

# Integrating Systems to “Talk Smarter”

*Event-Driven Architecture, SAP BTP and Advanced Event Mesh in Action*

SAP INSIDE TRACK  
İSTANBUL 2025



Barış Büyüktanır | | Deniz Zilyas

May'2025

# Agenda

- **Walk through SAP's Integration and Event-Driven Capabilities**
- SAP BTP Integration Suite: “**talking smarter**”
- Event-Driven Architecture “**in action**” with Advanced Event Mesh
- Recap and Roadmap for SAP BTP Integration & AEM
- Final words and key takeaways

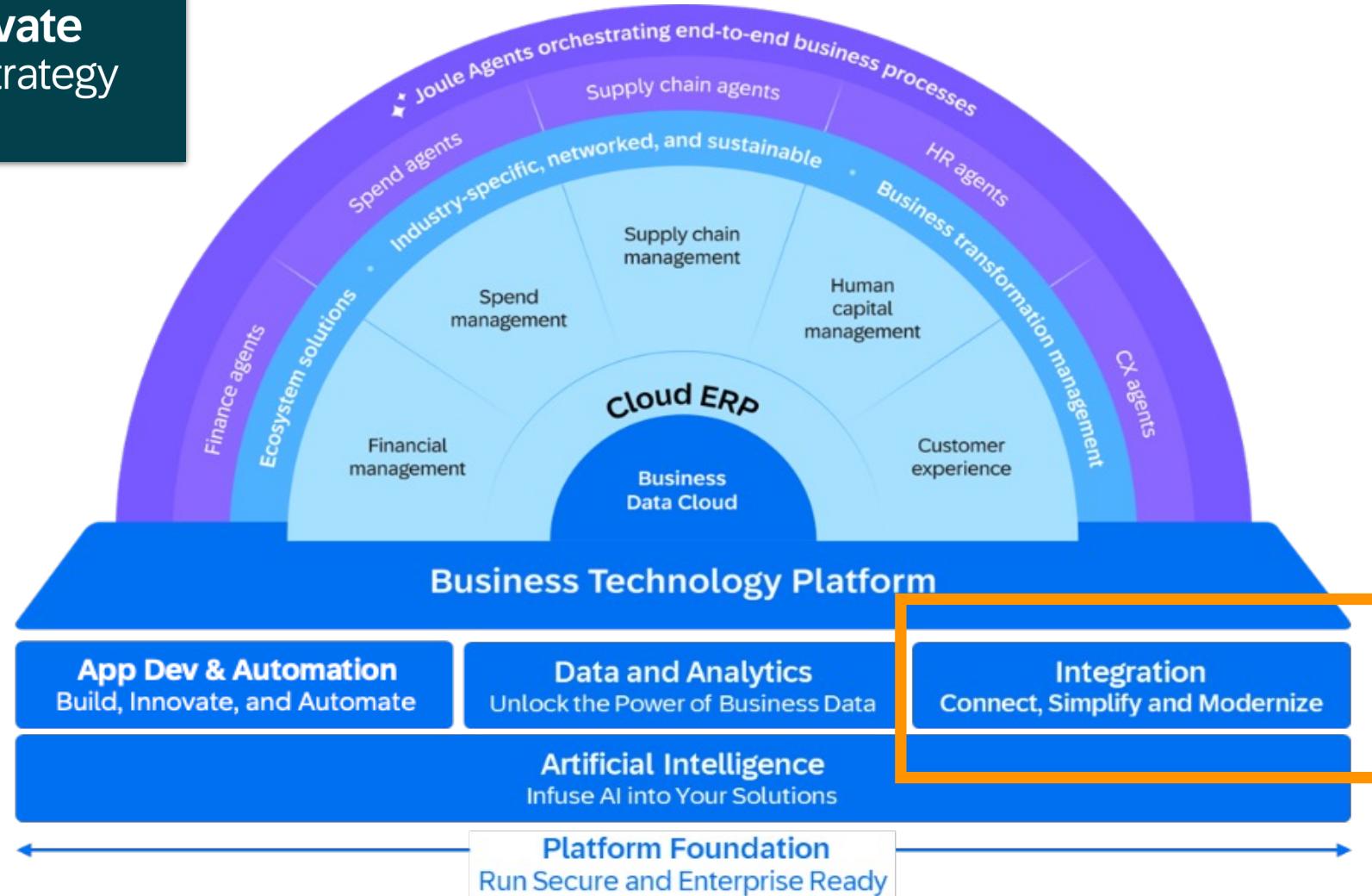




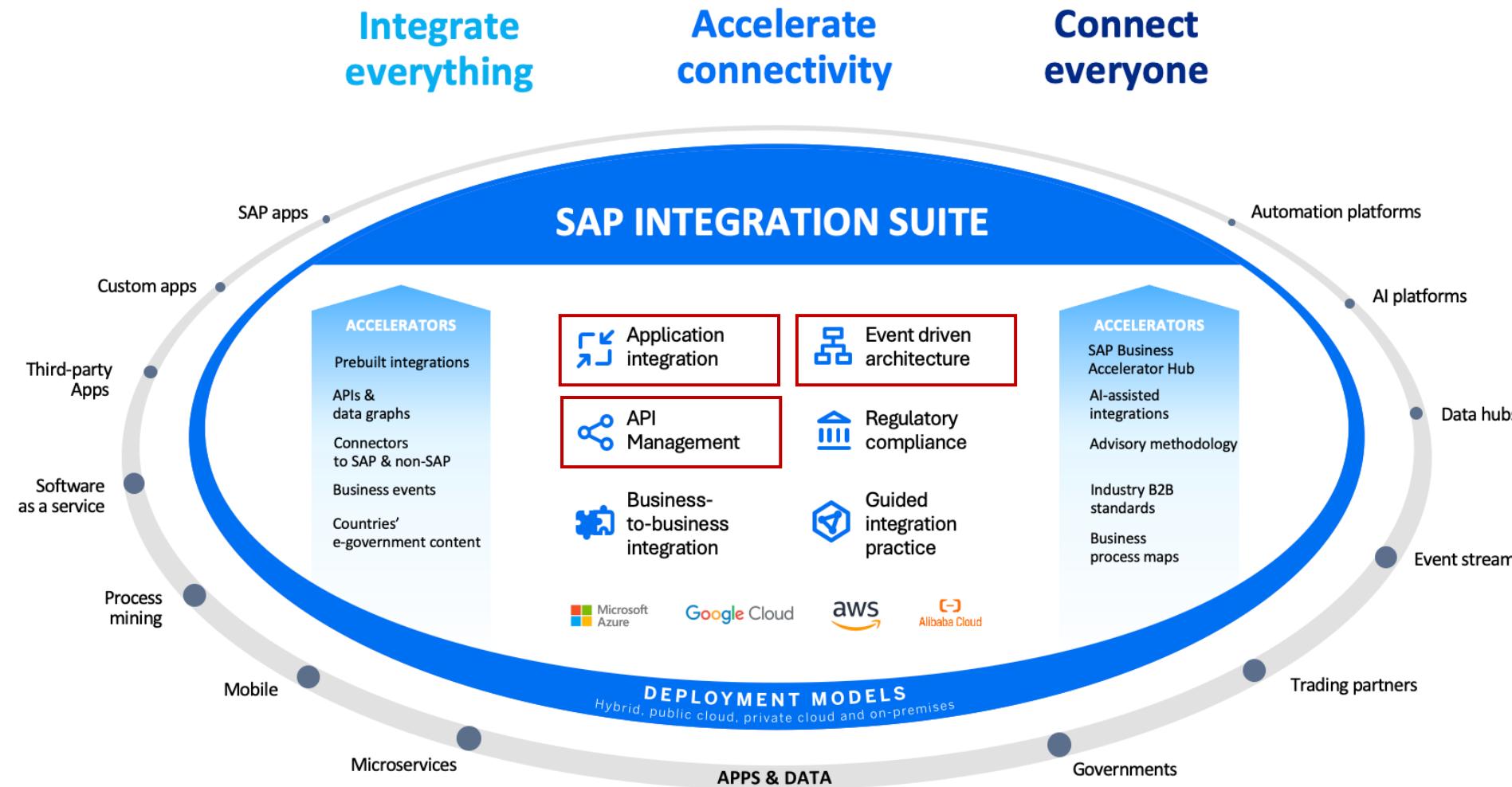
# Walk Through SAP's Integration and Event-Driven Capabilities

# SAP's Integration Strategy & BTP Integration Capability

**Integrate to Innovate**  
SAP's Integration Strategy  
White Paper

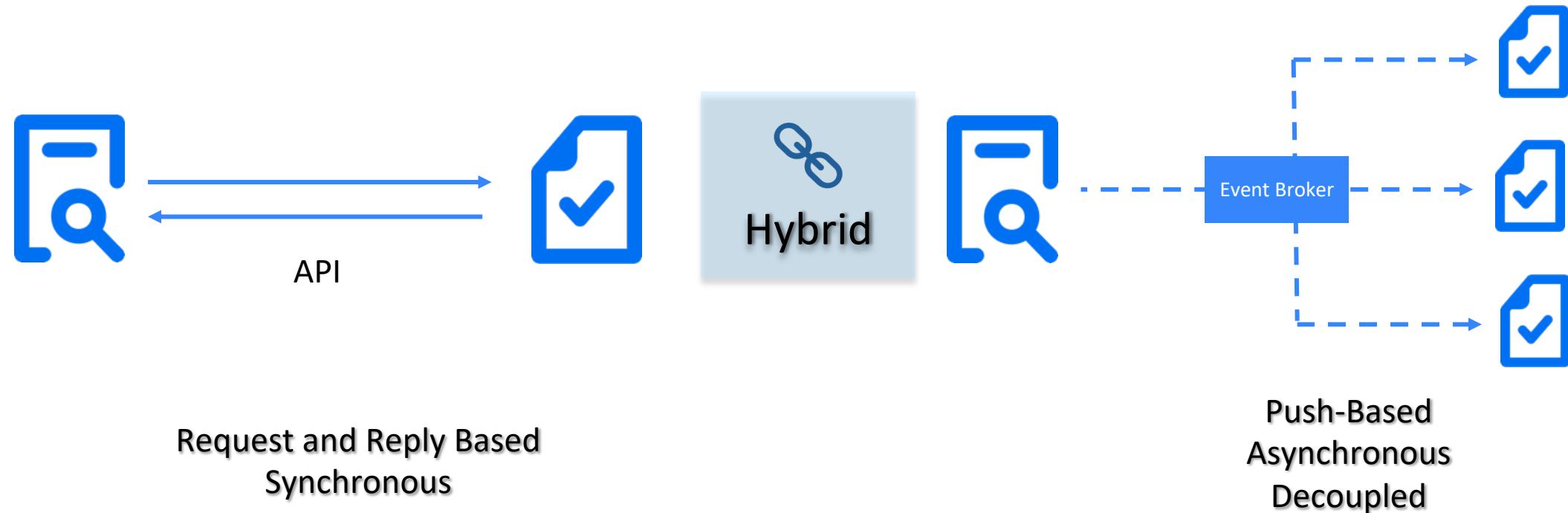


# Integration via BTP : SAP Integration Suite



# API-led classical, Event-Driven and Hybrid Integrations

## *When to use which*

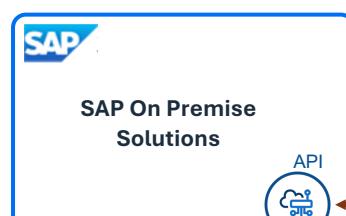


# Why Event-Driven ? – too many reasons

- Loose coupling of systems and applications (**minimum dependencies**)
- (Near)**real time** information published to **distributed systems**.. No polling .. **Everything is event**, every component acts as soon as possible
- **Publish once**, do not stop/wait for other service(s).
- Shifting from **huge batches to small blocks of data** over time
- **Handle slower consumers**, back-off / back-pressure
- Load balance - **scale and fail independently**, add new consumers easily
- No loss of messages, **replay the history** for late joiners / errors
- «**Smart routing**» based on topics

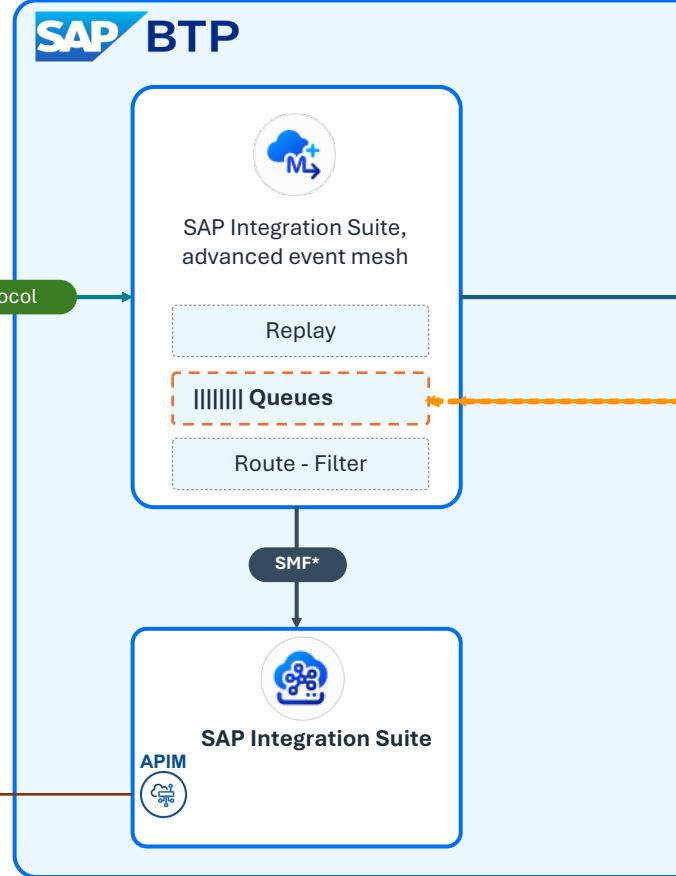
# The Big Picture – Architecture

Upstream(publishers)



Multi protocol

Brokers



2 Enrich via APIs / Apply other policies

Downstream(subscribers)

3 Subscribe



Subscribe

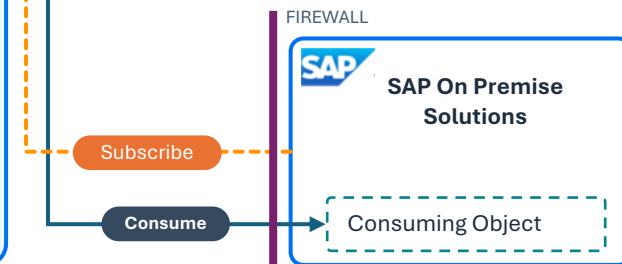
Consume

4 Consume Actively



Subscribe

Consume



Subscribe

Consume

FIREWALL

FIREWALL

Cloud connector

Invoke APIs  
Cache returns

# SAP BTP Integration Suite

## Talking “Smarter” than ever





# SAP's Event Brokers

# SAP Event-Driven Integration Offerings for BTP



Application development and automation

Data and analytics

Extended planning and analysis

Integration

Artificial intelligence



## Advanced Event Mesh

Use an event-streaming and management platform for the real-time enterprise.



