

January 2023

Cloud Patterns with SAP Environments

Presented by Ethan Jewett

MINDSET



Ethan Jewett

Sr. Director of DevOps,
Development, and Technology

A photograph of a person's hands writing on a notepad at a wooden table. Another person's hands are visible in the background, gesturing. The image has a blue tint and a dark overlay.

MINDSET

Experience the Human-Centered Enterprise



| Our Mission

Make the world of work a more positive, engaging, and Human-centered place.

A black and white photograph of a calendar and a smartphone on a wooden surface. The calendar shows days of the week with some dates marked with 'X'. A smartphone is placed over the calendar. The background is a dark blue gradient.

Today's Agenda

- What we'll cover today
- SAP and ERP overview
- Patterns and why we use them
- SAP/ERP + Cloud Patterns
- Wrap up and resources



Topics for today

Motivating question

How do we build great enterprise experiences on challenging underlying systems and infrastructure?



| Key Takeaways

- Learn some of the basic of ERP and SAP environments, which are often insulated from the rest of the enterprise.
- Understand the value of SAP systems and tools, and the challenges of integrating them into a distributed cloud environment.
- Learn techniques and terminology for working with SAP-related groups in your organization to merge SAP (and other) applications into your enterprise cloud strategy.
- Find out about resources available to support this integration.



SAP & ERP

What is SAP?



SAP SE or SAP AG
(company)



an SAP ERP
(system)



other products
(cloud / platform)

| SAP in Numbers

~75%

Of all business
transactions worldwide
touch
an SAP system

~85%

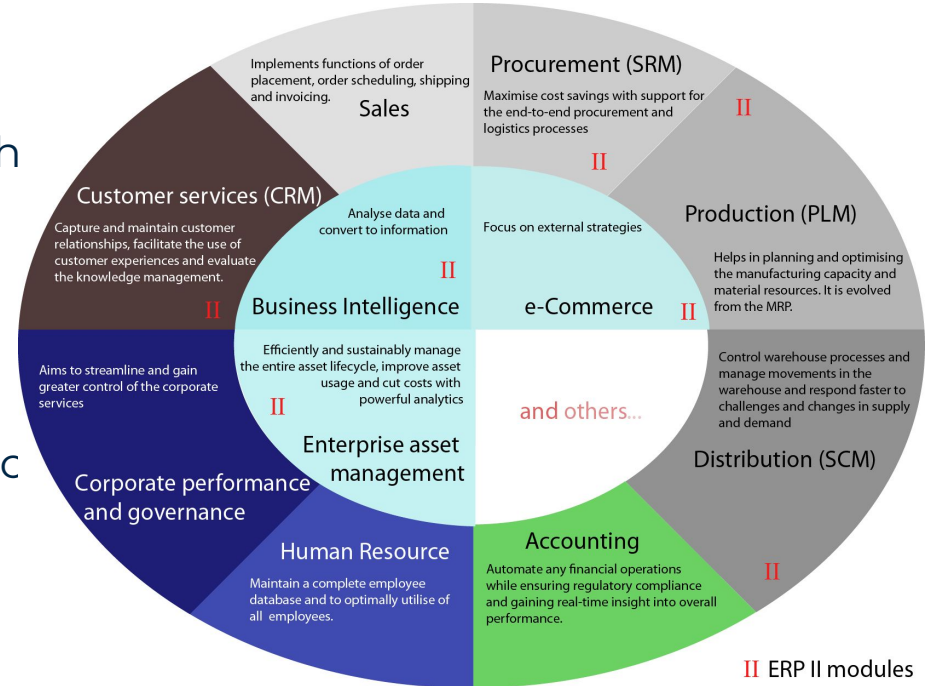
Of Forbes Global 2000
companies

99 of 100

largest companies in the
world are SAP customers

Unique aspects of SAP & ERP systems

- Not “QuickBooks on steroids”
- Complex, but can be integrated with
- Proprietary lifecycle management
- Proprietary languages
- Insular culture & terminology
- Deep and valuable process best prac ecosystems



