



Unleashing the Power of Event-Driven Integrations with Advanced Event Mesh

SAP TECHNICAL ACADEMY

THE BEST RUN 

Table of Contents

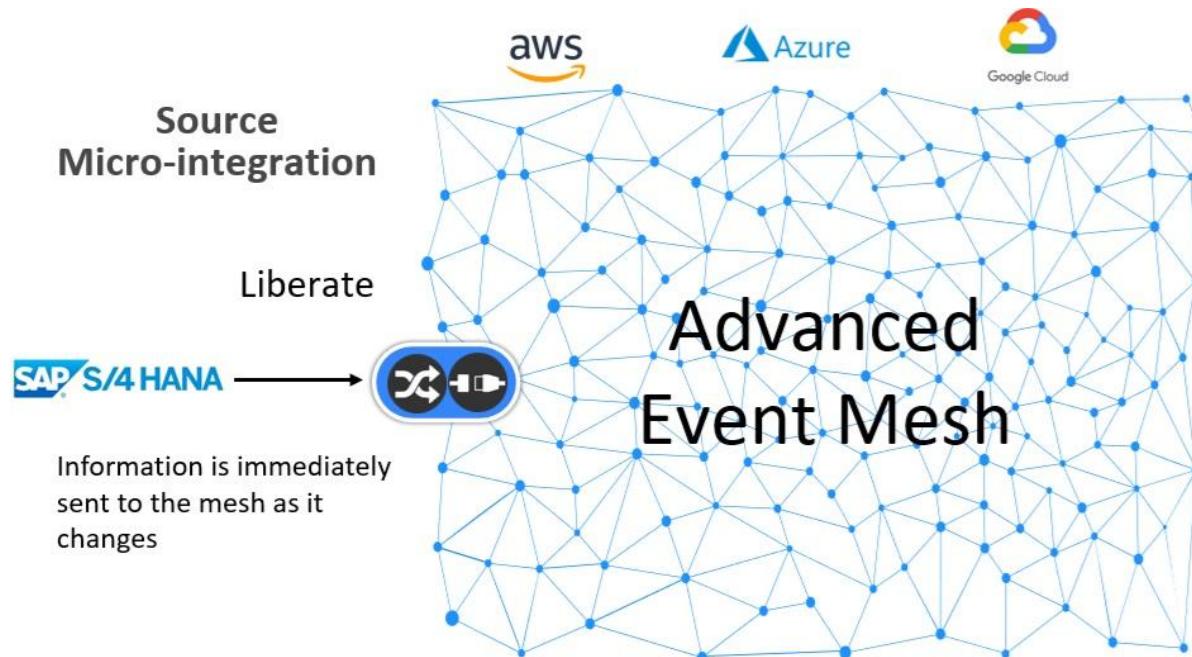
1	<u>INTRODUCTION TO SMART TOPICS</u>	2
1.1	SETTING UP OUR ENVIRONMENT.....	2
1.2	TOPICS 101	4
1.3	HANDS ON WITH AEM AND TOPICS	5
2	<u>AEM AND QUEUES</u>	8
3	<u>CONNECTING TO THE TRY ME UTILITY</u>	9
3.1	OBTAIN CONNECTION CREDENTIALS	9

1 Introduction to Smart Topics

1.1 Setting up our Environment

In this exercise 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 SAP Business Process Automation or Cloud Integration 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 you will connect to the broker mesh that you are using today.

We start by heading to the site <http://feeds.solace.dev>.

Community Feeds

The grid contains the following feeds:

- Aviation - ACARSEngine** by Solace Community (AVIATION)
- Barista Station** by Giri Venkatesan (RETAIL)
- Core Banking** by Hari-Rangarajan-Solace (BANKING)
- CRMSystem** by Giri Venkatesan (CRM)
- F1 Telematics** by Tamimi AlTamimi (AUTOMOTIVE)
- Fabrication Lab** by Tamimi (MANUFACTURING)
- Fraud Detection** by Hari-Rangarajan-Solace (BANKING)
- Mining - HR Service** by Solace Community (MINING)
- Omnichannel-Feed** by Swen@Solace (RETAIL)
- Oracle GoldenGate Streaming** by Alessandro Cagnetti (RETAIL)
- Point of Sale System** by Giri Venkatesan (RETAIL)
- SAP ERP** by Alessandro Cagnetti (Enterprise Resource Planning)
- Sentiment Analysis with AI** by Matt Mays (RETAIL)
- Shipping Service** by Tamimi (SHIPPING)

From this layout, you can see many feeds that you can experiment with but for this exercise, we will be using the SAP ERP Feed.