## Event Mesh

SAP Event Mesh allows applications to communicate through asynchronous events.



## SAP Event Hub

Flexible solution to distribute business events across the SAP cloud landscape.



FEATURED

## Integration Suite

Simplify and accelerate enterprise integration.

Event Mesh Capability (EM-IS)

Free Tier



# SAP's Event Brokers

Feature / Service	Event Mesh (Standalone Subscription)	Event Mesh (Integration Suite Capability)	Advanced Event Mesh (AEM)	Cloud Application Event Hub (CAEH)
What is it?	Standalone event broker service on BTP	Same broker engine, integrated into Integration Suite	Solace-powered, distributed high-scale event mesh	Lightweight event publisher for SAP Cloud apps
Who is it for?	BTP apps, CPI, external systems	CPI-based integration scenarios with event-driven logic	Global, high-load, enterprise-grade architectures	SAP Cloud apps sending events to 3rd party systems
Usage model	Pub-sub, queue/topic creation, general purpose	Pub-sub, usually tied to CPI flows	Mesh network, geo-distributed, high throughput	Publish-only (Cloud App → external consumer)
Subscriber support	<input checked="" type="checkbox"/> Yes (AMQP, Webhook, Kafka, REST)	<input checked="" type="checkbox"/> Yes (CPI, Webhook, Kafka, REST)	<input checked="" type="checkbox"/> Yes (same + Solace tools)	<input checked="" type="checkbox"/> Yes (REST, Webhook, Kafka, etc.)
Management UI	<input checked="" type="checkbox"/> BTP Cockpit + REST APIs	<input checked="" type="checkbox"/> BTP Cockpit (limited to CPI scenarios)	Solace Admin UI / CLI / APIs	Simple, embedded in SAP Cloud admin tools
Routing & QoS	Durable queues, topic filters, retry logic	Same (in CPI context)	Advanced: replay, mesh routing, guaranteed delivery	Basic delivery, limited routing
Setup & Operations	Separate subscription, full configuration	Automatically provisioned in Integration Suite tenant	High – dedicated brokers with infrastructure setup	Lightweight provisioning, low ops effort
Example Use Case	SAP → Event Mesh → AMQP → 3rd Party	S/4HANA Cloud → CPI (iFlow) triggered by event	Black Friday load handling across regions	SAP Subscription Billing → Kafka or webhook
Key Differentiator	Standalone, reusable across multiple apps	CPI-optimized, fast integration build	Enterprise-grade mesh for massive throughput	SAP SaaS-native, fast external exposure

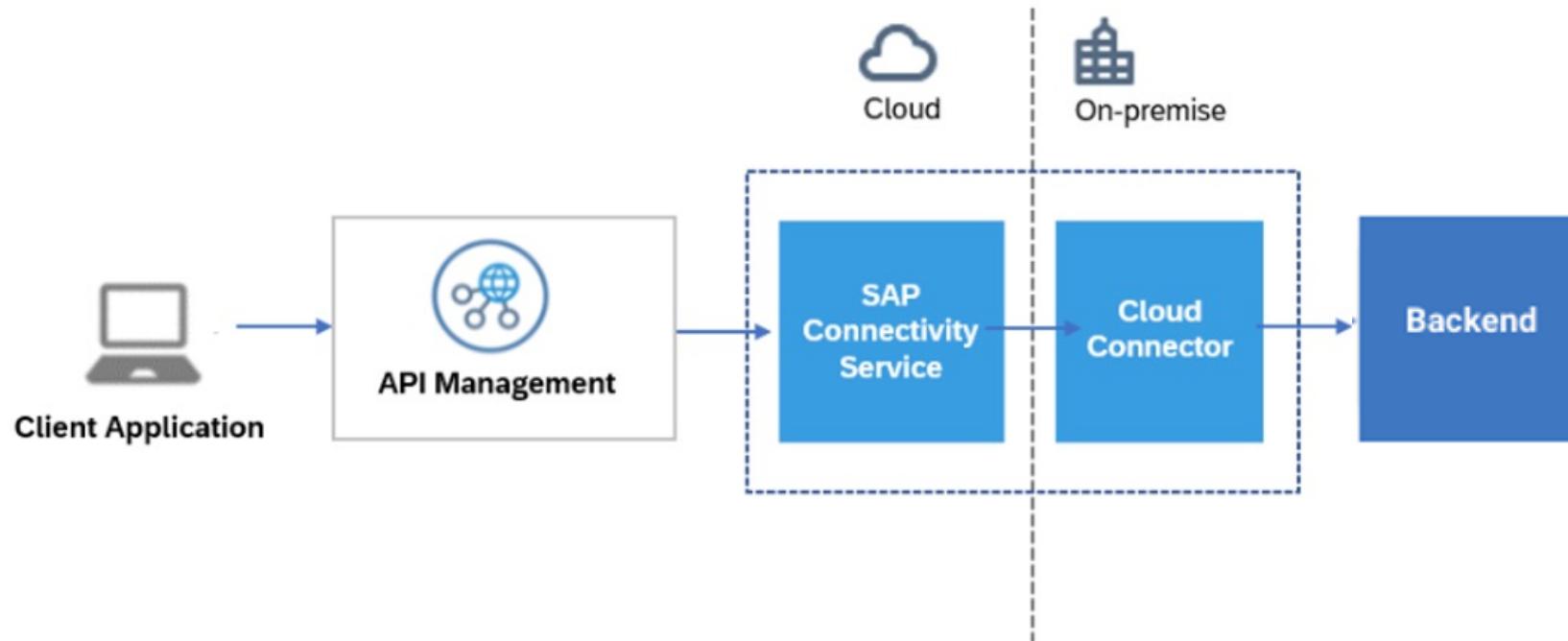


# SAP API Management

“Unlocking the Power of APIs»

# What is API Management?

SAP API Management is an API management solution running on the SAP BTP. Its main purpose is to expose services from on-premise or cloud systems to the outside world (or internal systems) in a secure, scalable, and manageable way.



# The Key to Digital Ecosystems: SAP API Management

## What Does It Do?

**Exposure of backend services**

**API Security**

**Monitoring & Analytics**

**Rate Limiting, Caching, and Traffic Control**

**Versioning of APIs**

## Where Is It Used?

Exposing SAP systems (e.g., **S/4HANA, SF, Ariba**) to partners and third-party applications

**B2B and partner integrations**, mobile apps, and portals

**IoT scenarios** needing secure, controlled access to SAP data

Acting as a **central API gateway** in hybrid system landscapes

Supporting **digital business models** with service-based offerings

# Real-World Use Cases of SAP API Management

## **Response Caching**

Improve performance by caching repetitive requests when backend data changes infrequently.

## **Key-Value Map (KVM)**

Manage dynamic parameters like tokens or endpoints externally using get/set scripting.

## **Quota + Spike Arrest (Rate Limiting)**

Protect public APIs from overload and define per-subscriber call limits.

## **VerifyAPIKey + OAuth2 Security**

Control who accesses your API with robust, app-based authentication policies.

## **Backend Failover / Mock Response**

Ensure business continuity with static or KVM-based fallback responses when backend fails.

## **JavaScript Payload Manipulation**

Enrich responses with dynamic fields like timestamp, clientId, or geo info.

## **Header Injection & Sanitation**

Auto-inject headers (e.g., Authorization, X-Correlation-ID) to enhance security and traceability.

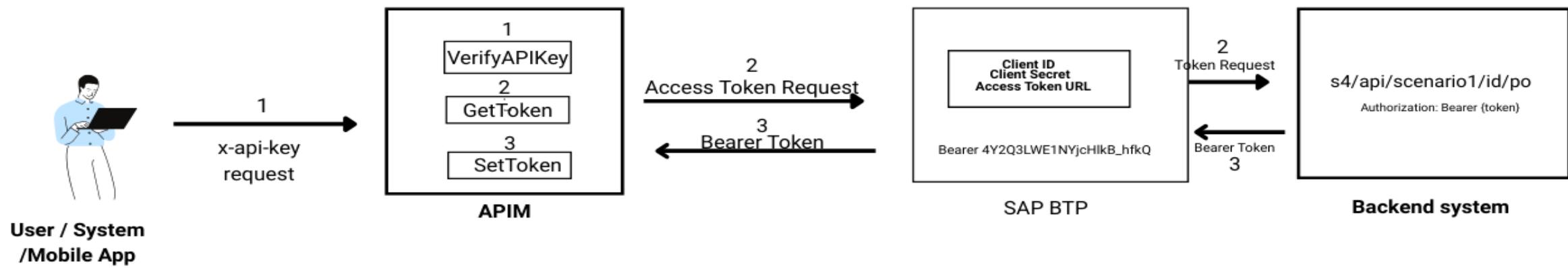
## **Request/Response Validation (JSON/XML)**

Enforce strict schema validation to prevent malformed data from affecting backend systems.

