



# Cloud Native & Serverless SaaS Extensions

---

Angelo Santagata

Architect A-Team  
Oracle HQ

Jürgen Kress

Director of Product Management  
Oracle HQ

## Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.



# Agenda

---

- A-Team
- Why use OCI to Extend SaaS?
- Collection of patterns and solutions
- Conclusions



## SaaS A-Team : Who are we ?

- Central, outbound, highly technical team comprised of Architects, Solution Specialists and Software Engineers within the Oracle SaaS Product Development Organization.
- Work closely with customers and partners, world wide, providing guidance on architecture, best practices, troubleshooting and how best to use Oracle Cloud Services and products to solve customer business needs.
- Team that specializes in cross product scenarios that span IaaS, PaaS and SaaS as well as hybrid deployments
- Operate at the points of intersection across our product lines and technologies
- Purview and focus follows Oracle's product strategy, involved in product engineering decision making
  - Focus on both supporting pre and post sale activities
- A-Team Blog: <http://www.ateam-oracle.com/>



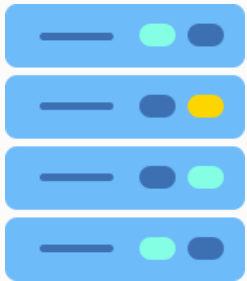
## Why use OCI to Extend SaaS?

---

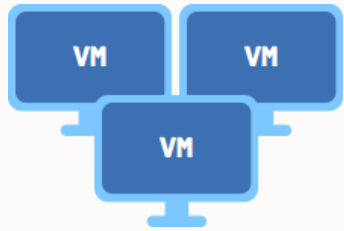
- Oracle is promoting OCI to our customers , this obviously includes our SaaS customers
- OCI provides services which range from Oracle managed to Customer managed
  - Most customers prefer managed infrastructure but we provide both
- IDCS is now federated with Fusion Apps and easily useable by OCI applications
- Many customers already have Cloud Native skills in house
  - Re-use their existing CI/CD pipelines, governance , monitoring , manageability frameworks etc
  - Use their existing skills to build extensions
  - Sometimes extensions are only needed for a short period of time.
    - OCI allows them the agility to spin up and destroy solutions very quickly
- OCI/Cloud Native skillset freely available in the market
- SaaS and OCI can be in the same, or closely located, datacenter
  - Fusion SaaS is rapidly moving to OCI datacenters



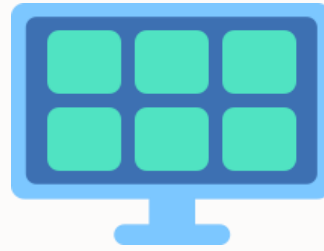
# Customer Managed vs Oracle Managed or a Mix of both



**Bare Metal**



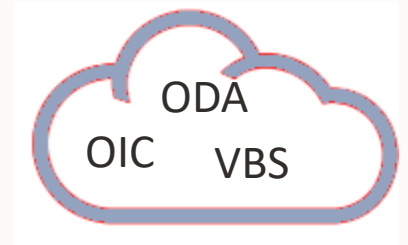
**Virtual machines**



**Containers**



**Functions**



**PaaS**

Code	Code	Code	Code	Declarative
App Container	App Container	App Container	App Container	App Container
Language Runtime	Language Runtime	Language Runtime	Language Runtime	Language Runtime
Operating System	Operating System	Operating System	Operating System	Operating System
Hardware	Hardware	Hardware	Hardware	Hardware