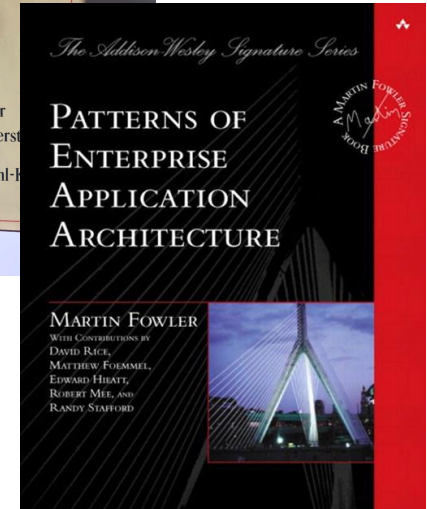
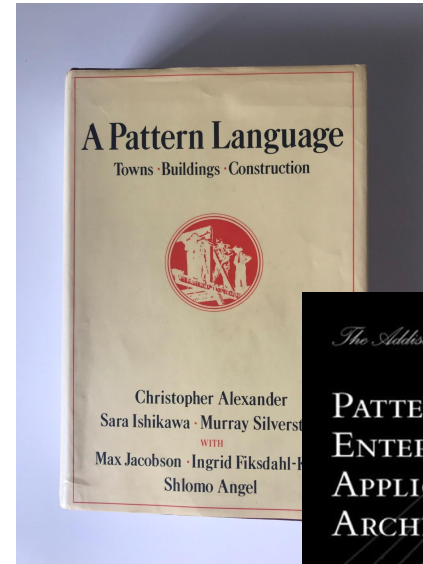
Shing Hin Yeung, CC BY-SA 3.0 <<https://creativecommons.org/licenses/by-sa/3.0/>>, via Wikimedia Commons



Why patterns

Patterns

- Since the late '70s, pattern languages were introduced as a standard way of communicating complex concepts
- A shorthand and simplification that allows for working with and communicating about enterprise system design



| Our goals for patterns

Look for patterns to **mitigate the weaknesses of ERP and SAP systems while preserving their strengths**, to enable the creation of human-centered enterprise experiences that help us unlock our human capital.



SAP/ERP patterns

| Pattern 1: On-premises → Cloud infrastructure

Current state

- ERPs tend to be relatively monolithic applications
- Often hosted in local data centers
- More often currently in managed data centers or on hyper-scaler infrastructure
- Usually, but not always, in virtualized infrastructure

| Pattern 1: On-premises → Cloud infrastructure

Challenges

- Time and cost of change can be challenging due to manual processes
 - E.g. standing up new application server involves ordering a server
 - Databases tend to scale vertically, so needing to resize a database server can break the bank
- Geographic proximity to cloud services can be sub-optimal
- Networking changes can be equally challenging, precluding integration scenarios