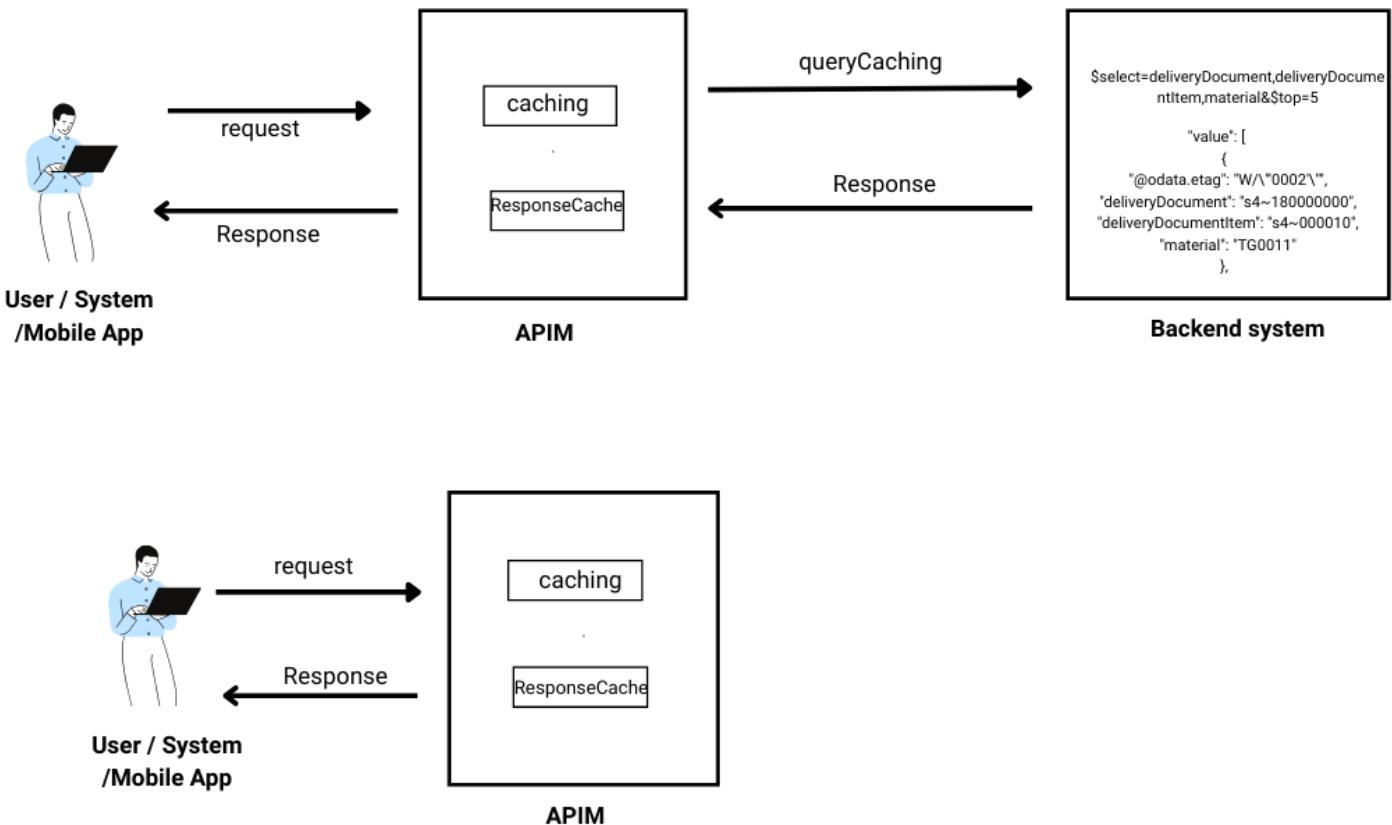
# SAP API Management

**”Fast and Secure: SAP S/4HANA API Consumption with APIM»**

# Fast and Secure: SAP S/4HANA API Consumption with APIM



# Fast and Secure: SAP S/4HANA API Consumption with APIM



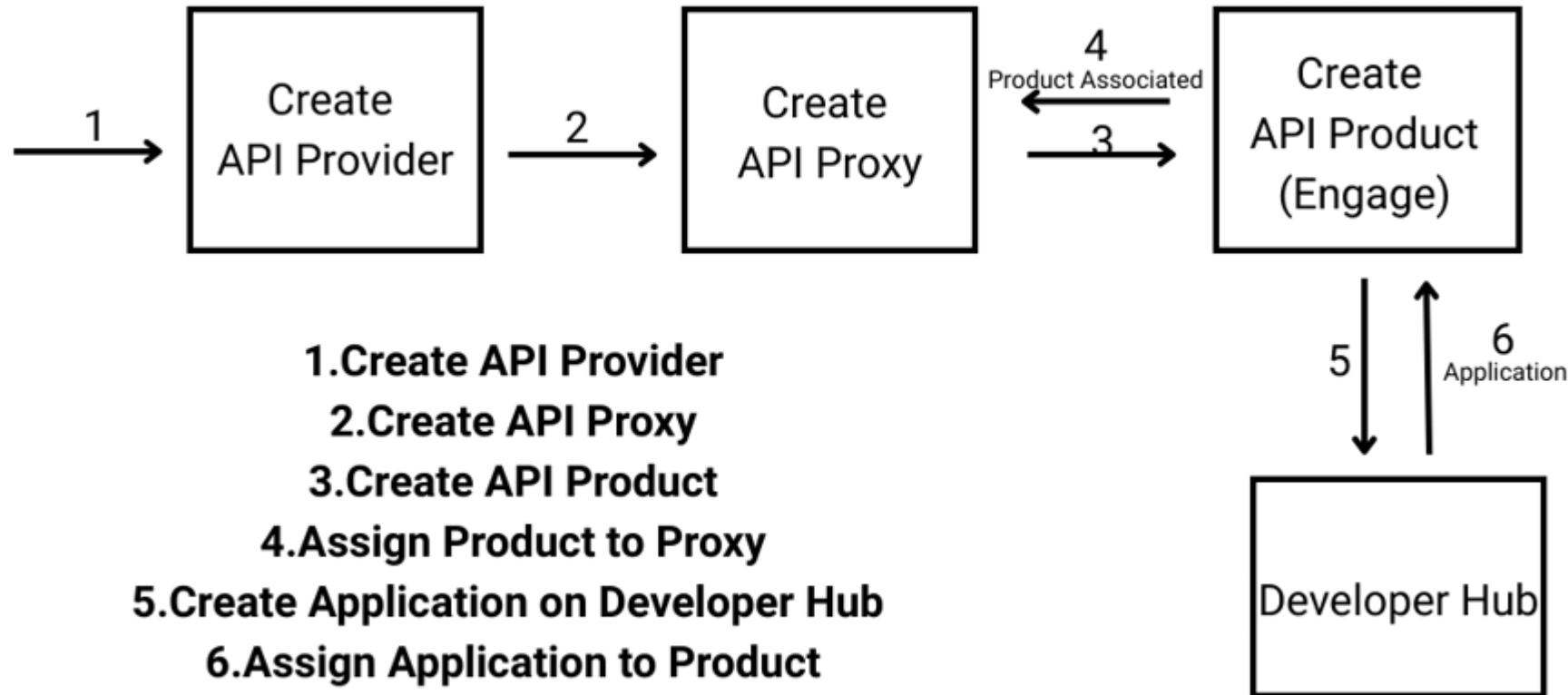
## Purpose

Thanks to caching, we can return responses to frequently repeated queries directly through APIM without calling the backend system.

This not only preserves SAP BTP resources but also allows us to deliver near-instant responses to users.

By defining our own cache duration, we provide second-level access while only engaging the backend system when truly necessary.

# Fast and Secure: SAP S/4HANA API Consumption with APIM



# Fast and Secure: API Management

The screenshot shows the SAP Integration Suite API Management interface. The left sidebar has a navigation menu with the following items:

- Home
- Discover
- Design
- Test
- APIs
  - Configure (selected, highlighted with a red box)
  - APIs (selected, highlighted with a blue box)
- Monitor
- Analyze
- Engage
- Inspect
- Monetize
- Settings

The main content area is titled "Configure" and contains the following sub-sections:

- Create and configure API proxies, API providers, certificates, key-value maps, and policy templates.
- API Proxies (8) (selected)
- API Providers (3)
- Certificates (0)
- Key Value Maps (0)
- Policy Templates (0)

A table lists the configured API proxies:

Name	Title	Status	Type
OAuth_QueryCaching_v1	OAuth_QueryCaching	Deployed	REST
Graph_API_ID-PO_AUTO	Graph_API_ID-PO	Deployed	REST
Graph_API_ID-PO_v1	Graph_API_ID-PO	Deployed	REST
PurchaseOrder_v1	PurchaseOrder	Deployed	ODATA
API-PurchaseOrder_v1	API-PurchaseOrder-Caching	Deployed	REST
validationAPI_v1	validation	Deployed	REST
CurrencyRateAPI_v1	Currency Rate API	Deployed	REST

# Fast and Secure: Create API Proxy

API Proxies (7) API Providers (3) Certificates (0) Key Value Maps (0) Policy Templates (0)

Name	Title	Status	Type	Changed By	Last Updated	Calls	Action
Graph_API_ID-PO_AUTO	Graph_API_ID-PO	Deployed	REST	[REDACTED]	28.04.2025 00:05:07	0	...
Graph_API_ID-PO_v1				[REDACTED]	27.04.2025 23:48:54	0	...
PurchaseOrder_v1				[REDACTED]	27.04.2025 22:55:27	0	...
API-PurchaseOrder_v1				[REDACTED]	27.04.2025 22:37:57	0	...
validationAPI_v1				[REDACTED]	24.04.2025 01:25:48	0	...
CurrencyRateAPI_v1				[REDACTED]	23.04.2025 02:33:20	0	...
HelloWorldAPI				[REDACTED]	23.04.2025 00:17:20	0	...

**Create API**

Select:  API Provider  API Proxy  URL

URL \*

API Details

Name: \* OAuth\_QueryCaching\_v1

Title: \* OAuth\_QueryCaching

Short Text:

API State: \* Active

Host Alias: \*

API Base Path: \*

Version: v1

Service Type: REST

**Create** **Cancel**

# Fast and Secure: Create Product

The screenshot shows the SAP Integration Suite interface. On the left, a navigation sidebar lists various modules: Home, Discover, Design, Test, Configure (expanded to show APIs, Monitor, Analyze, Engage, Inspect, Monetize, and Settings), and Engage (which is highlighted with a red box). The main content area is titled "Create Product" and shows a product named "Caching\_Secure". The product details include:

- Name:** Caching\_Secure
- Title:** Caching\_Secure
- Short Text:** (empty)
- Quota:** 100
- Requests Every:** 10 Month(s) (with a dropdown menu for other time units like Day(s), Week(s), etc.)
- Scope:** (empty)
- Description:** (empty)

Below the description is a rich text editor toolbar with icons for bold, italic, underline, strikethrough, and various alignment and list options.

# Fast and Secure: Create Product

< Create Product      Save As Draft    Publish    Cancel

## Caching\_Secure

Overview APIs (0) Permissions

Name:
-------

Add APIs

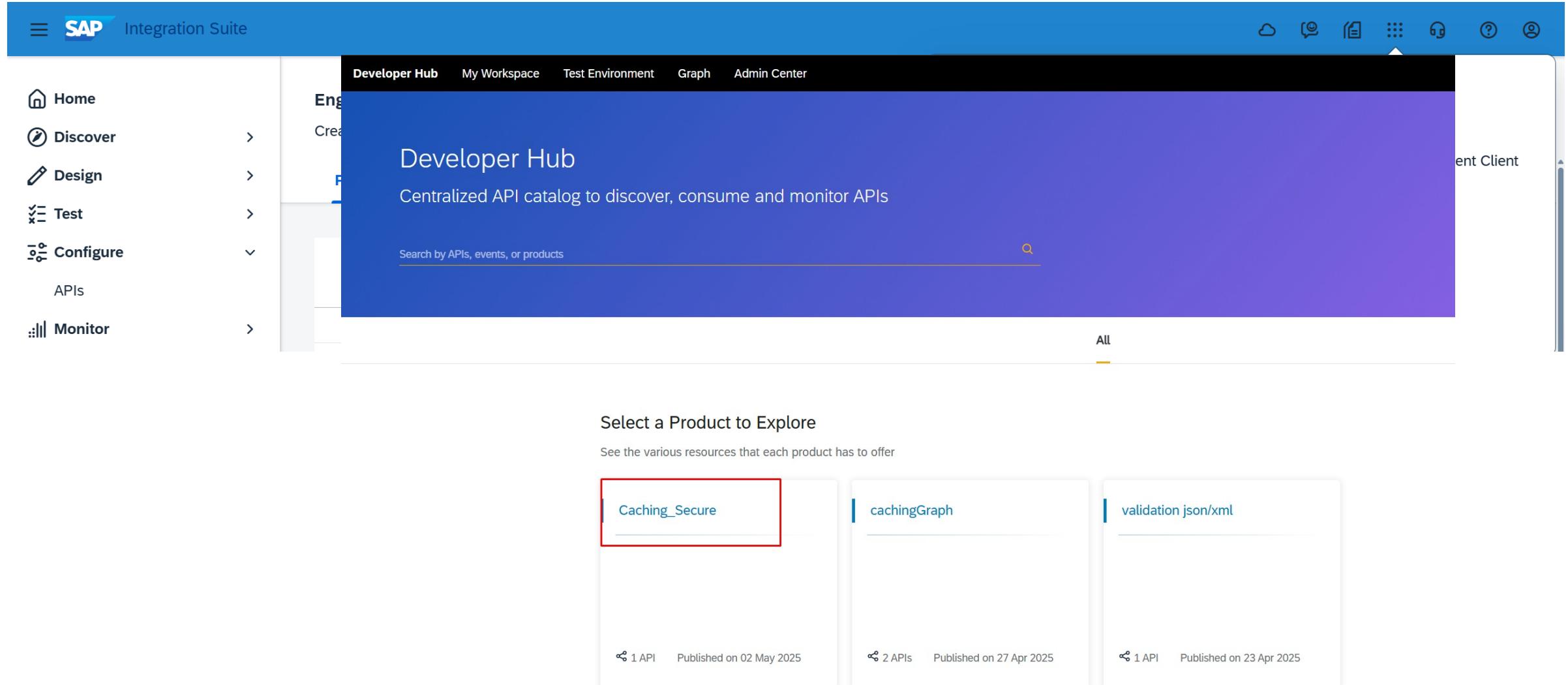
Some of the APIs may have saved changes that are not deployed yet. If such APIs are added to the Product, they will not get published.

	API Name	API State
> <input type="checkbox"/>	API-PurchaseOrder_v1	Deployed
> <input type="checkbox"/>	CurrencyRateAPI_v1	Deployed
> <input type="checkbox"/>	Graph_API_ID-PO_AUTO	Deployed
> <input type="checkbox"/>	Graph_API_ID-PO_v1	Deployed
> <input type="checkbox"/>	HelloWorldAPI	Deployed
<input checked="" type="checkbox"/>	OAuth_QueryCaching_v1	Deployed
> <input type="checkbox"/>	PurchaseOrder_v1	Deployed
> <input type="checkbox"/>	validationAPI_v1	Deployed

**Add** **Remove**

**OK** **Cancel**

# Fast and Secure: Developer Hub



The screenshot shows the SAP Integration Suite interface with the "Developer Hub" selected in the top navigation bar. The main content area displays the "Developer Hub" page, which is a centralized API catalog for discovering, consuming, and monitoring APIs. It features a search bar at the top and a section titled "Select a Product to Explore" below it. Three products are listed: "Caching\_Secure" (highlighted with a red border), "cachingGraph", and "validation json/xml". Each product card includes a small icon, the product name, the number of APIs it contains, and the date it was published.

SAP Integration Suite

Developer Hub My Workspace Test Environment Graph Admin Center

Home Discover Design Test Configure APIs Monitor

Developer Hub

Centralized API catalog to discover, consume and monitor APIs

Search by APIs, events, or products

Caching\_Secure

cachingGraph

validation json/xml

All

Select a Product to Explore

See the various resources that each product has to offer

Caching\_Secure

1 API Published on 02 May 2025

cachingGraph

2 APIs Published on 27 Apr 2025

validation json/xml

1 API Published on 23 Apr 2025

# Fast and Secure: Create Application

The screenshot displays a 'Create New Application' interface with two main sections: 'Application Info' and 'Add Products'.

**Application Info Section:**

- Title:** CachingDemo
- Short Text:** (Empty)
- Description:** (Empty)
- Callback URL:** (Empty)
- Checkboxes:**
  - Create this application on behalf of someone else
  - Custom Application Key & Secret [?](#)
- Checkboxes at the bottom:**
  - Take me to this new application now

**Add Products Section:**

- Selected Products:** Caching\_Secure
- Products:**
  - Caching\_Secure
  - cachingGraph
  - validation json/xml
- Buttons:**
  - Find Products
  - Sort by: Newest on top
  - Create** (button)

**Bottom Buttons:**

- Create** (button)
- Cancel**

# Fast and Secure: Associate with Product

The screenshot shows a web application interface for managing an application named "CachingDemo". The top navigation bar includes a back arrow, the application name "CachingDemo", and an "Edit Application" button with a pencil icon. Below the header, there are tabs for "Application Details", "Products", "Custom Attributes", and "Analytics". The "Application Details" tab is active, indicated by a yellow underline. The "About" section contains fields for "Callback URL:" (empty), "Application Secret:" (redacted), and "Application Key:" (redacted). A red rectangular box highlights the "Application Key" field. To the right, a summary box displays "Created By:" (redacted), "Created On: 02 May 2025", "Calls this month: 0", "Last Modified By:" (redacted), and "Last Modified On: 02 May 2025".

» CachingDemo

Edit Application

Application Details Products Custom Attributes Analytics

About

Callback URL:

Application Secret: zlxfg8ACUZcAOBQ [Regenerate](#)

Application Key: A4tpwcVluFotNTndkO3oP9ir0zL1pxtE [Regenerate](#)

Created By: [REDACTED]

Created On: 02 May 2025

Calls this month: 0

Last Modified By: [REDACTED]

Last Modified On: 02 May 2025

# Fast and Secure: API Proxy Policies

SAP Integration Suite

View API

**Policies** (highlighted with a red box)

**OAuth\_QueryCaching\_v1**

**Overview**   [Proxy EndPoint](#)   [Target EndPoint](#)   [Resources](#)   [Revisions](#)

Title: OAuth\_QueryCaching

Host Alias: [REDACTED]10.hana.ondemand.com

Short Text:

API Base Path: /v1/caching

Version: v1

API State: Active

Description:

Products Associated (1)

Caching\_Secure  
Caching\_Secure

Calls(05/01/2025 - 05/08/2025)  
0

API Health

Key Value Map Associated (0)

No data

Created On: 02.05.2025 23:30:10   Created By: [REDACTED]

Changed On: 02.05.2025 23:34:25   Changed By: [REDACTED]

# Fast and Secure: API Proxy Policies

API Artifacts for Graph\_API\_ID-PO\_AUTO (Draft-1) Edit Policy Template Cancel

Policy Editor

Flows 2 < ProxyEndpoint

PreFlow 7

PostFlow 0

TargetEndpoint: default

Created Policies 0

Scripts 0

verifyAPIKey... caching getToken extractToken... setToken JS

JS caching addServer

verifyAPIKey

Condition String

```
1 | 
2 | <!--Specify in the APIKey element where to look for the variable containing the api key-->
3 | <VerifyAPIKey async='true' continueOnError='false' enabled='true'
4 | xmlns='http://www.sap.com/apimgmt'
5 |   <APIKey ref='request.header.x-api-key'/>
6 | </VerifyAPIKey>
```

Policies ?