When you click into it, you will see the option to connect to a broker.

For the connection details, you will take your broker connection details for the WebMessaging protocol and provide them here. Once you are connected, you can then select the Events you would like to test.

Configure Broker

* URL: ws://localhost:8008 VPN: default Username: default Password:
 Direct Guaranteed Text Message Bytes Message

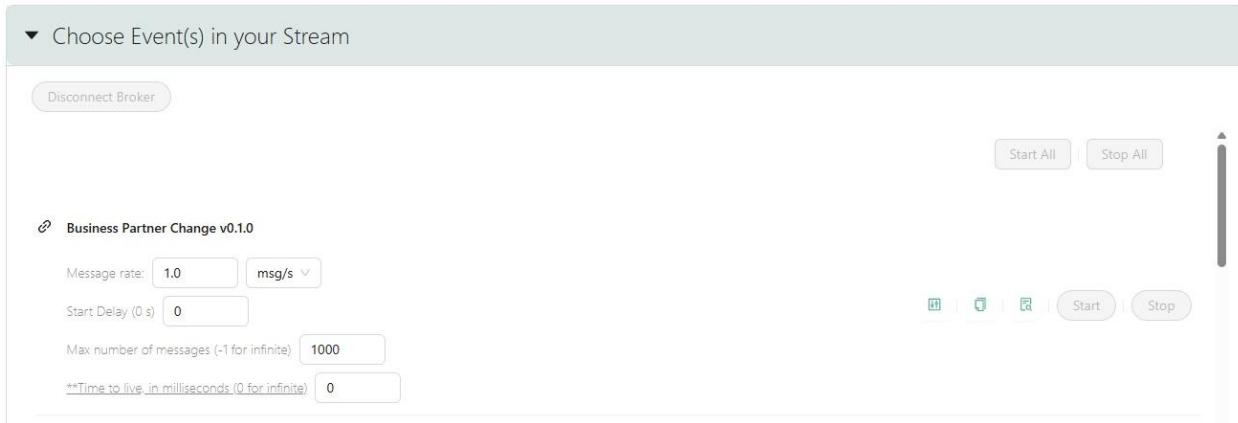
To get started with creating a solace PubSub+ event broker follow the instructions on [Try PubSub+ Event Brokers page](#).

Choose Event(s) in your Stream

Disconnect Broker Start All Stop All

Business Partner Change v0.1.0
<sap/erp/businesspartner/change/V1/{businessPartnerType}/{partnerId}>

For the workshop, we have activated the Business Partner Change and the Business Partner Create and this will be the 2 events that we are “simulating” from an ERP perspective.



For example, if you had a website or vendor portal that allows your customers to change their address for example, you might want to publish that Address Change as an event so you could send that customer an immediate email to confirm the change was made and the address reflects accurately what the customer wanted. For the purposes of the workshop, we will now use this setup to help you understand how to subscribe dynamically to this information coming out of SAP.

1.2 TOPICS 101

As we walk into the next section, it's useful to understand the basic structure of a topic and how it should be constructed. As you can see in the diagram below, our best practices suggest to use a “NOUN”, followed by “VERB” and (eg.SALESORDER/CREATED) then a set of properties that you choose depending on how you would like to access the data later. The noun and the verb are typically fixed and/or static and the properties that follow the static portion are usually populated at runtime.