# What about OIC – Oracle Integration Cloud (OIC)?



Digital assistant



Blockchain

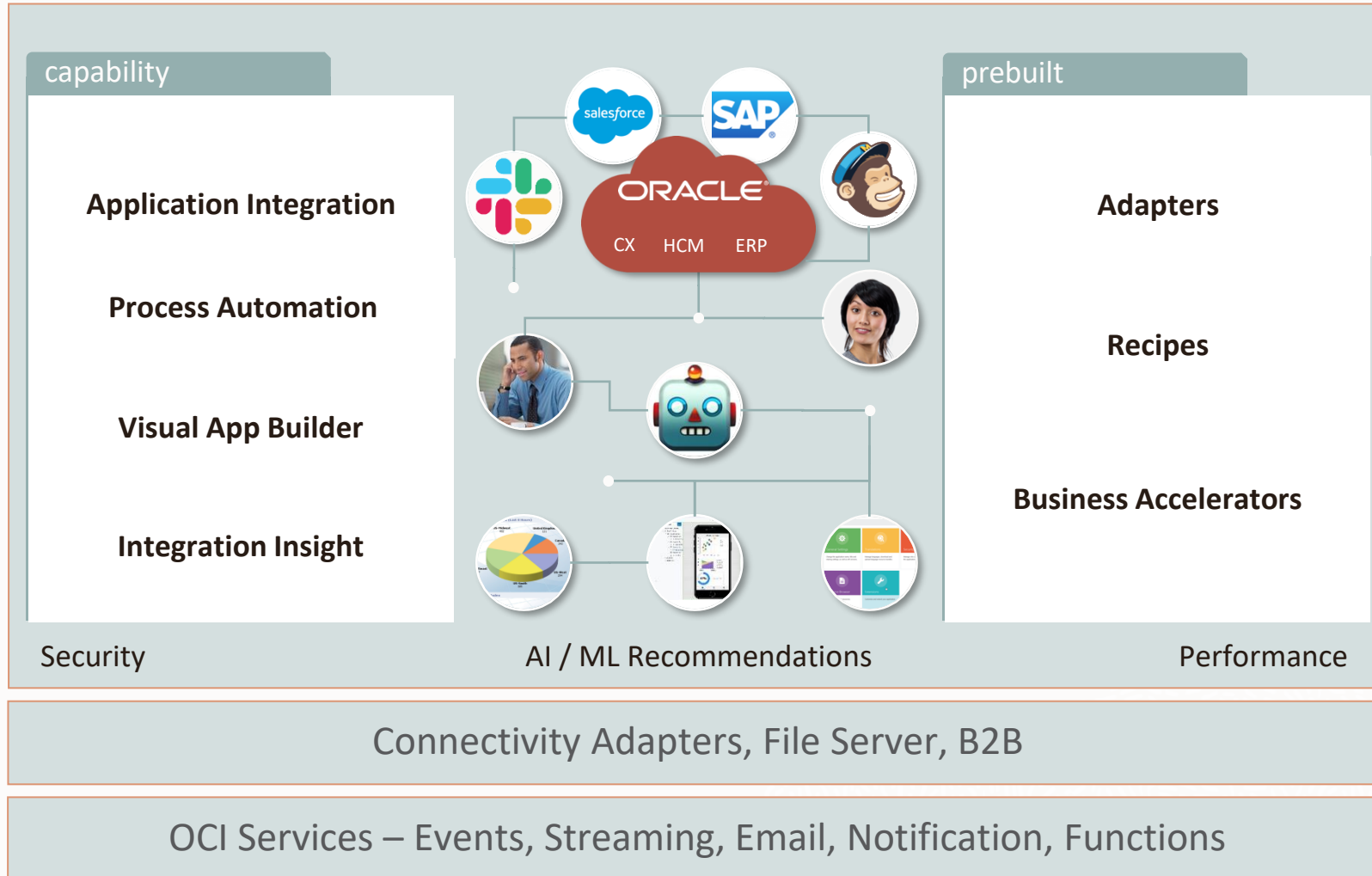
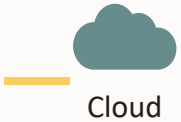


IoT



Augmented analytics

## Deployment flexibility



## API Management



Design

Govern

Manage

Monetize



# What is Cloud Native

**Cloud native** is a more general approach to building and running applications that take advantage of cloud computing

## CNCF Cloud Native Definition v1.0

*Approved by TOC: 2018-06-11*

[العربية](#) (Arabic) | [中文版本](#) (Chinese) | [日本語版](#) (Japanese) | [한국어](#) (Korean) | [Deutsch](#) (German) | [Español](#) (Spanish)  
[Français](#) (French) | [Polski](#) (Polish) | [Português Brasileiro](#) (Portuguese) | [Русский](#) (Russian) | [Bahasa Indonesia](#) (Indonesian)

Cloud native technologies empower organizations to build and run scalable applications in modern, dynamic environments such as public, private, and hybrid clouds. Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

These techniques enable loosely coupled systems that are resilient, manageable, and observable. Combined with robust automation, they allow engineers to make high-impact changes frequently and predictably with minimal toil.

The Cloud Native Computing Foundation seeks to drive adoption of this paradigm by fostering and sustaining an ecosystem of open source, vendor-neutral projects. We democratize state-of-the-art patterns to make these innovations accessible for everyone.