Security Policies

- Basic Authentication
- DecodeJWT
- GenerateJWT
- JSON Threat Protection
- OAuth v2.0
- OAuth v2.0 GET
- OAuth v2.0 SET
- Regular Expression Protection
- SAML Assertion Generation
- SAML Assertion Validation
- Verify API Key
- VerifyJWT
- XML Threat Protection

Traffic Management Policies

- Access Control
- Invalidate Cache
- Lookup Cache

```
graph LR
    User((User)) --> Verify[verifyAPIKey...]
    Verify --> Cache1[caching]
    Cache1 --> Get[getToken]
    Get --> Extract[extractToken...]
    Extract --> Set[setToken]
    Set --> JS1[JS]
    JS1 --> User
    JS1 --> Cache2[caching]
    Cache2 --> AddServer[addServer]
    AddServer --> JS1
```

# Fast and Secure: API Proxy Policies

API Artifacts for Graph\_API\_ID-PO\_AUTO (Draft-1) Edit Policy Template Cancel

Policy Editor

Flows ? <>

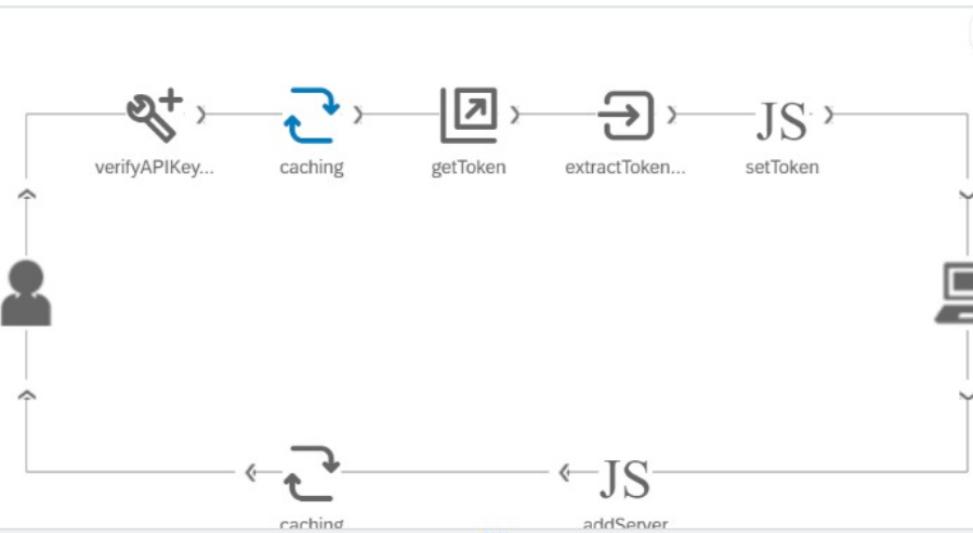
ProxyEndpoint

- PreFlow 7
- PostFlow 0
- 0

> TargetEndpoint: default

> Created Policies ?

> Scripts ?



caching

Condition String ?

```
1 <ResponseCache async="false" continueOnError="false" enabled="true" xmlns="http://www.sap.com/apimgmt">
2   <CacheKey>
3     <KeyFragment ref="request.queryparam.$filter"/>
4     <KeyFragment ref="request.queryparam.$select"/>
5     <KeyFragment ref="request.queryparam.$top"/>
6     <KeyFragment ref="request.queryparam.$expand"/>
7     <KeyFragment ref="request.queryparam.$skip"/>
8     <KeyFragment ref="request.path"/>
9   </CacheKey>
10  <Scope>Exclusive</Scope>
11  <ExpirySettings>
12    <TimeoutInSec>3600</TimeoutInSec>
13  </ExpirySettings>
14
15 </ResponseCache>
```

Policies ?

Security Policies

- Basic Authentication
- DecodeJWT
- GenerateJWT
- JSON Threat Protection
- OAuth v2.0
- OAuth v2.0 GET
- OAuth v2.0 SET
- Regular Expression Protection
- SAML Assertion Generation
- SAML Assertion Validation
- Verify API Key
- VerifyJWT
- XML Threat Protection

Traffic Management Policies

- Access Control
- Invalidate Cache
- Lookup Cache
- Populate Cache
- Quota
- Reset Quota

# Fast and Secure: Testing the Endpoint with Postman

The screenshot shows the Postman application interface for testing an API endpoint. The request URL is set to `/api/v1/deliveryDocument?select=deliveryDocument,deliveryDocumentItem,material,headerGrossWeight,purchasingGroup&$filter=purchasingGroup eq '002'`. The Headers tab is selected, displaying the following configuration:

Key	Description	Value
Postman-Token	<calculated when request is sent>	
Host	<calculated when request is sent>	
User-Agent	PostmanRuntime/7.43.4	
Accept	*/*	
Accept-Encoding	gzip, deflate, br	
Connection	keep-alive	
Accept-Encoding	gzip, deflate	
x-api-key	mvBoJnOvLY2q52GpecDShJ5dxfIV5rGM	

The response status is 200 OK, with a duration of 4.09 s, a size of 1.47 KB, and a timestamp of 2025-05-02T21: 04: 11.603Z. The response body is displayed in JSON format:

```
1 2025-05-02T21: 04: 11.603Z{  
2   "@odata.context": "$metadata#po(deliveryDocument,deliveryDocumentItem,material,headerGrossWeight,purchasingGroup)",  
3   "value": [  
4     {  
5       "deliveryDocument": "s4~180000000",  
6       "deliveryDocumentItem": "s4~000010",  
7       "material": "TG0011",  
8       "headerGrossWeight": 10,  
9       "purchasingGroup": "002"  
10      },  
11      {  
12        "deliveryDocument": "s4~180000000",  
13        "deliveryDocumentItem": "s4~000010",  
14        "material": "TG0011",  
15        "headerGrossWeight": 10,  
16        "purchasingGroup": "002"  
17      }  
18    ]  
19  ]  
20}  
21
```

# Fast and Secure: Testing the Endpoint with Postman

The screenshot shows the Postman application interface with a successful API request. The URL in the header is: `http://192.168.1.102/sap/opu/odata/sap/bc/odata/v1/odata.svc/$entityset?&$select=deliveryDocument,deliveryDocumentItem,material,headerGrossWeight,purchasingGroup&$filter=purchasingGroup eq '002'`. The Headers tab is selected, showing various headers including `x-api-key` with the value `mvBoJnOvLY2q52GpecDShJ5dxfIV5rGM`. The response status is `200 OK` with a response time of `431 ms` and a size of `1.47 KB`.

Key Value Description

Key	Description	
x-api-key	mvBoJnOvLY2q52GpecDShJ5dxfIV5rGM	200 OK • 431 ms • 1.47 KB

Body Cookies Headers (14) Test Results ⚙️

{ } JSON ▾ Preview Visualize

```
1  2025-05-02T21:04:11.603Z{  
2      "@odata.context": "$metadata#po(deliveryDocument,deliveryDocumentItem,material,headerGrossWeight,purchasingGroup)",  
3      "value": [  
4          {  
5              "deliveryDocument": "s4~1800000000",  
6              "deliveryDocumentItem": "s4-0000010",  
7              "material": "TG0011",  
8              "headerGrossWeight": 10,  
9              "purchasingGroup": "002"  
10         },  
11         {  
12             "deliveryDocument": "s4~1800000001",  
13             "deliveryDocumentItem": "s4-0000011",  
14             "material": "TG0011",  
15             "headerGrossWeight": 10,  
16             "purchasingGroup": "002"  
17         }  
18     ]  
19 }
```

# Business Benefits of Response Caching in SAP APIM

## GAINS

**Faster Response Times** → Improves user experience, especially for mobile users

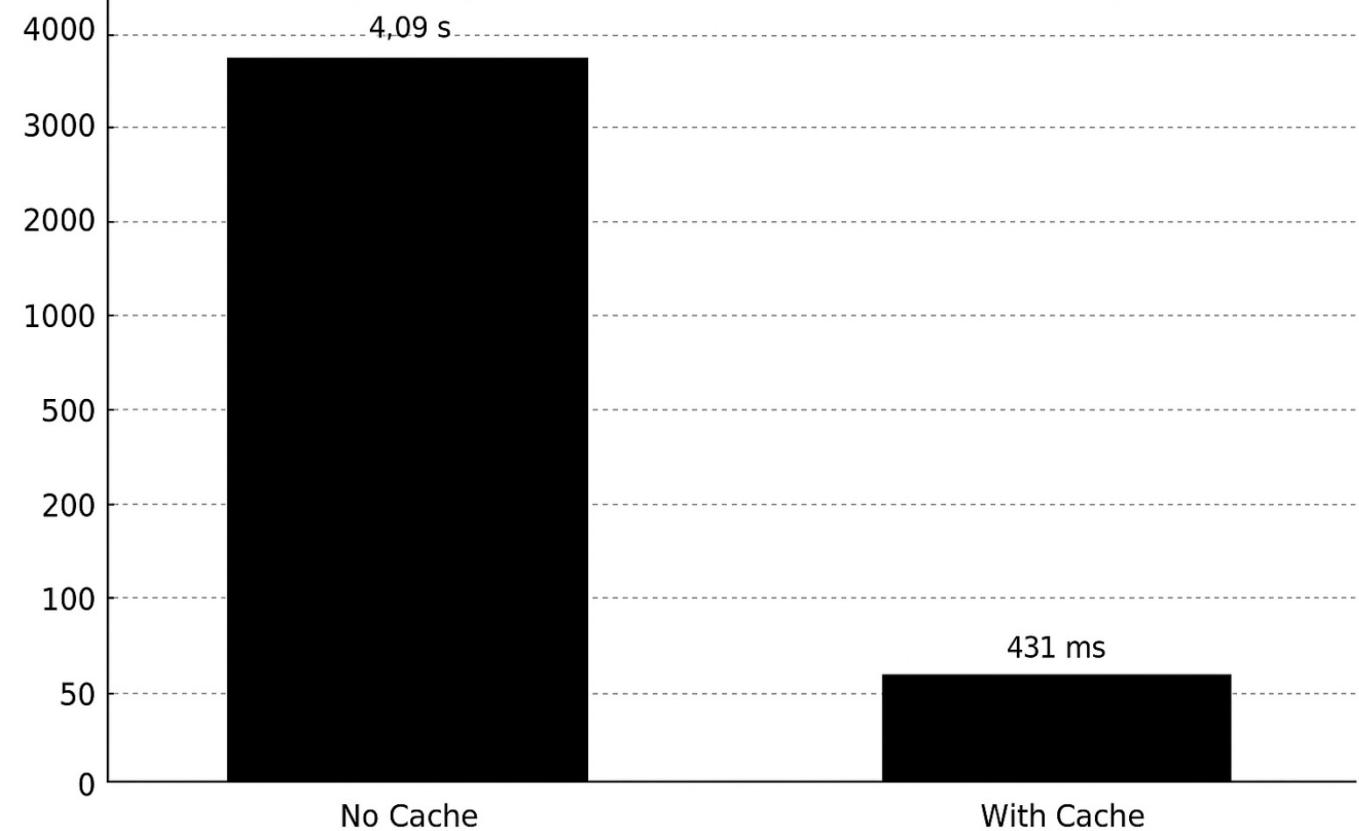
**Lower Backend Load** → Reduces pressure on SAP Graph or backend systems

**Cost Savings** → Avoids redundant backend calls to pay-per-use services

**API Quota Efficiency** → Prevents hitting rate limits from repeated queries

**Improved Stability** → Enables continued responses even during backend outages

Average Response Time Before and After Caching



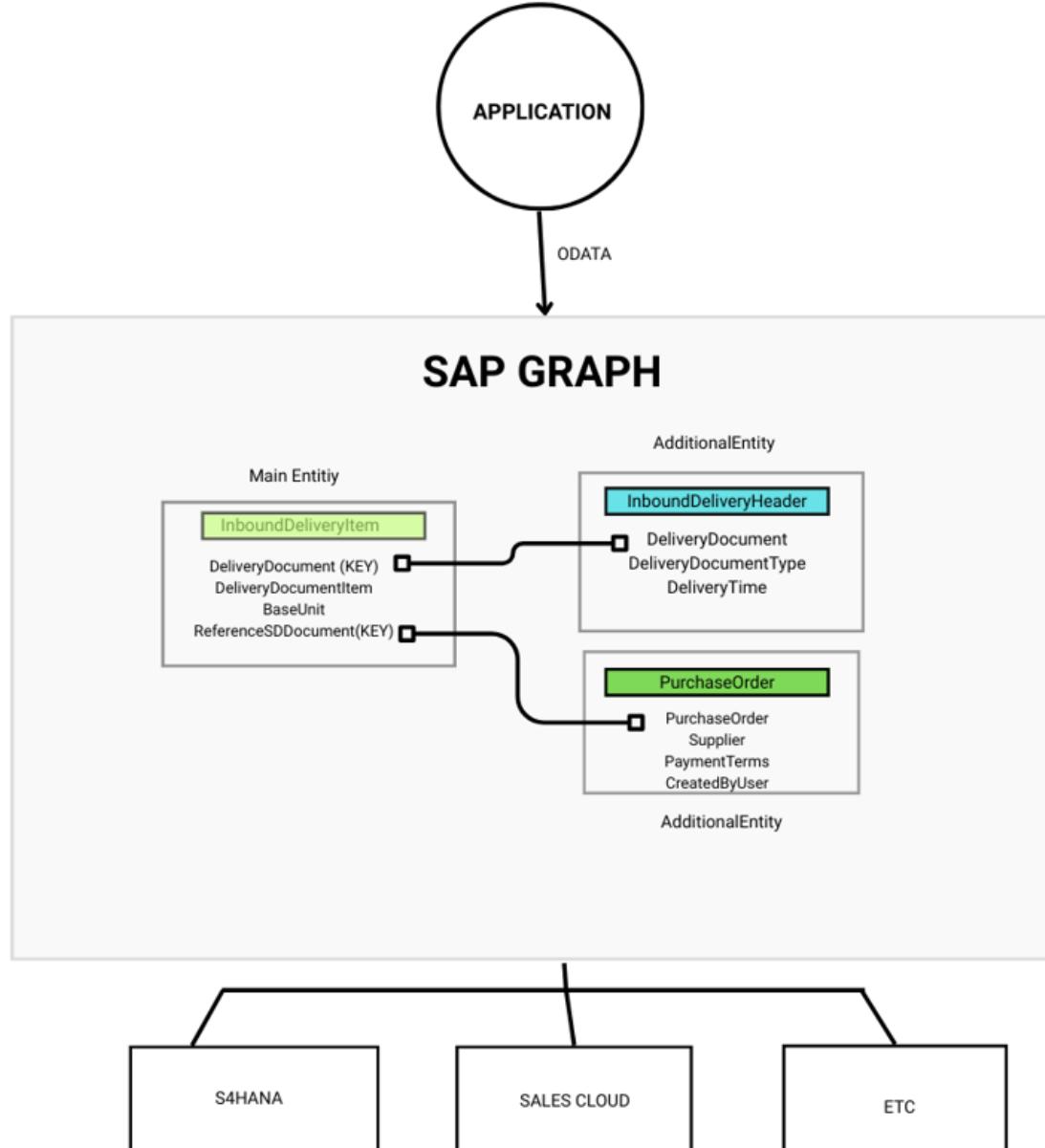
# SAP Graph

**”Single Access Point to Multiple SAP Systems»**

# What is SAP Graph?

- **SAP Graph** is a service offered by SAP that consolidates data from various SAP systems into a single unified API layer.  
It allows access to data from different systems such as **SAP S/4HANA**, **SAP SuccessFactors**, **SAP Sales Cloud**, and **SAP Concur** through one unified API.  
While doing so, it hides the diversity and complexity of the backend systems.  
The end user (or the integration developer) can access the data through a single **global data model**.