| Pattern 1: On-premises → Cloud infrastructure

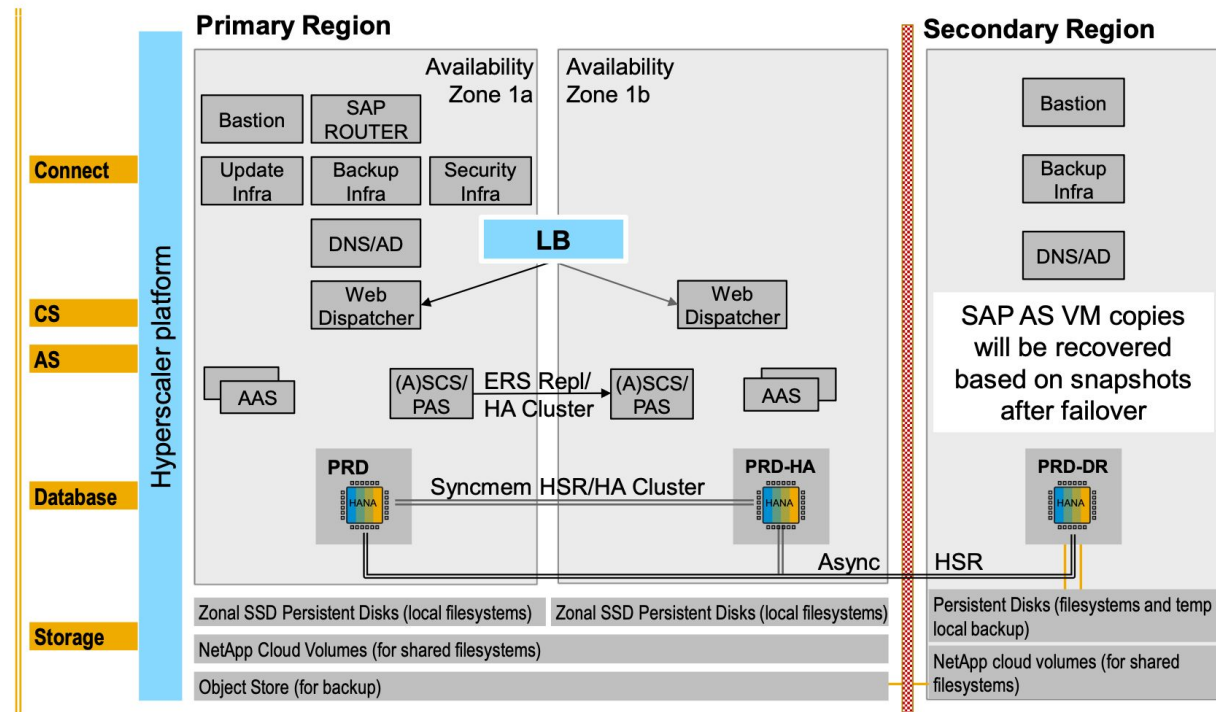
Approach

- Largely replicate on-premises setup in cloud
- Design hardware and network architecture in alignment with hyper-scaler architecture
- Abide by reference architectures provided by ERP vendors and hyper-scalers

Pattern 1: On-premises → Cloud infrastructure

Architecture on a Hyperscaler

High-Availability/Disaster Recovery architecture



An example
SAP
high-availability
-disaster
-recovery on
hyper-scaler
architecture

Pattern 2: ERP as set of modern, cloud services

Current state

- ERP resides in the cloud, but doesn't act cloud-native
 - APIs are limited
 - APIs that do exist are very heterogeneous, mostly synchronous, badly documented, and locking is a problem
- Native UI/UX **can** (in SAP's case) be made fast and with relatively good user experience, but it takes significant work and thought

The challenge: The rest of the enterprise rightly perceives ERP systems as difficult to use, either as a user or as a developer of another system.