<https://github.com/cncf/toc/blob/master/DEFINITION.md>



# What is Serverless?

Serverless is of an execution model in the cloud

Article [Talk](#) [Read](#) [Edit](#) [View history](#)

## Serverless computing

From Wikipedia, the free encyclopedia

**Serverless computing** is a [cloud computing execution model](#) in which the cloud provider runs the [server](#), and [dynamically](#) manages the allocation of machine resources. Pricing is based on the [actual amount of resources consumed by an application, rather than on pre-purchased units of capacity.](#)<sup>[1]</sup> It can be a form of [utility computing](#). Serverless is a [misnomer](#) in the sense that servers are still used by cloud service providers to execute code for developers. The management and details of these servers are [transparent](#) to the application developers.

Serverless computing can simplify the process of [deploying code](#) into production. [Scaling, capacity planning and maintenance operations](#) [may be hidden](#) from the developer or operator. Serverless code can be used in conjunction with code deployed in traditional styles, such as [microservices](#). Alternatively, applications can be written to be purely serverless and use no provisioned servers at all.<sup>[2]</sup> This should not be confused with computing or networking models that do not require an actual server to function, such as [peer-to-peer \(P2P\)](#).



[https://en.wikipedia.org/wiki/Serverless\\_computing](https://en.wikipedia.org/wiki/Serverless_computing)



# Two Primary Perspectives When Extending SaaS



- **Integrating** with SaaS
  - Usually involves getting data in, or out, of Fusion SaaS
  - Consuming events coming out SaaS
  - This is the most popular perspective, everyone has integration!
- **Extending** SaaS functionality
  - Usually involves adding functionality to SaaS
    - e.g. additional or custom UI components / iframes / rules / extra fields
  - Custom standalone screens that interact with SaaS Services , direct or via a middle tier
  - Custom mobile applications
  - Extending Process



# Extending SaaS with OCI Use cases and Patterns

A collection of patterns we have seen





# Security is essential

Every pattern must be secure

- Security is just as important as ever
- All Fusion Applications instances are now automatically federated to IDCS
- Protect your APIs using API Gateway
  - Custom OCI Applications can leverage IDCS by using oAuth and IDCS “confidential” apps
  - Two Approaches
    - Quick and dirty : Add FA resources your IDCS oAuth confidential app
    - Proper and clean : Create your own scopes and then perform a token exchange with Fusion
- Any passwords used within any OCI app should be stored within the OCI Vault

Token Issuance Policy ⓘ

Authorized Resources  All  
 Tagged  
 Specific

Resources

[+ Add Scope](#)

Resource	Protected	Scope
Oracle Applications Cloud (Fusion)	No	<code>urn:opcresource:fa:instanceid=501840138urn:opc:resource:consumer::all</code>

Grant the client access to Identity Cloud Service Admin APIs

[+ Add](#)

App Roles	Protected
No data to display.	