# SAP Graph Architecture: Unified Access Across SAP Systems



# SAP Graph : Hands-on

Design (Graph) / test Business Data Graph

Overview Schema Data Sources Model Extensions Cues Key Mappings

Model extensions are copied to the business data graph during activation. Changes to model extensions will only become visible when reactivating the business data graph. Learn more

**Model Extensions (1)**

Name	Description
ID_PO	

Design (Graph) / test Business Data Graph

Overview Schema Data Sources Model Extensions Cues Key Mappings

**Status**  
Available  
Last Modified: 1 minutes ago  
[Show Logs](#) [Reactivate](#)

**Description**

**URL**  
Access data & metadata  
OData: .../api/test  
GraphQL: .../api/test/graphql  
Catalog: .../catalog/test

**Data Sources**  
Connectivity to your underlying business systems  
s4

**Schema**  
Schema of your business data graph  
sap.s4  
id.po  
[Show in Developer Hub](#)

**Model Extensions**  
Customize your data model  
ID\_PO

# SAP Graph : Testing with Postman

The screenshot shows the Postman application interface with the following details:

- Method:** GET
- URL:** `https://graph-sap-test.svc.cluster.local:443/graph/api/scenario1/id/po?$filter=material eq 'TG0011'`
- Params:** Query Params table:
  - Key
  - \$filter

Key	Value	Description	...	Bulk Edit
\$filter	material eq 'TG0011'			
- Body:** JSON response (200 OK):

```
1 {  
2   "@odata.context": "$metadata#po",  
3   "value": [  
4     {  
5       "deliveryDocument": "s4~180000000",  
6       "deliveryDocumentItem": "s4~000010",  
7       "costCenter": "",  
8       "distributionChannel": "",  
9       "division": "",  
10      "material": "TG0011",  
11      "deliveryBlockReason": "",  
12      "deliveryDate": "2017-01-30T00:00:00.000Z",  
13      "deliveryDocumentType": "EL",  
14      "deliveryTime": "22:30:00",  
15      "headerGrossWeight": 10,  
16      "headerNetWeight": 9,  
17      "orderID": "",  
18      "id": "4500001435",  
19      "companyCode": "1710",  
20      "purchaseOrderType": "NB",  
21      "purchasingProcessingStatus": "02",  
22      "purchasingGroup": "002",  
23      "validityStartDate": null,  
24      "validityEndDate": null,  
25      "supplierPhoneNumber": "",  
26      "supplyingSupplier": "",  
27      "sunolvingPlant": ""  
}
```
- Headers:** (14) including `Content-Type: application/json`, `Accept: application/json`, etc.
- Metrics:** 200 OK, 3.31 s, 1.09 KB
- Buttons:** Send, Save Response, Copy, etc.

# AI in Integration Suite

# AI in Integration Suite

The screenshot illustrates the SAP Integration Suite interface, specifically the 'Replicate Business Partner to SAP S4HANA' scenario. A modal window titled 'Create Integration Flow' is open, prompting the user to choose a method: 'Create Integrations with assistance from AI' or 'Create Integration manually'. The 'Discover' section of the sidebar is currently selected. Another modal window, 'Generate Integration Flow', is also visible, detailing the scenario description and settings. The main workspace shows a list of artifacts, including several draft packages.

**Create Integration Flow**

Choose the preferred method to create your integrations

**Create Integrations with assistance from AI**

Let AI generate your integration flow based on the description of your integration scenario

**Create Integration manually**

Create an art traditional way of every step of the scenario

**SAP Integration Suite**

**Generate Integration Flow**

Describe your integration scenario: \* ⓘ

Replicate business partner from S4HANA-cloud to S4 Hana Object Plant every Monday and Wednesday at 9:00 Am Berlin time

882 characters remaining

Example: I would like to replicate newly hired employee data from SAP S/4HANA to SAP SuccessFactors to run the onboarding process.

AI response:

Thank you for providing the requirement. We will proceed with the integration of replicating business partner from S4HANA-cloud to S4 Hana Object Plant every Monday and Wednesday at 9:00 AM Berlin time.

Sender system: \*

Business Partner (A2X)

Receiver system: \*

Plant Substitution Exclusion

Name: \*

Replicate Business Partner from S4HANAcloud to S4 Hana

**Header**

Save Export Cancel Delete Package

Header Overview Artifacts Documents Content

Artifacts

Version Actions

Draft  >

1.0.0  >

Draft  >

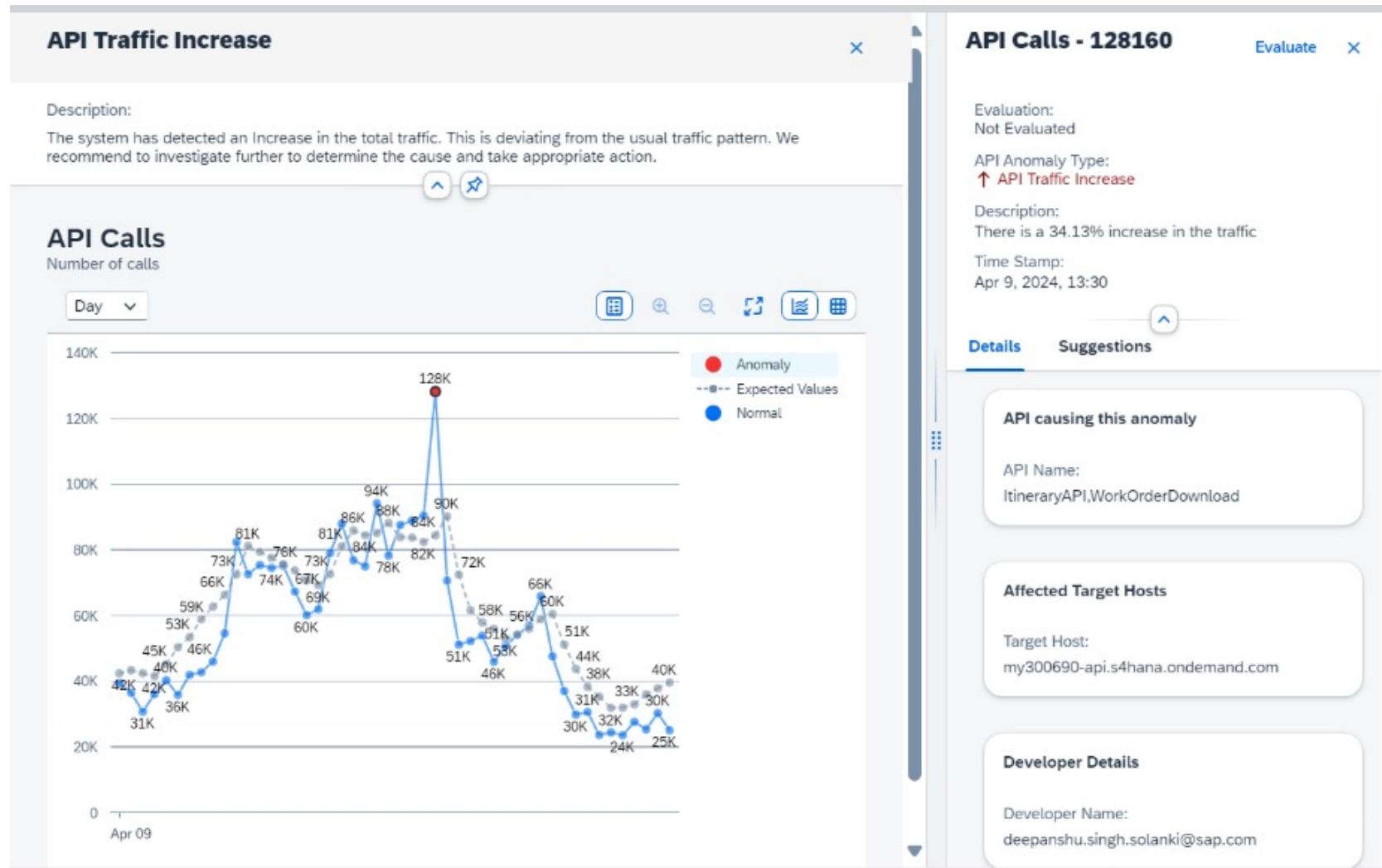
Draft  >

Draft  >

Draft  >

Save Export Cancel ...

# AI in Integration Suite



# AI in Integration Suite



... / log.groovy /

**log.groovy**

```
1 import com.sap.gateway.ip.core.customdev.util.Message;
2 import com.sap.it.op.agent.mpl.*;
3 import groovy.transform.Field;
4 import groovy.xml.XmlUtil;
5 import groovy.json.JsonOutput;
6
7 @Field String S4_MESSAGE_XML = '1 - Inbound Message XML';
8 @Field String IFLOW_MESSAGE_XML = '2 - Outbound Message XML';
9 @Field String IFLOW_MESSAGE_JSON = '3 - Outbound Message JSON';
10 @Field String RESPONSE_JSON = '4 - OSTA Response JSON';
11 @Field String EX_DETAILS = '4 - Exception Details';
12
13 def Message logS4MessageXML(Message message) {
14     return processData(S4_MESSAGE_XML, true, message);
15 }
16
17 def Message logIflowMessageJson(Message message) {
18     return processData(IFLOW_MESSAGE_JSON, false, message);
19 }
20
21 def Message logIflowMessageXML(Message message) {
22     return processData(IFLOW_MESSAGE_XML, true, message);
23 }
```

Apply   **Optimize**   Close

# AI in Integration Suite

SAP Integration Suite

Message Implementation Guidelines / **orders.orders05**  
Version: 4.0 (Draft)

Save Simulate Get Proposals Cancel ...

Overview Structure Notes (0) Namespaces MIG Codelists (0) Runtime Context Status

**Structure**

Node	Constraint	Cardinality	Position	Primitive ...	Sy...	Length	Codelist	Status
ORDERS05 – Purchasing/Sale:		1..1						
EDI_DC40 – EDI DC40		1..1						
E1EDK01 – E1 EDK01		1..1						
E1EDK14 – E1 EDK14 - Divi	006	0..1						
E1EDK14 – E1 EDK14 - Cor	011	0..1						
E1EDK03 – E1 EDK03 - Rec	002	0..1						
E1EDK04 – E1 EDK04		0..10						
E1EDK05 – E1 EDK05		0..16						

A high-angle, aerial photograph of a bustling city street, likely in London, showing heavy traffic and pedestrian activity.

Event-Driven Architecture “in action”  
with  
**SAP Advanced Event Mesh**

# A lot of Event-Driven Use Cases

HR across multiple systems	Data Lake Updates	Medication Delivery and Tracking	Flight Tracking	Production Line Analysis	Demand Based Supply Chain	Attendance Tracking	Financial Market Data Distribution	Global Event Localization and Distribution	Inventory Management	Event-Driven Forecasting
Weather based Pricing	Water Quality Monitoring	Harbour Management	Elevator Emergency Service	Real Time Inventory	Sales Support	Budgeting and Forecasting	Smart Order Routing	Meter to Cash	Ticketing	Document Update Notification
Supply Chain Control Tower	Digital Twin	Supply Chain Visibility	Customer Guidance	Shipment Notification and Tracking	Credit Card Authorizations	Automatic Ordering	Food Quality Tracking	Sales Order Shock Absorber	Oil Well Monitoring	Mobile Workforce Connectivity
Industrial Maintenance	Real Time Trading	Real Time Loyalty Points	Luggage Tracking	Buffering when Offline	Smart Financial Order Routing	Across Vendor Business Processes	Regulatory Reporting	Online Store Buffering	Smart City	Event Log / Kafka Data Updates
Smart Meter Readings	Predictive Maintenance	Real Time Order Management	Connected Car	Real Time Test Data	Across Vendor Integration	Airport Management	Real Time Payments	Point of Sale	Smart Trashcans	Business Process Documentation

# Smart Routing – multi source / multi target

## TOPIC

[object] / [action] / [source] / [objectID] / [timestamp] / [type]

