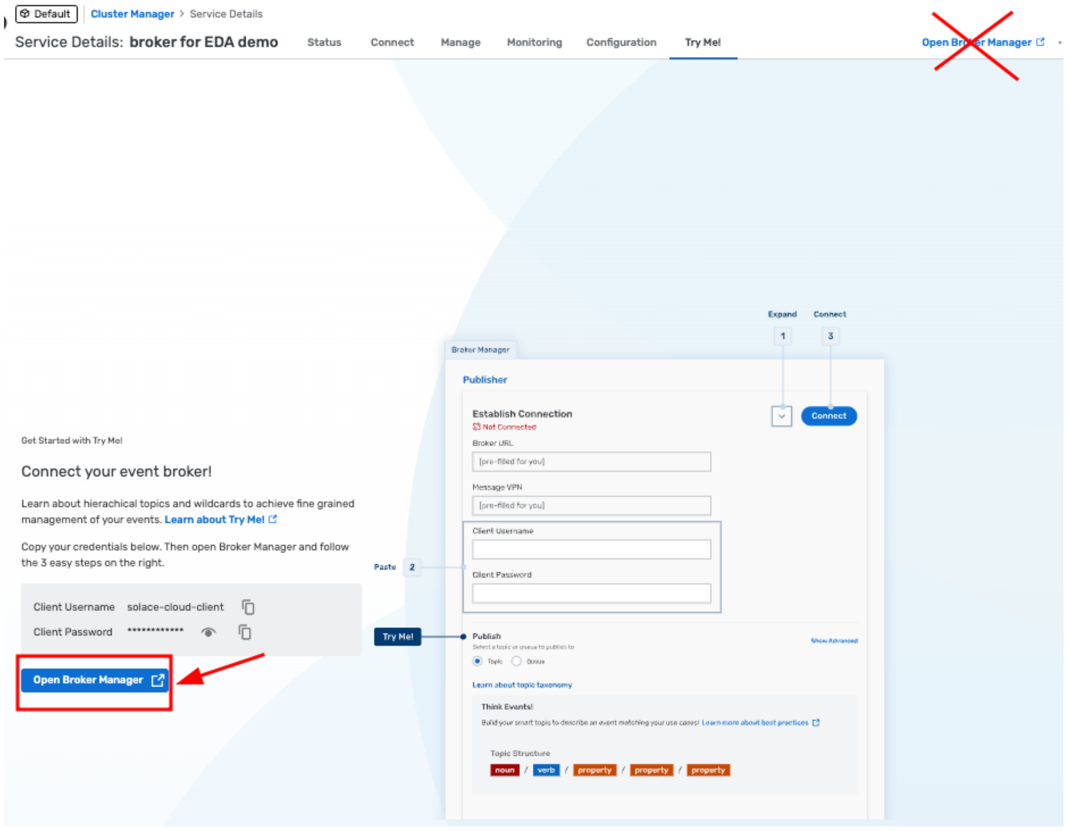
MQTT
Start messaging with open APIs that use the MQTT protocol.

REST
Start messaging with the Solace Messaging REST API.

Connect Using Management Tools
Select a management tool to view connection information and credentials.

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

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



Service Details: **broker for EDA demo**

Get Started with Try Me!

Connect your event broker!

Learn about hierachical topics and wildcards to achieve fine grained management of your events. Learn about Try Me!

Copy your credentials below. Then open Broker Manager and follow the 3 easy steps on the right.