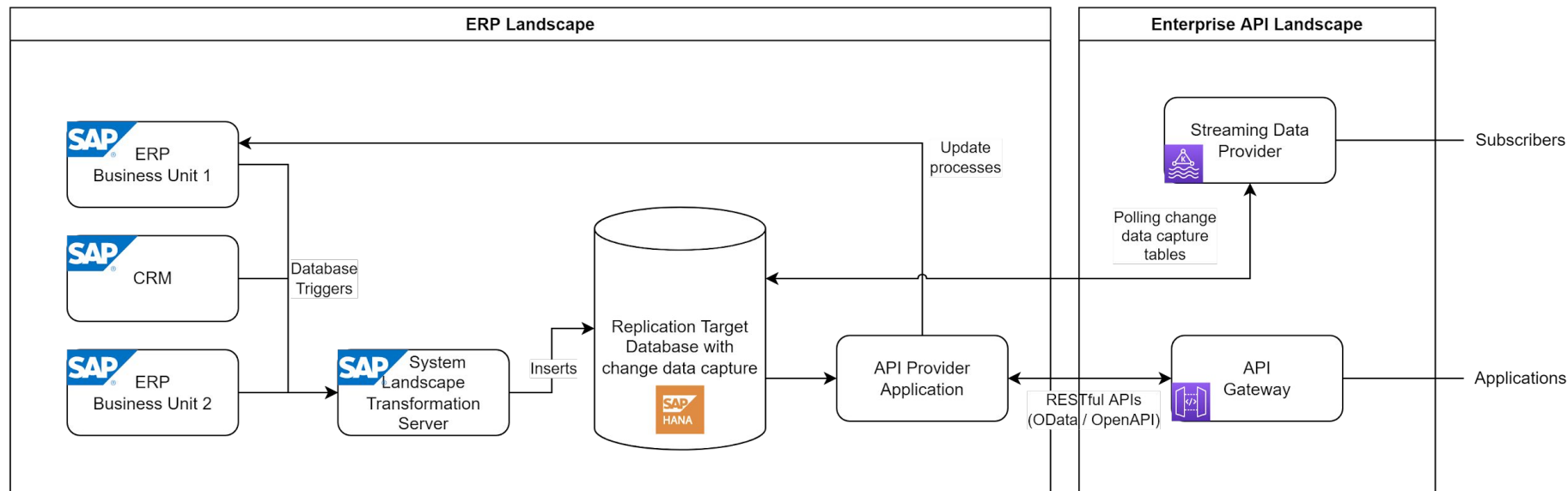
| Pattern 2: ERP as set of modern, cloud services

Approach:

- API-enable the ERP system through 2 mediums:
 - RESTful microservices
 - Queues
- Incorporate these APIs into the enterprise API strategy
 - API architecture patterns
 - Documentation

Pattern 2: ERP as set of modern, cloud services

A common API-enablement pattern.



Pattern 3: Enhancements in the cloud

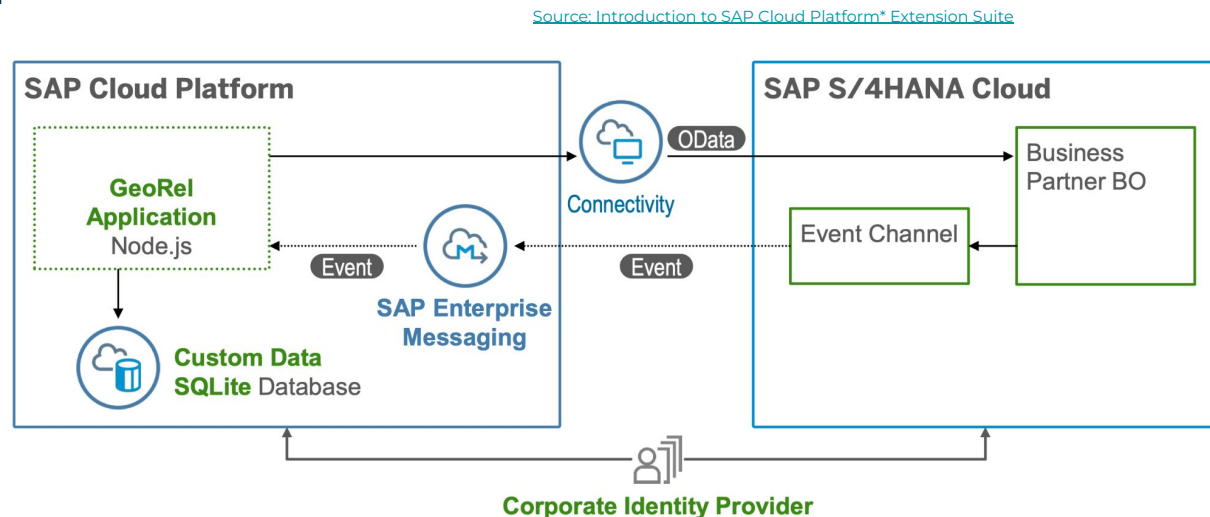
Current state:

- ERP systems tend to have customizations or enhancements put in place by customers
 - These enhancements can be difficult to manage and impede upgrade paths
- SAP and other vendors provide enhancement frameworks, but they locked to SAP's proprietary cloud environment (BTP)