### Events published

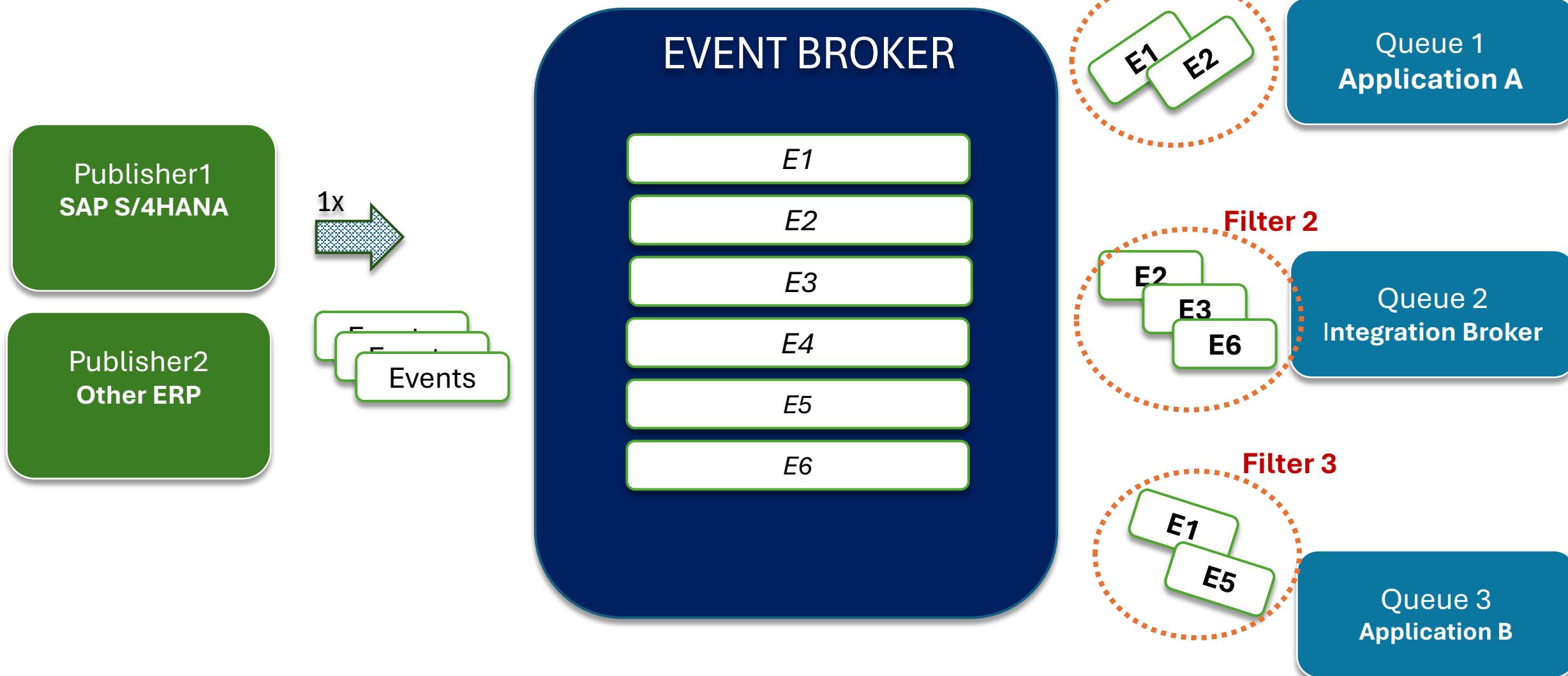
EVENT1	customer	/	created	/	s4p500	/	998844	/	20250510131522	/	XYZ
EVENT2	customer	/	created	/	erp200	/	999423	/	20250510131522	/	ABC
EVENT3	customer	/	changed	/	s4p500	/	998844	/	20250510134322	/	XYZ

### Route Events via Filters

FILTER 1	customer	*		s4p500	*		*		*		*
FILTER 2	customer		created		*		*		*		XYZ
FILTER 3	*		*	*		998844		*		*	

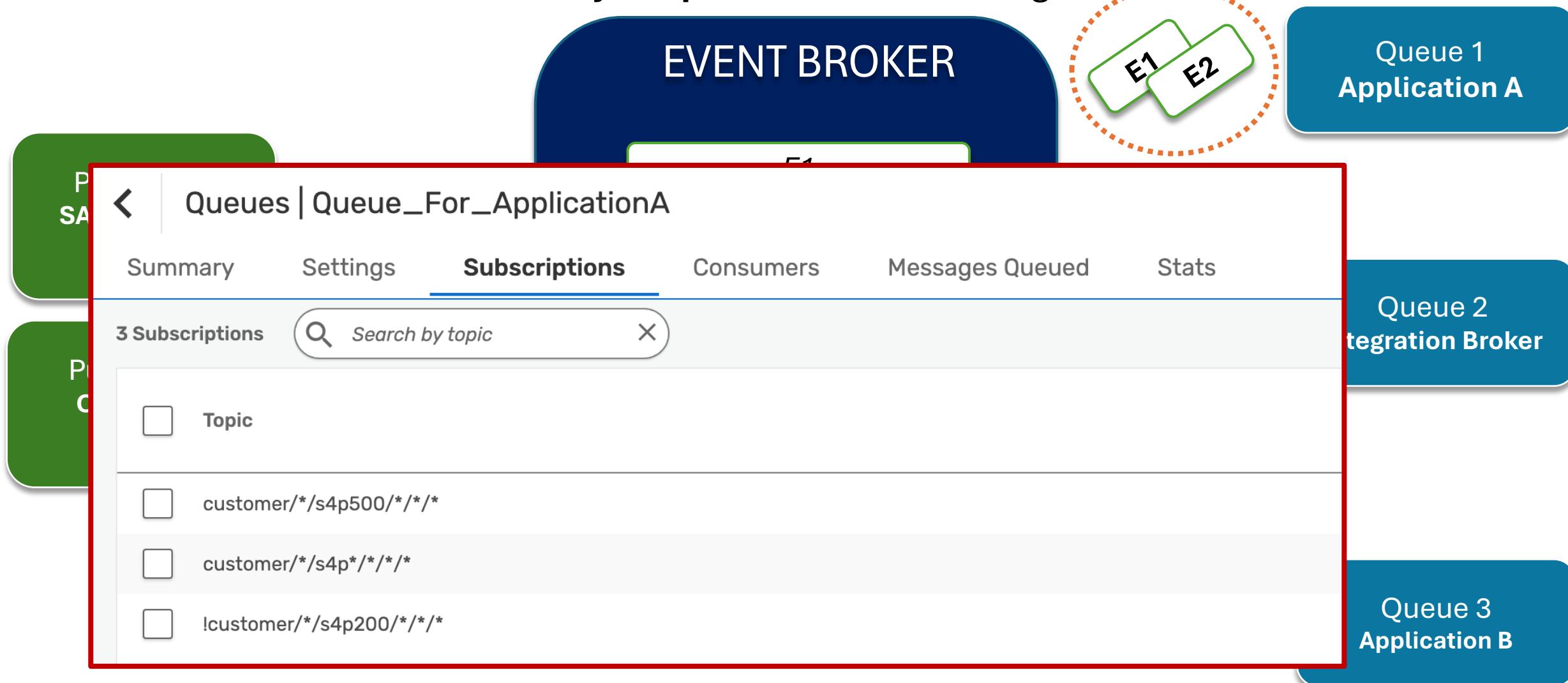
# Smart Routing – multi source / multi target

Millions of Events ..MANY “loosely coupled” sources and targets

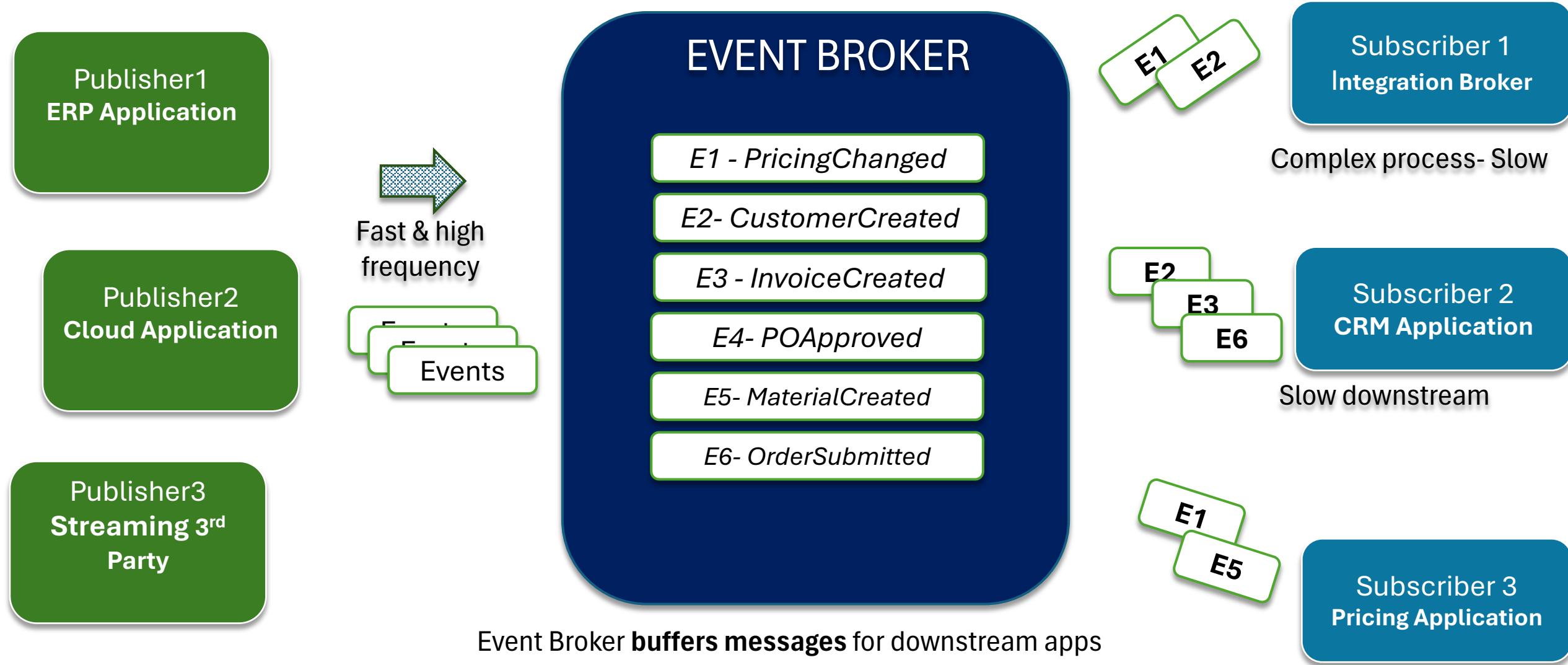


# Smart Routing – multi source / multi target

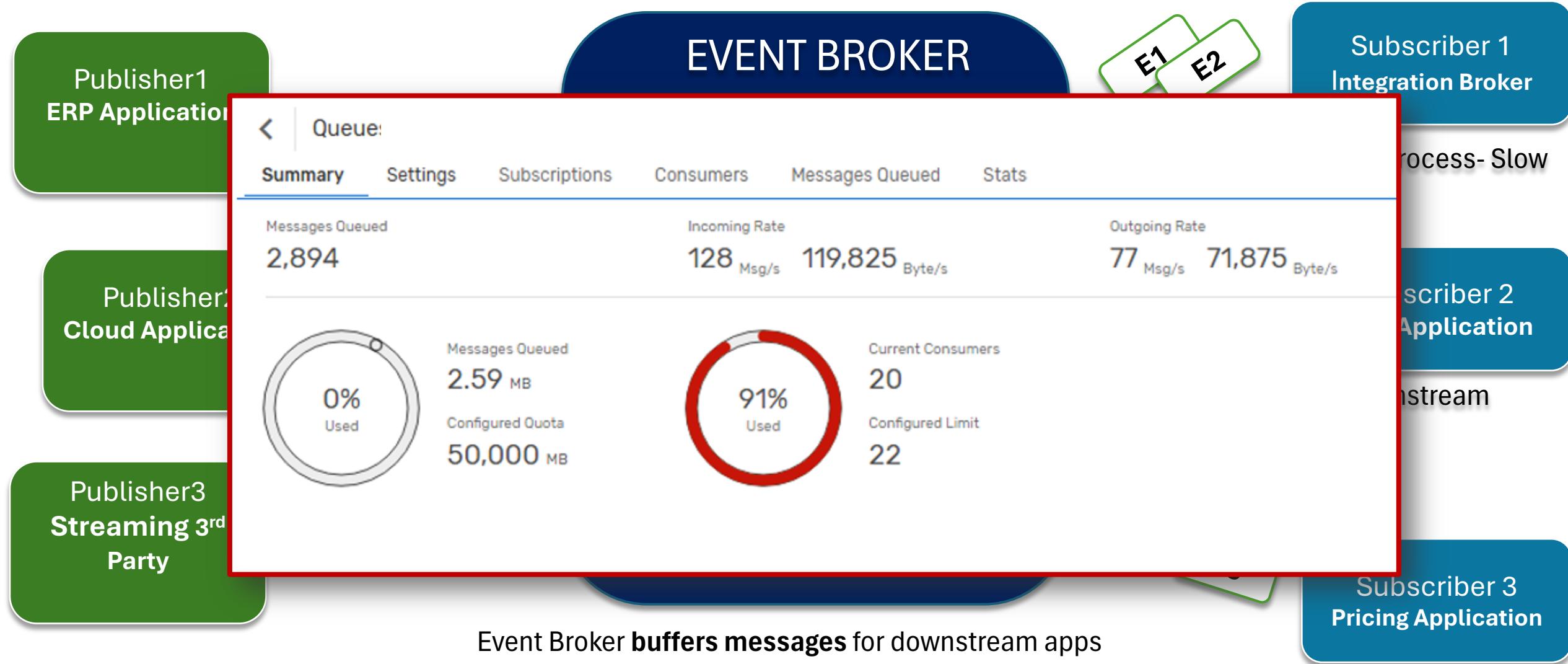
Millions of Events ..MANY “loosely coupled” sources and targets