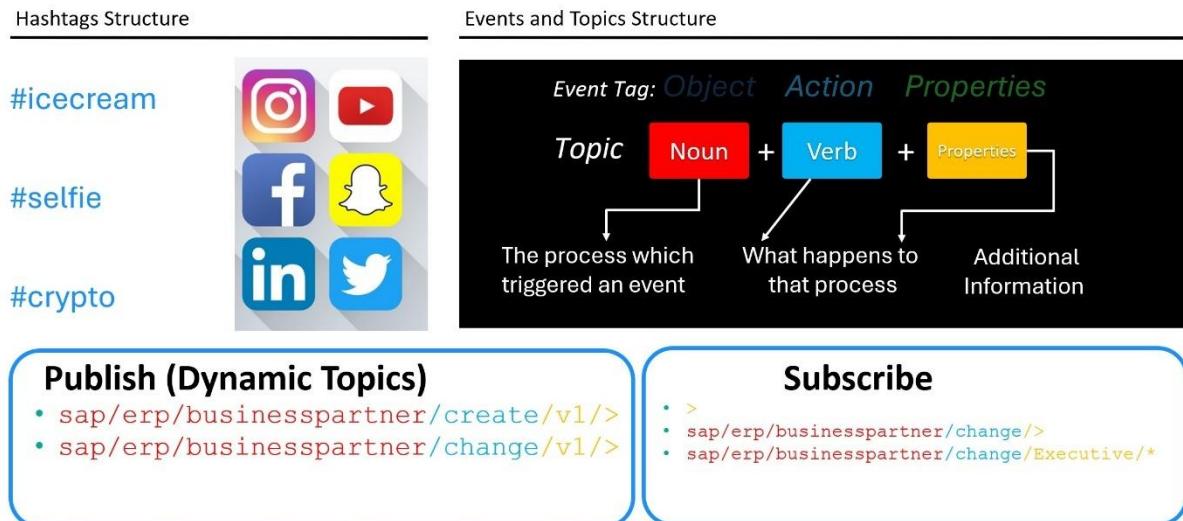
So, in the base of our example below all events published related to business partner would have the first section of their Topic as “sap/erp/businesspartner/create/v1” as the static portion and what follows would be the dynamic properties.

Properties for a business partner could include city/state/zip code, lat, long, etc...this would allow you to later subscribe and filter on customers in a certain city or geographic location as 2 easy examples.

Additionally, the Advanced Event Mesh service allows you to also use wildcards which will be seen below.

Everything that occurs is an event

What if you could “tune in” to specific kinds of events occurring across your enterprise and across clouds?



1.3 Hands On with AEM and Topics

1.3.1 Introduction to Wild Cards

Your first challenge will be to create a subscription to the Advanced Event Mesh that will attract any Business Partner that has been created or changed.

The events that are being generated from the Simulator have the following topic structure:

sap/erp/businesspartner/change/V1/{businessPartnerType}/{partnerId}

sap/erp/businesspartner/create/V1/{businessPartnerType}/{partnerId}

In your first challenge, you will want to use the TRY ME tab on the broker to connect and create a subscription that will show you all created or changed Business Partners. For this we will introduce 2 wildcards:

‘*’ is used to match pieces of data at a certain level. For example, if I wanted all events that matched any Version number I could create a subscription that looks like this:

sap/erp/businesspartner/change/*/Executive/50

So in this case, you can see below the topic structure and we have placed “*” in the position where the Version number would appear.

sap/erp/businesspartner/change/V1**/**{businessPartnerType}**/**{partnerId}****

So in this case, the asterisk is being used to attract events that have any version number and also just Executives with the ID of 50. Important to note that the asterisk only matches that level.

'>' is used to match any and all levels that follow it.

sap/erp/>

So in this case, a subscription like this would attract any and all events that have a topic starting with sap/erp and it would match all business partners, sales orders, purchase orders, etc...

The interesting thing is that you can use these in combination with each other so you can have something like:

sap/erp/*/change/>

What exactly would this Subscription attract?

1.3.2 Subscription Challenge #1

Now, lets navigate to the “TRY ME” tab to try our First Subscription which will attract any Business Partner that has been created or changed in the SAP System. Use the instructions contained in the Appendix to connect and return here [CONNECTING TO THE TRY ME UTILITY](#)

For starters, lets understand how to use the wild card to subscribe to every single event coming into the AEM service.

Once you have successfully connected to the TRY ME facility as per the instructions located in the Appendix, enter a “>” into the “Topic Subscriber” input box and press enter.