Client Username	solace-cloud-client
Client Password	••••••••••••

Open Broker Manager

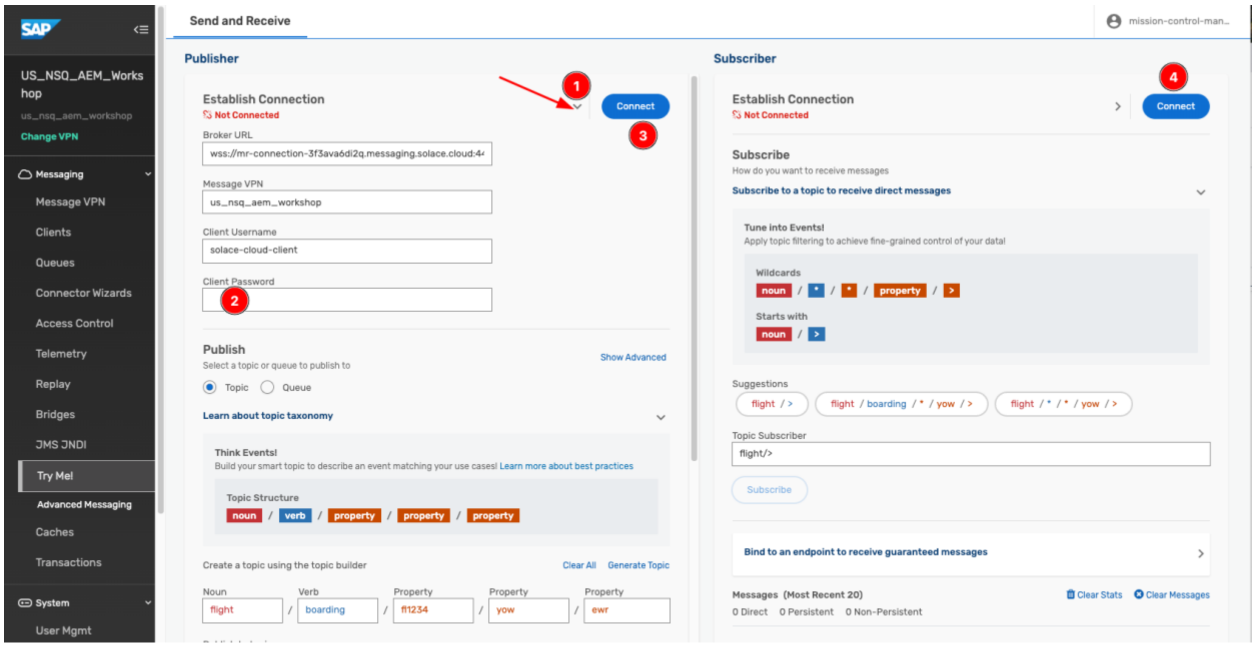
Establish Connection
Not Connected
Broker URL
[pre-filled for you]
Message VPN
[pre-filled for you]
Client Username
Client Password

Publish
Select a topic or queue to publish to
Learn about topic taxonomy

Think Events!
Build your smart topic to describe an event matching your use cases! Learn more about best practices

Topic Structure
noun / verb / property / property / property

Once the Broker Manager is open, select the "Try Me" option from the left side of the menu. You will then use the credentials that you copied above to populate the left side of the screen…AKA the Publisher Side. Once the publisher side says "connected", you can simply hit the "Connect" button on the right side to also connect your subscriber.



Broker URL: wss://mr-connection-3f3ava6di2q.messaging.solace.cloud:44

Message VPN: us_nsq_aem_workshop

Client Username: solace-cloud-client

Client Password:

Select a topic or queue to publish to: [✓ Topic] Queue

Think Events! Build your smart topic to describe an event matching your use cases! Learn more about best practices

Topic Structure: noun / verb / property / property / property

Create a topic using the topic builder — Clear All — Generate Topic

Noun	Verb	Property	Property	Property
flight	boarding	fl1234	yow	ewr

How do you want to receive messages

Tune into Events! Apply topic filtering to achieve fine-grained control of your data!

Wildcards: noun / * / * / property / >

Starts with: noun / >

Suggestions: flight / > — flight / boarding / * / yow / > — flight / * / * / yow / >

Topic Subscriber: flight/>

Messages (Most Recent 20): 0 Direct 0 Persistent 0 Non-Persistent

You are now connected to the AEM service with a publisher and subscriber that can be used to send/receive messages.