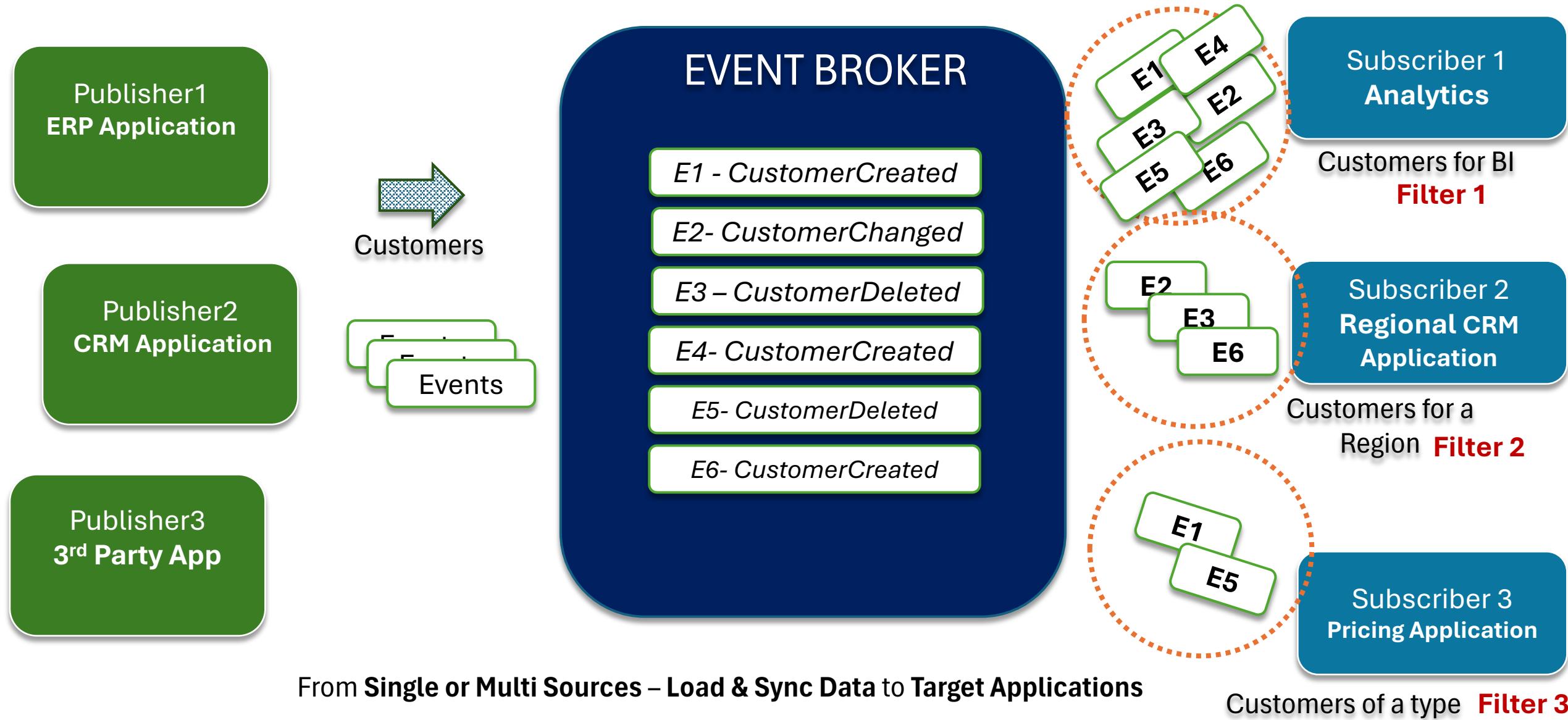
# Managing back-pressure – Relaxing Consumers



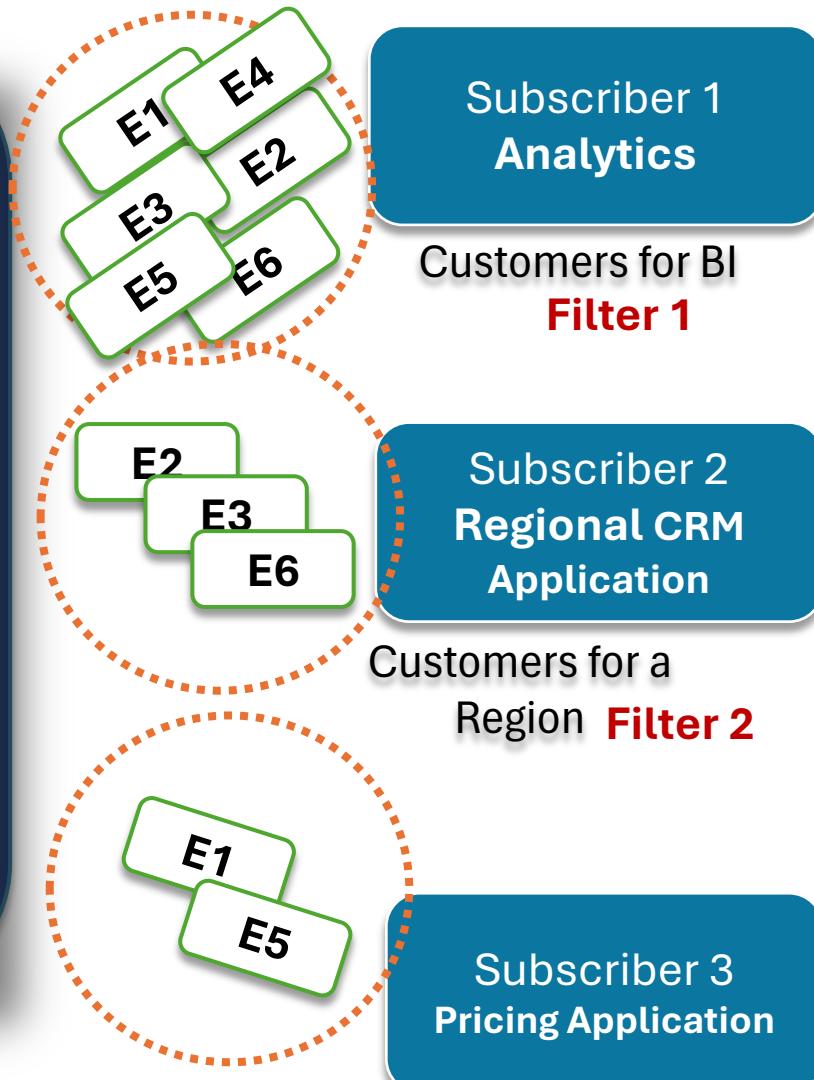
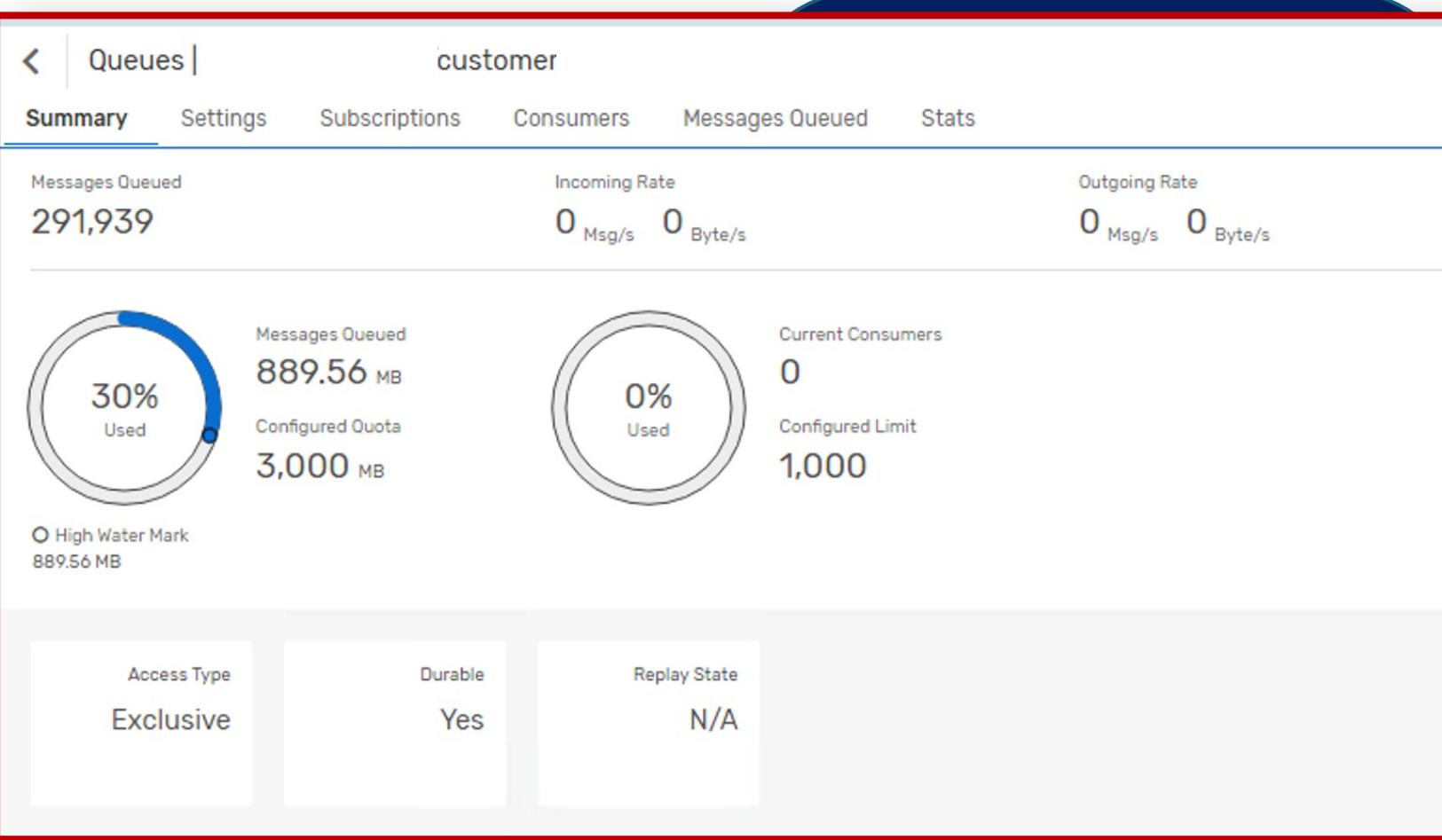
# Managing back-pressure – Relaxing Consumers



# Initial Data Load – and sync near-real time



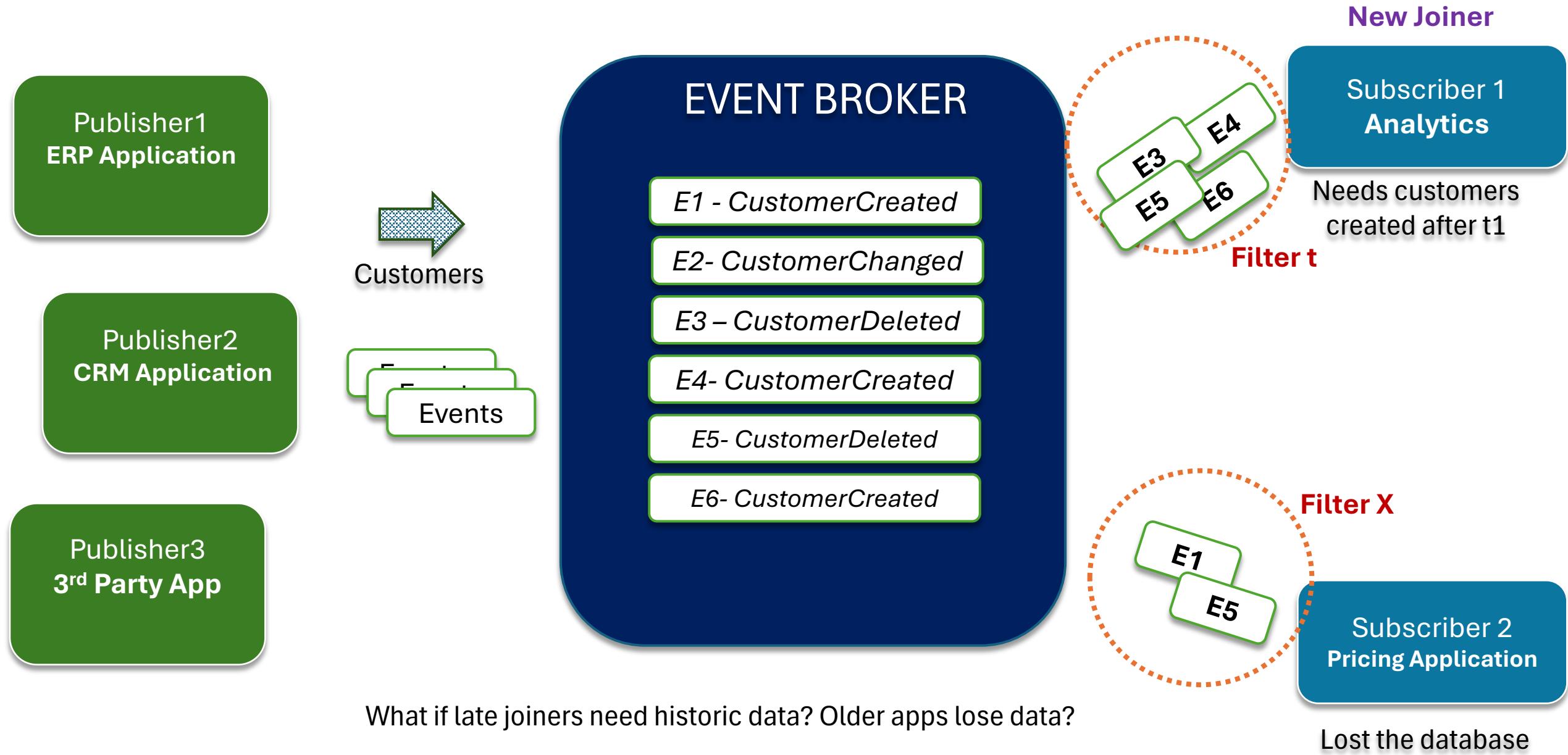
# Initial Data Load – and sync near-real time