Pattern 3: Enhancements in the cloud

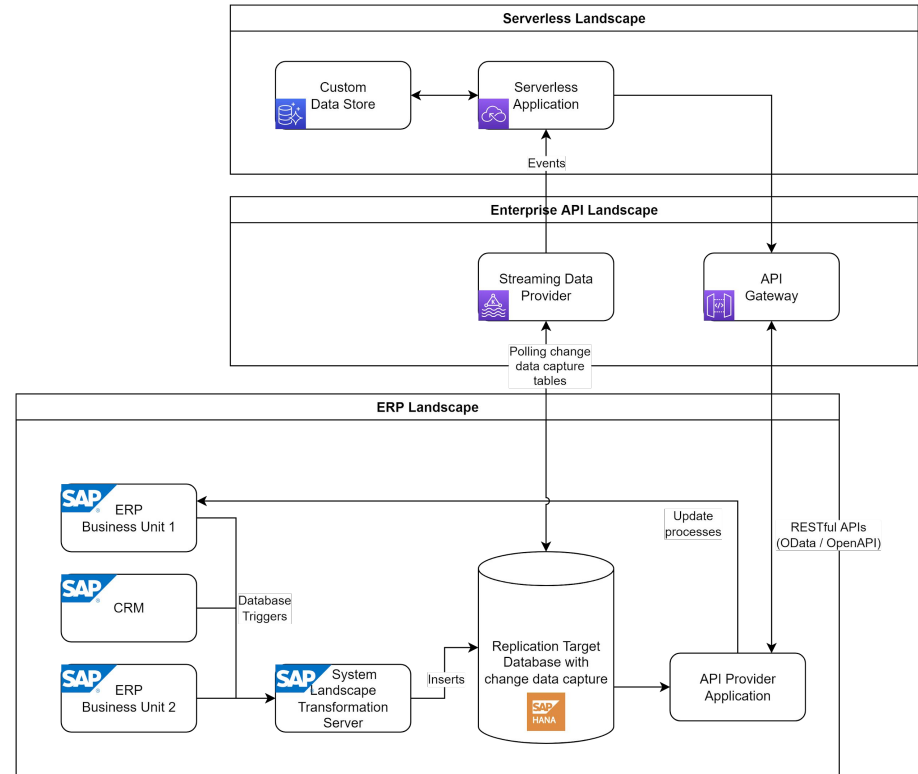
An SAP reference architecture for cloud extension.

- Built-in event model only available on modern S/4HANA
- Only supported entities
- Use of SAP technologies is OK, but we need options



Pattern 3: Enhancements in the cloud

A general pattern for cloud enhancement follows naturally from our cloud services pattern