The screenshot shows the SAP TRY ME interface with two main sections: Publisher and Subscriber.

Publisher Section:

- Establish Connection:** Shows a green "Connected" status.
- Publish:** Allows selecting a topic or queue to publish to. The "Topic" radio button is selected.
- Learn about topic taxonomy:** A dropdown menu.
- Topic Structure:** Displays a path: noun / verb / property / property / property. The first "noun" field contains "flight".
- Create a topic using the topic builder:** Fields for Noun (flight), Verb (boarding), Property (ff1234), Property (yow), and Property (ewr). Buttons for "Clear All" and "Generate Topic".
- Publish to topic:** A text input field containing "flight/boarding/ff1234/yow/ewr".
- Delivery Mode:** A dropdown menu.

Subscriber Section:

- Establish Connection:** Shows a green "Connected" status.
- Subscribe:** Allows selecting a topic to receive direct messages. The "Subscribe to a topic to receive direct messages" dropdown is open.
- Tune into Events!**: A section for applying topic filtering with a "Wildcards" field containing "noun / * / * / property / >".
- Suggestions:** A list of suggestions: "flight / >", "flight / boarding / * / yow / >", and "flight / * / * / yow / >".
- Topic Subscriber:** An input field containing ">".
- Subscribe:** A blue button.

Once you have subscribed as shown in the next screenshot, you should immediately see events appearing in the window towards the bottom of the screen. If so, you have dynamically subscribed to the mesh and had your first experience with using a Topic Subscription. It's REALLY important to recognize that the TRY ME tab is not doing the

filtering here, is the AEM service. AEM is only sending events to your “Try Me” tab that match the subscription you have entered.

The screenshot shows the AEM Send and Receive interface. On the left, the Publisher section is active, showing a connection to 'montrealbroker-10-1...'. Under 'Establish Connection', it says '% Connected'. In the 'Publish' section, 'Topic' is selected. The 'Topic Structure' is shown as noun / verb / property / property / property. Below this, a topic builder shows 'flight' / 'boarding' / '#1234' / 'yow' / 'ewr'. The 'Published to topic' field contains 'flight/boarding/#1234/yow/ewr'. The 'Delivery Mode' dropdown is set to 'Direct'. On the right, the Subscriber section shows 'Subscribed Topics' with 'sap/erp/businesspartner/>' highlighted with a red box. The 'Messages (Most Recent 20)' section shows a single message from '2025-07-23 11:32:39:701 [Topic sap/erp/businesspartner/create/V1/Assistant/62]'. The message content is: {businessPartner:{'firstName':'Janelle','lastName':'Smitham','addressLink':{'address':{'country':'AU','nation':'AW','city':'South Litz'},'street':'Yost Squares','postalCode':'17569','houseNumber':'2438','addressNumber':449,'dateFrom':'Tue Jul 22 2025'}}, which is also highlighted with a red box.

1.3.3 Subscription Challenge #2

So, now try to create a subscription that only attracts Business Partners that have been changed or created. Resist the temptation to look below this line and see the answer 😊

This screenshot is identical to the one above, showing the Publisher section with a connection to 'montrealbroker-10-1...'. The 'Topic' is selected, and the 'Topic Structure' is noun / verb / property / property / property. The 'Published to topic' field is 'flight/boarding/#1234/yow/ewr'. The 'Delivery Mode' is 'Direct'. On the right, the Subscriber section shows 'Subscribed Topics' with 'sap/erp/businesspartner/>' highlighted with a red box. The 'Messages (Most Recent 20)' section shows a message from '2025-07-23 11:37:30:690 [Topic sap/erp/businesspartner/create/V1/Architect/68]'. The message content is: {businessPartner:{'firstName':'Mustafa','lastName':'Runolfsson','addressLink':{'address':{'country':'SO','nation':'CW','city':'Boynton Beach'},'street':'Walter Lake','postalCode':'02266-1578','houseNumber':676,'addressNumber':8619,'dateFrom':null}}, which is also highlighted with a red box.

So, if you remove the previous subscription and enter a new subscription with this format :

sap/erp/businesspartner/>

You should see all created and changed business partners arrive in the window.