From Single or Multi Sources – Load & Sync Data to Target Applications

Customers of a type **Filter 3**

# Replay– what if some target apps need historical data?



# Replay – what if some target apps need historical data?

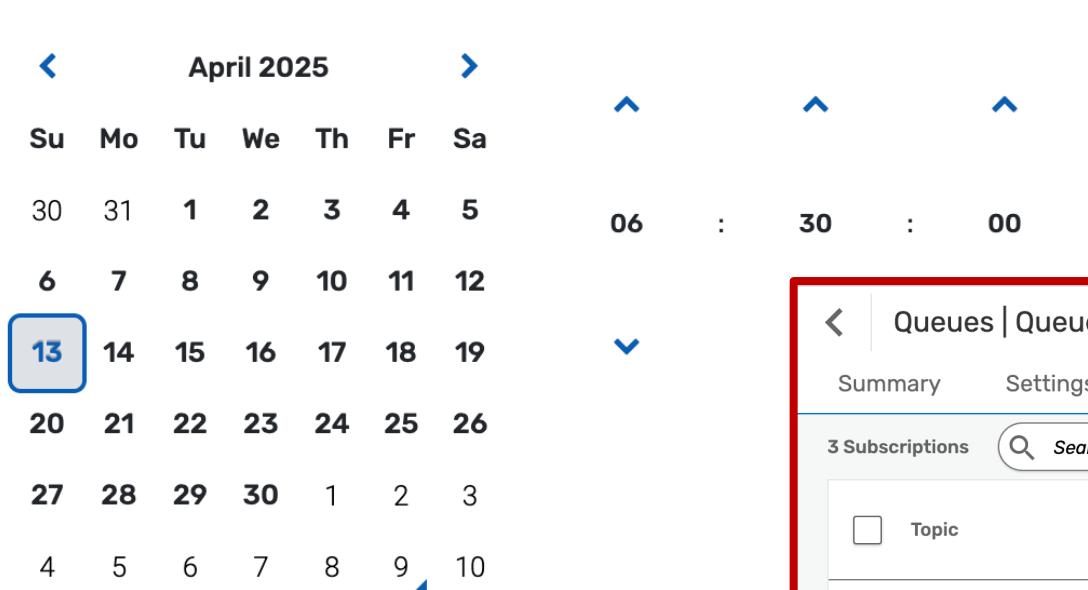
ER

3r

**Start Replay**

Start Replay from Beginning  
Start to replay messages from the oldest message

Start Replay from Date (in Local Time)  
2025-04-13 06:30:00



**New Joiner**

**Subscriber 1 Analytics**  
Needs customers created after t1

**Filter t**

**Subscriber 2 Pricing Application**  
Lost the database

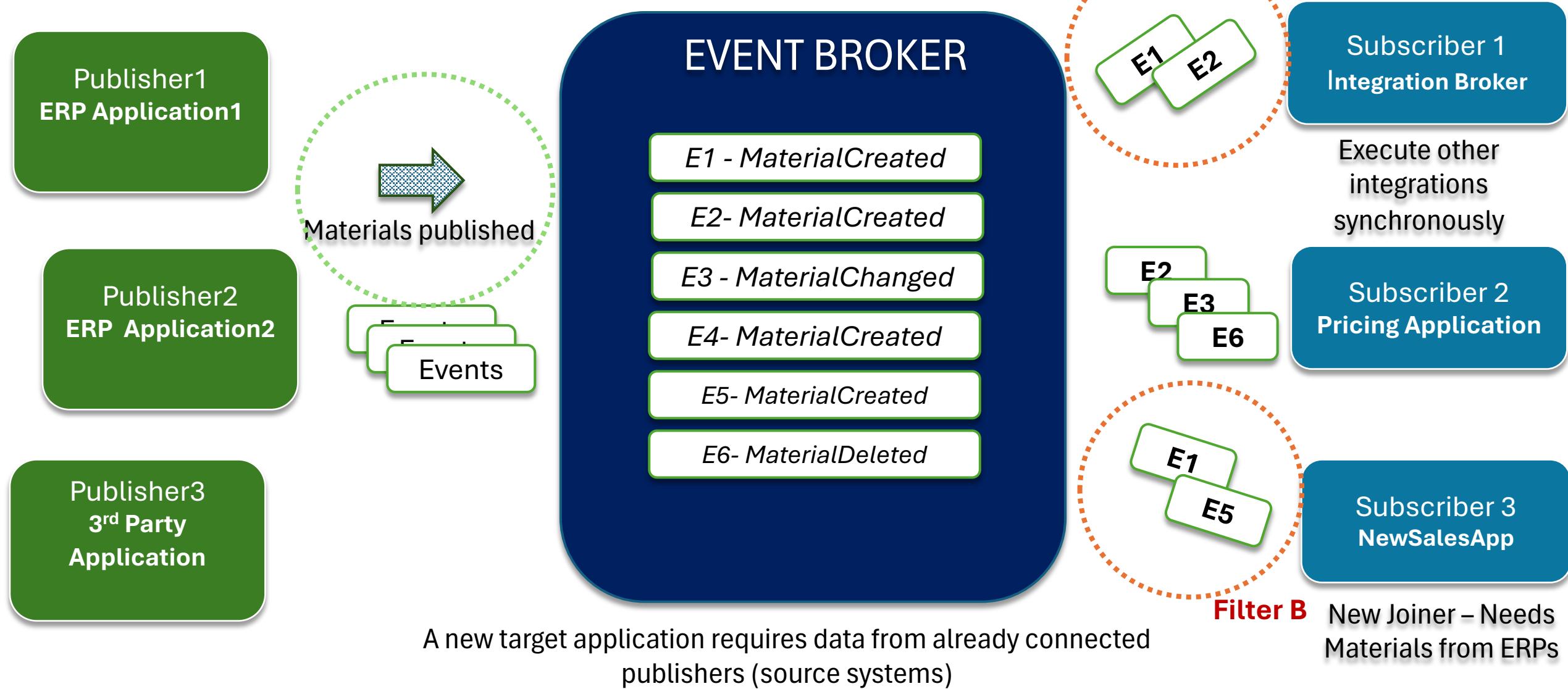
**Filter X**

**Queues | Queue\_For\_ApplicationA**

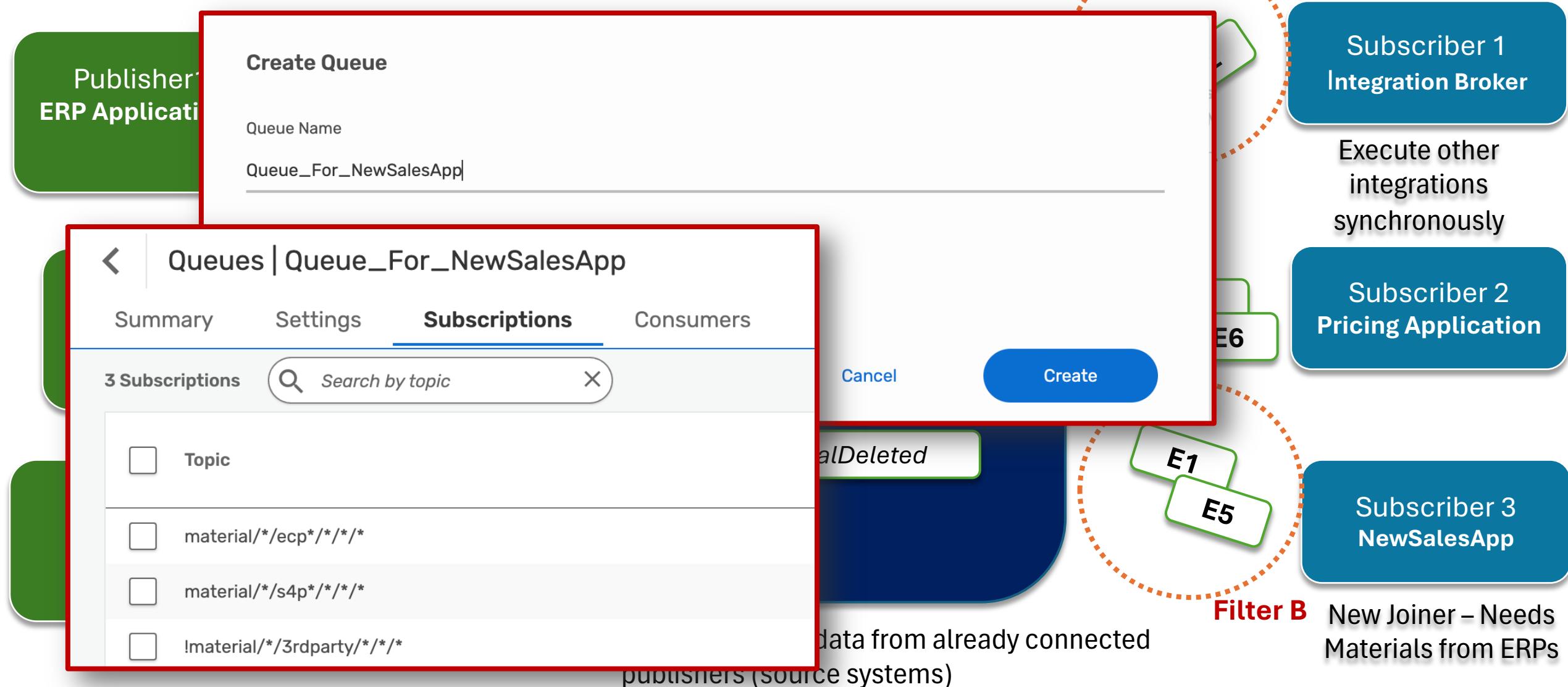
Subscriptions 

Topic
customer/*/s4p500/*/*
customer/*/s4p*/*/*
!customer/*/s4p200/*/*

# Plug&Play - How easy to engage the next app?



# Plug&Play - How easy to engage the next app?





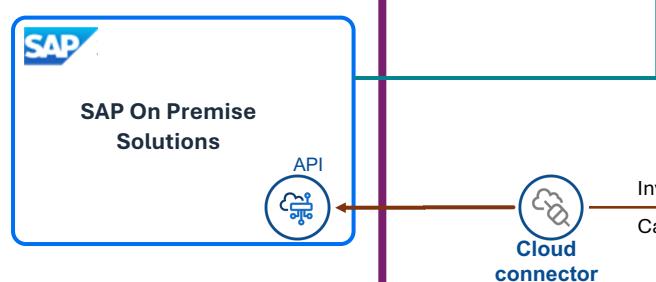
# Recap & Next..

**”Integration & EDA Roadmap and Take Aways»**

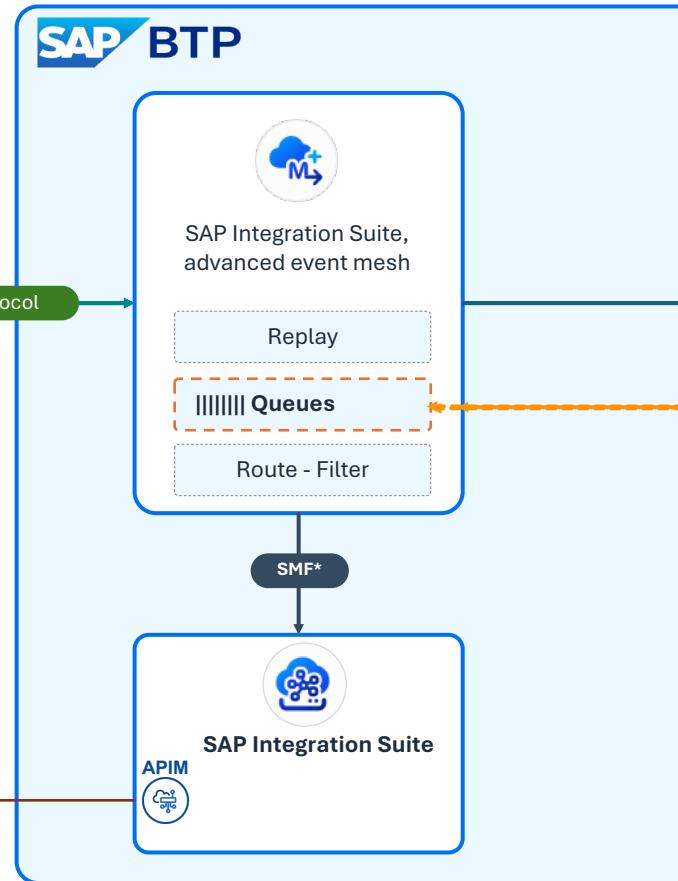
# The Big Picture – Architecture

Upstream(publishers)

1 Publish data as events

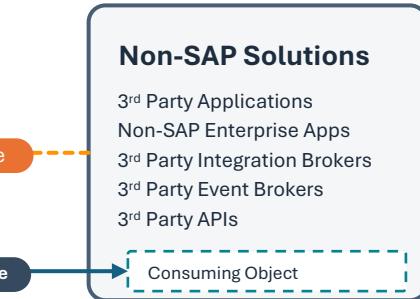


Brokers

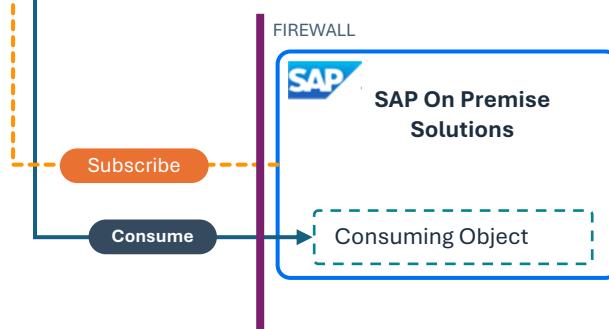


Downstream(subscribers)

3 Subscribe



4 Consume Actively



# The Roadmap – BTP Integration & Advanced Event Mesh

**FUTURE RELEASE**

**Integrating SAP Cloud Application Programming Model and advanced event mesh**

Providing support for SAP Integration Suite, advanced event mesh in SAP Cloud Application Programming Model

SAP Integration Suite, advanced event mesh

**Open** ^

**FUTURE RELEASE**

**Integration of advanced event mesh for health monitoring in SAP Cloud ALM and SAP Focused Run**

Use SAP Cloud ALM, our central entry point for monitoring SAP landscapes, for monitoring the health of SAP Integration Suite, advanced event mesh

SAP Integration Suite, advanced event mesh

**Open** ^

**FUTURE RELEASE**

**Distributed tracing enablement between SAP S/4HANA and advanced event mesh**

Enable the distributed tracing feature of SAP Integration Suite, advanced event mesh for SAP S/4HANA as an event source

SAP Integration Suite, advanced event mesh

**Open** ^

**FUTURE RELEASE**

**Scenario monitoring in SAP Cloud ALM and SAP Focused Run**

Use SAP Cloud ALM, our central entry point for monitoring SAP landscapes, for monitoring integration scenarios through SAP Integration Suite, advanced event mesh

SAP Integration Suite, advanced event mesh

**Open** ^

**FUTURE RELEASE**

**AI adapter**

Provide an AI adapter as an additional built-in connectivity option for the Cloud Integration capability within SAP Integration Suite to connect to large language models

SAP Integration Suite

**Open** ^

**FUTURE RELEASE**

**"Where-used" functionality for security materials**

- Enable "where-used" functionality in SAP Integration Suite for identifying the integration artifacts in credentials, key pairs, and certificates
- Availability with standard and premium editions of SAP Integration Suite and with CPEA

Cloud Integration SAP Integration Suite

**Open** ^

**FUTURE RELEASE**

**AI-based payload size anomaly identification in API traffic**

Detect unusual request payload sizes and flag them as anomalies

SAP Integration Suite

**Open** ^

**FUTURE RELEASE**

**XML threat protection policy for XML-based APIs**

XML threat protection policy in the API Management capability to help mitigate various security threats, such as XML bombs attacks, by validating, filtering, and restricting XML payloads before they reach backend services

SAP Integration Suite Edge Integration Cell

**Open** ^

# Final Words - Key Takeaways

- Design for the long-term benefits (consider maintenance, reusability)
- Modernize your existing interfaces whenever possible (replace inefficient, old-technology)
- Think End-to-End (for all current & possible future sources and targets)
- Start smart / Start small



Thank you..

Barış BÜYÜKTANIR

[baris@blackbelt.solutions](mailto:baris@blackbelt.solutions)

 <https://www.linkedin.com/in/barisbt/>

 @barisbt

Deniz ZİLYAS

[kilitoglu.deniz@gmail.com](mailto:kilitoglu.deniz@gmail.com)

 <https://www.linkedin.com/in/deniz-zilyas/>