| Pattern 4: Principal propagation patterns

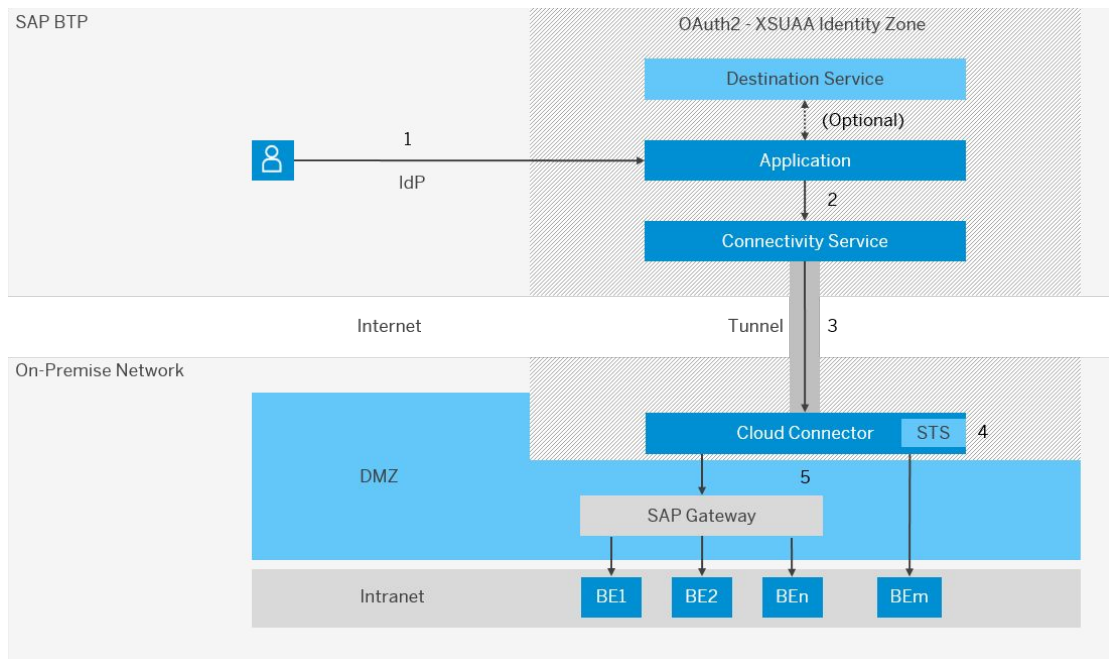
Current state

- ERP system largely rely on trusted system approaches rather
- Modern system design uses zero-trust architectures
- SAP provides reference architectures for BTP to On-premises and BTP to Hyper-scaler integration, but little for On-premises to Hyper-scaler

Pattern 4: Principal propagation patterns

Approach

- Use SAP's reference architectures when they work
- Derive a more general pattern
- Here, SAP BTP cloud to on-premises

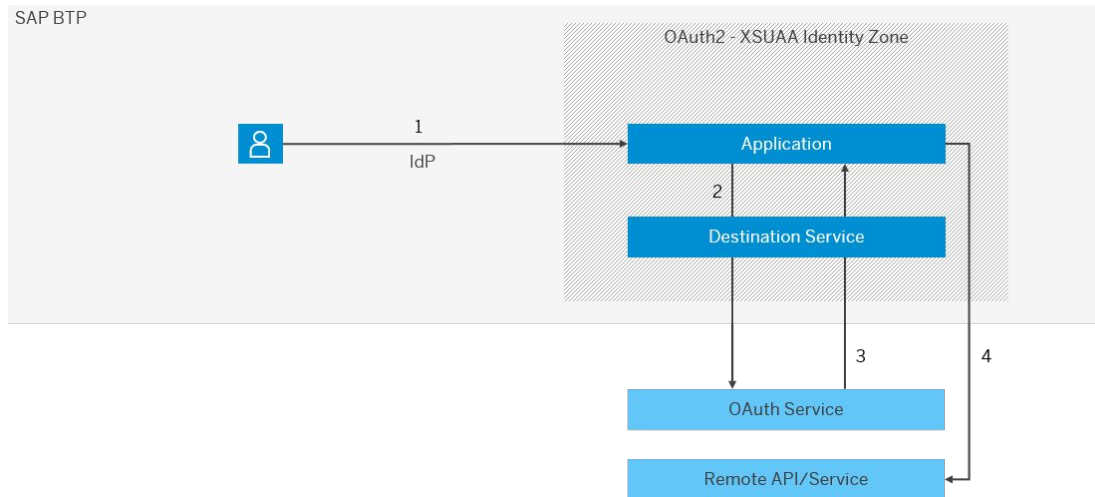


[Principal Propagation \(help.sap.com\)](https://help.sap.com)