1.3.4 Subscription Challenge #3

Create a subscription that only attracts Business Partners that have changed with a business partner type of “Developer”.

Publisher

Establish Connection
% Connected

Publish
Select a topic or create a new one matching your use case! Learn more about best practices

Learn about topic taxonomy

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

Topic Structure
`root / verb / properties / property / property`

Create a topic using the topic builder

Noun	Verb	Property	Property	Property
flight	/ boarding	/ #1234	/ you	/ out

Clear All **Generate Topic**

Publish to topic
`flight/boarding/#1234/you/out`

Delivery Mode
 Direct Persistent

Message Content

```
{ "action": "exampleAction", "region": "Ontario", "requester": "44444", "flight": "Y", "flightMode": "boeing737"}
```

Subscribe
How do you want to receive messages?

Subscribe to a topic to receive direct messages

Turn into Event!
Apply Topic filtering to achieve fine-grained control of your data!

Wildcards
`root / * / * / property / *`

Starts with
`root / *`

Suggestions
`flight /` `flight / boarding /` `/ you /` `flight / * / you /`

Topic Subscriber

Subscribe

Subscribed Topics
`sage@businessuser1@change/Y/Developer/` **X**

Bind to an endpoint to receive guaranteed messages

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

`2023-07-23 12:03:28.697 [Topic subscriber/businessuser1@change/Y/Developer/995]`

Delivery Mode: Direct
JMS Header: Yes
Time To Live (sec): 0

Clear Stats **Clear Messages**

Now, if you remove the previous subscription and replace it with the following subscription:

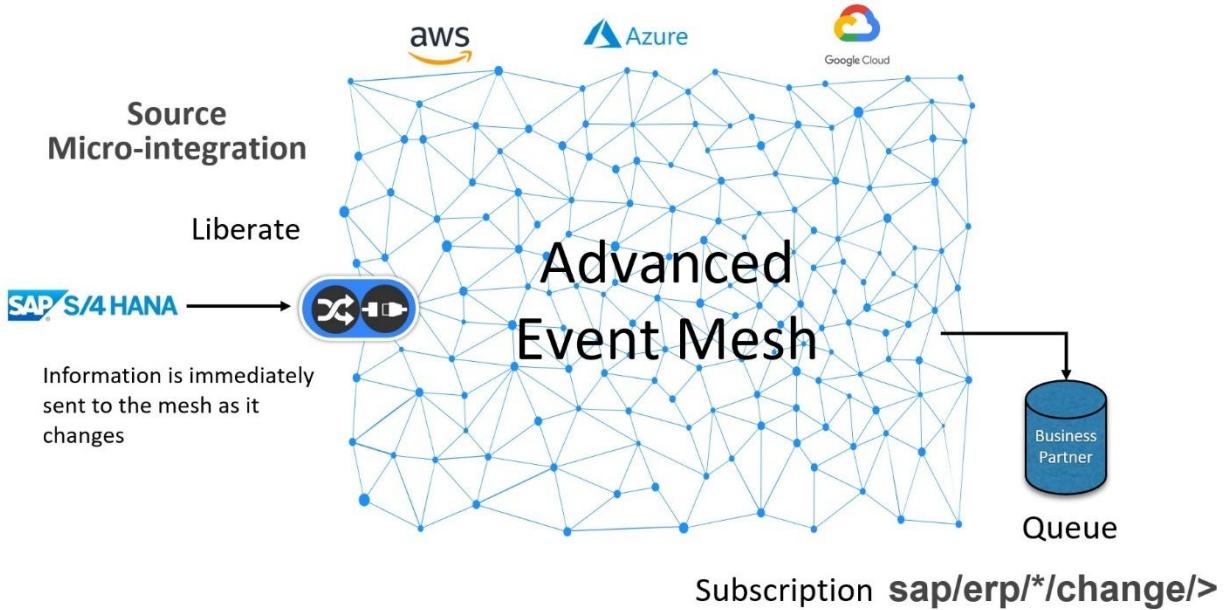
sap/erp/businesspartner/change/V1/Developer/>

You should only see events coming into the Try Me window that are changed developer business partner records.

The use of Smart Topics allows you to filter/stream exactly the information you need and the filtering is handled completely by the AEM service so that filtering is not necessary on the client. This makes the construction of the Topic Taxonomy extremely important.

2 AEM and Queues

Queues can be used by customers to ensure guaranteed delivery of messages in circumstances where delivery of messages might not be possible. For example, what if the target system has been shutdown for maintenance. For example, perhaps you have an application that needs to be updated with the new business partner record but it has been stopped for maintenance. Without taking the necessary steps, records could potentially be lost. This is one situation where Queues can be really helpful.



A Queue will be created and uses the same Subscription concept to pull the messages into a staging area and will persist them safely in order until the subscribing application is available to receive them. In cases like this, clients are consuming the events from the queue and not consuming based on Topics. This is an important concept and will be illustrated below as a way to guarantee delivery.

3 CONNECTING TO THE TRY ME UTILITY

3.1 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.

Open your Advanced Event Mesh console and click on **Cluster Manager**

Welcome to SAP Integration Suite, advanced event mesh

Subaccount ID: 47ad3af1-3d8a-40ca-8500-4cc0d0aef9e3

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**: View metrics for your event use in your event driven architecture

Next, select one of the brokers that you have created for this workshop.

Default | Cluster Manager

Cluster Manager: Services

All services

Service	Description	Status
AWS Basic_Trail_AEM eks-eu-central-1a	Broker 1k sivaramaju.danthuluri@sap.com	Running
AWS broker for EDA demo eks-us-east-1a	Broker 100 AC791299.00@sapexperienceacademy.com	Running
AWS US_NSQ_AEM_Workshop eks-us-east-1a	Broker 100 Patrick.Bollinger	Running

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

- Solace Messaging
- Solace Web Messaging **3**
- AMQP
- MQTT
- REST

Connect Using Management Tools

- SEMP - REST API
- Broker Manager - Web Application

Solace JavaScript API

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

Host URIs

Secured Web Messaging URI: <wss://mr-connection-3f3avedb2q.messaging.solace.cloud:443>

Get Started

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

Try Me

Connect Using a Supported Client Library

- Solace Messaging
- Solace Web Messaging
- AMQP
- MQTT
- REST

Connect Using Management Tools

- SEMP - REST API
- Broker Manager - Web Application

Service Details: broker for EDA demo

Try Me

Open Broker Manager

Publisher

Establish Connection **Not Connected**

Broker URL: [pre-filled for you]

Message-VPN: [pre-filled for you]

Client Username: _____

Client Password: *****

Try Me

Open Broker Manager

Broker Manager

1 Expand 3 Connect

Paste

Try Me

Learn about topic taxonomy

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.

The screenshot shows the SAP Broker Manager interface with the "Send and Receive" tab selected. On the left, the navigation menu includes "Change VPN", "Messaging" (with sub-options like Message VPN, Clients, Queues, Connector Wizards, Access Control, Telemetry, Replay, Bridges, JMS JNDI), "Try Me" (selected), "Advanced Messaging", "Caches", "Transactions", and "System" (with User Mgmt). The main area is divided into "Publisher" and "Subscriber".

Publisher Side:

- "Establish Connection" status: Not Connected (circled with red number 1).
- Broker URL: `wss://mr-connection-3f3ava6di2q.messaging.solace.cloud:443`
- Message VPN: `us_nsq_aem_workshop`
- Client Username: `solace-cloud-client`
- Client Password: `(REDACTED)` (circled with red number 2).
- "Publish" section: Topic selected (radio button circled with red number 3).
- "Topic Structure": `noun / verb / property / property / property`
- "Create a topic using the topic builder": Fields for Noun (flight), Verb (boarding), Property (fl1234), Property (yow), and Property (ewr).

Subscriber Side:

- "Establish Connection" status: Not Connected (circled with red number 4).
- "Subscribe" section: "Subscribe to a topic to receive direct messages" dropdown.
- "Tune into Events!" section: Wildcards and Suggestions (e.g., flight />, flight / boarding / * / yow />, flight / * / * / yow />).
- "Topic Subscriber": `flight/>`
- "Bind to an endpoint to receive guaranteed messages" section.
- "Messages (Most Recent 20)": Clear Stats, Clear Messages buttons.

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