Pattern 4: Principal propagation patterns

SAP BTP Cloud to generic cloud provider

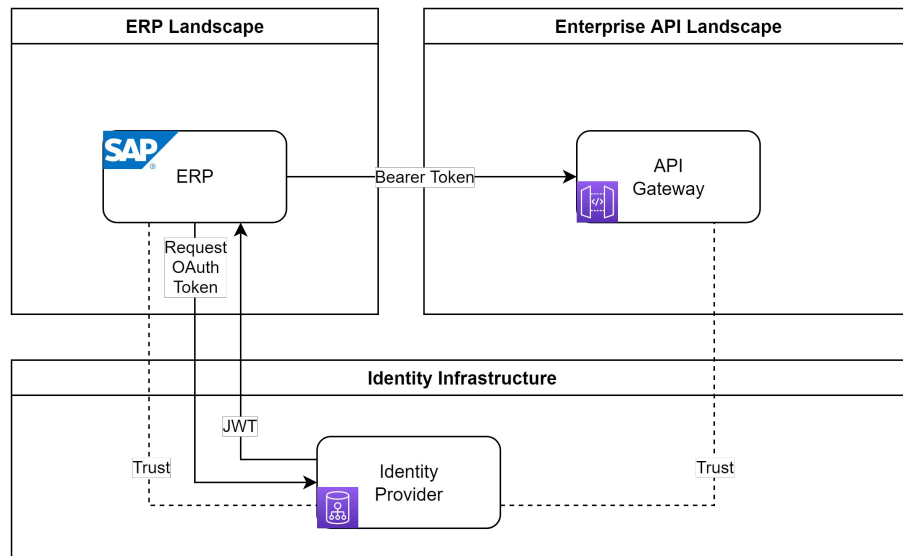
- Limited detail in this architecture pattern
- Several different patterns deeper in the documentation



[Principal Propagation \(help.sap.com\)](https://help.sap.com)

Pattern 4: Principal propagation patterns

- A generic principal propagation pattern
- Multiple OAuth flows are supported
 - **SAML Assertion**
 - Use for principal propagation without user interaction
 - **Auth Code**
 - Principal propagation with user interaction
 - **Client credentials**
 - Does not propagate identity
 - Only supported in modern S/4HANA platforms
- OAuth flows are patterns!
 - [Azure IdP reference](#) is usually a great pattern reference manual





Wrap-up

| Key Takeaways

- Learn some of the basic of ERP and SAP environments, which are often insulated from the rest of the enterprise.
- Understand the value of SAP systems and tools, and the challenges of integrating them into a distributed cloud environment.
- Learn techniques and terminology for working with SAP-related groups in your organization to merge SAP (and other) applications into your enterprise cloud strategy.
- Find out about resources available to support this integration.



Thank You!

Ethan Jewett

Sr. Director of DevOps, Development, and Technology

Presentation:

<https://www.mindsetconsulting.com/cloud-patterns-with-sap-environments/>

Email: ethanjewett@mindsetconsulting.com

Twitter: @esjewett

Mastodon: @esjewett@micro.esjewett.com

LinkedIn: <https://www.linkedin.com/in/ethanjewett/>



Reference

Reference

Patterns:

- [Enterprise Patterns \(Martin Fowler\)](#)
- [Catalog of Patterns of Enterprise Application Architecture](#)
- [A Pattern Language for Microservices](#)
- [Pattern Languages](#)

Pattern 1 - On-premise → Cloud Infrastructure

- [SAP on Hyperscalers – Strategy, Architecture and Deployment](#)

| Reference

- Pattern 2 - ERP as a set of modern, cloud services
 - Mindset Case Study - [Global provider of water, hygiene, and energy technologies](#)
 - Mindset Case Study - [Xcel Energy](#)
- Pattern 3 - Enhancements in the cloud
 - Mindset Case Study - [Utility Holding Company – Outage Project](#)
 - OpenSAP - [Introduction to SAP Cloud Platform* Extension Suite](#)

Reference

- Pattern 4 - Principal propagation patterns
 - help.sap.com - [Principal Propagation](#)
 - help.sap.com - [ABAP OAuth Client](#)
 - help.sap.com - [SAP BTP Destinations](#) (for BTP → Cloud principal propagation)
 - [OAuth 2.0 application type](#) (Azure)
 - [Token grant flows](#) (Azure)