# Integration Usecases

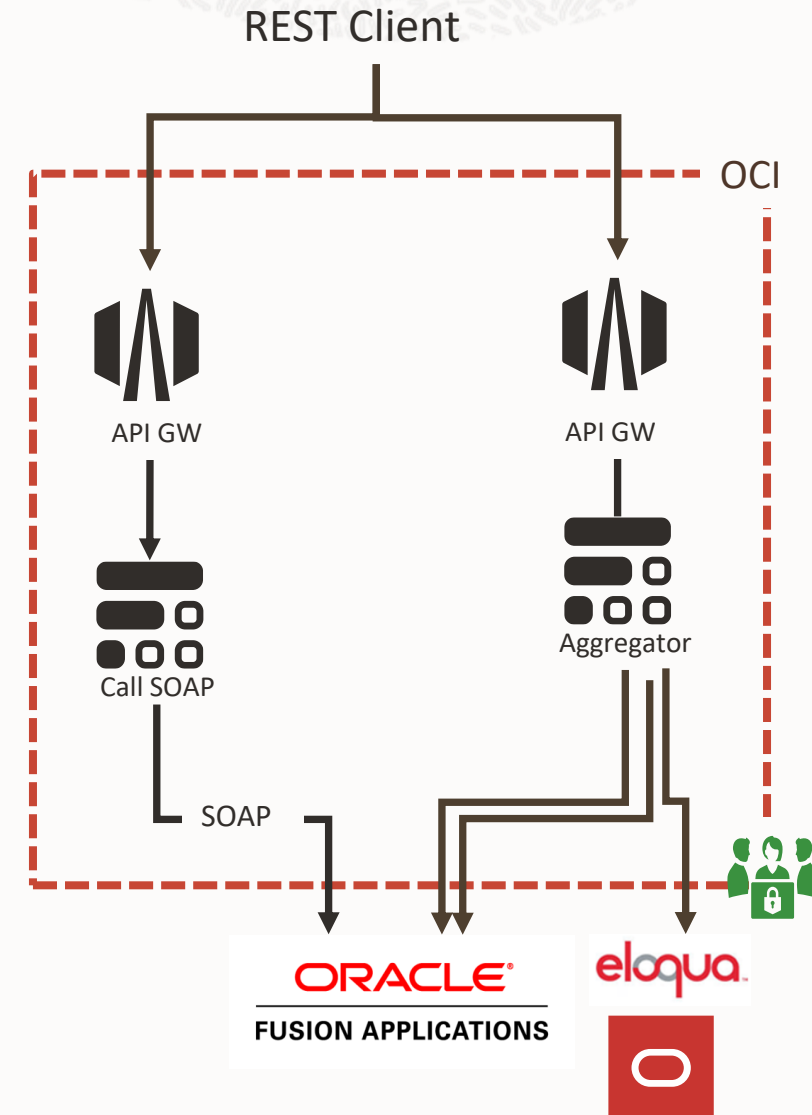
---



# Integrating with SaaS using OCI

## Simple OCI Serverless Integration Examples

- Use-Cases
  - Calling SOAP services from VB
  - Orchestration of FA Services which are SOAP/REST Based
  - Aggregating SOAP/REST Services
- Benefits of using Serverless
  - Cost, depending on the customer the consumption can fall within the Cloud Functions free tier
  - Manageability, it is serverless nothing to manage
  - Very microservice like
- Issues
  - Code based approach vs a declarative one
  - Agility, code changes need to be managed
- Alternatives
  - OKE, OIC , WLCS etc
- Sample code will be made available

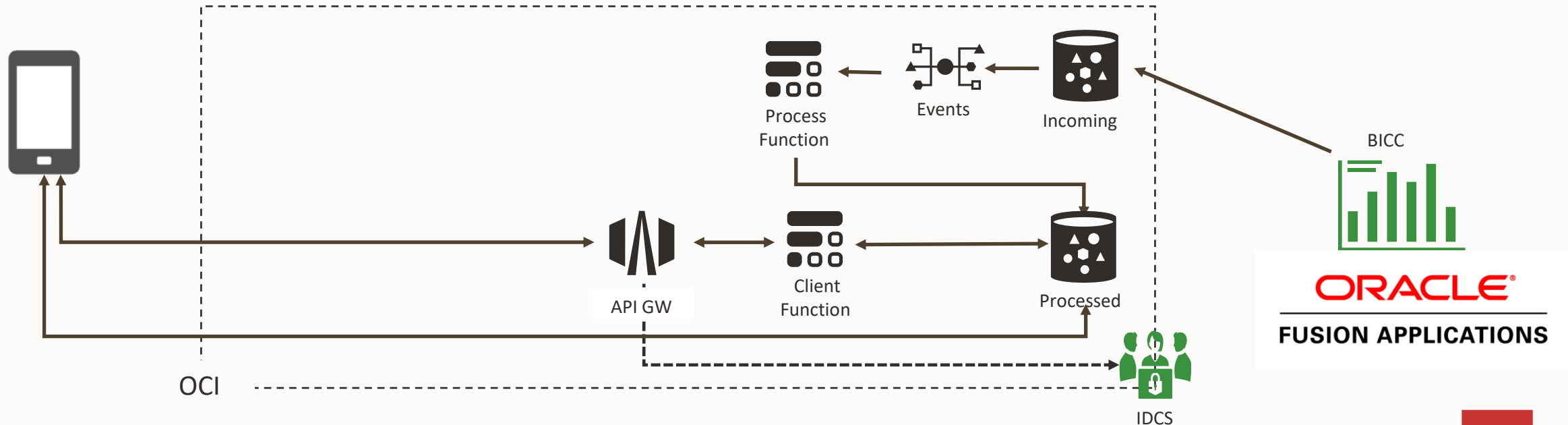




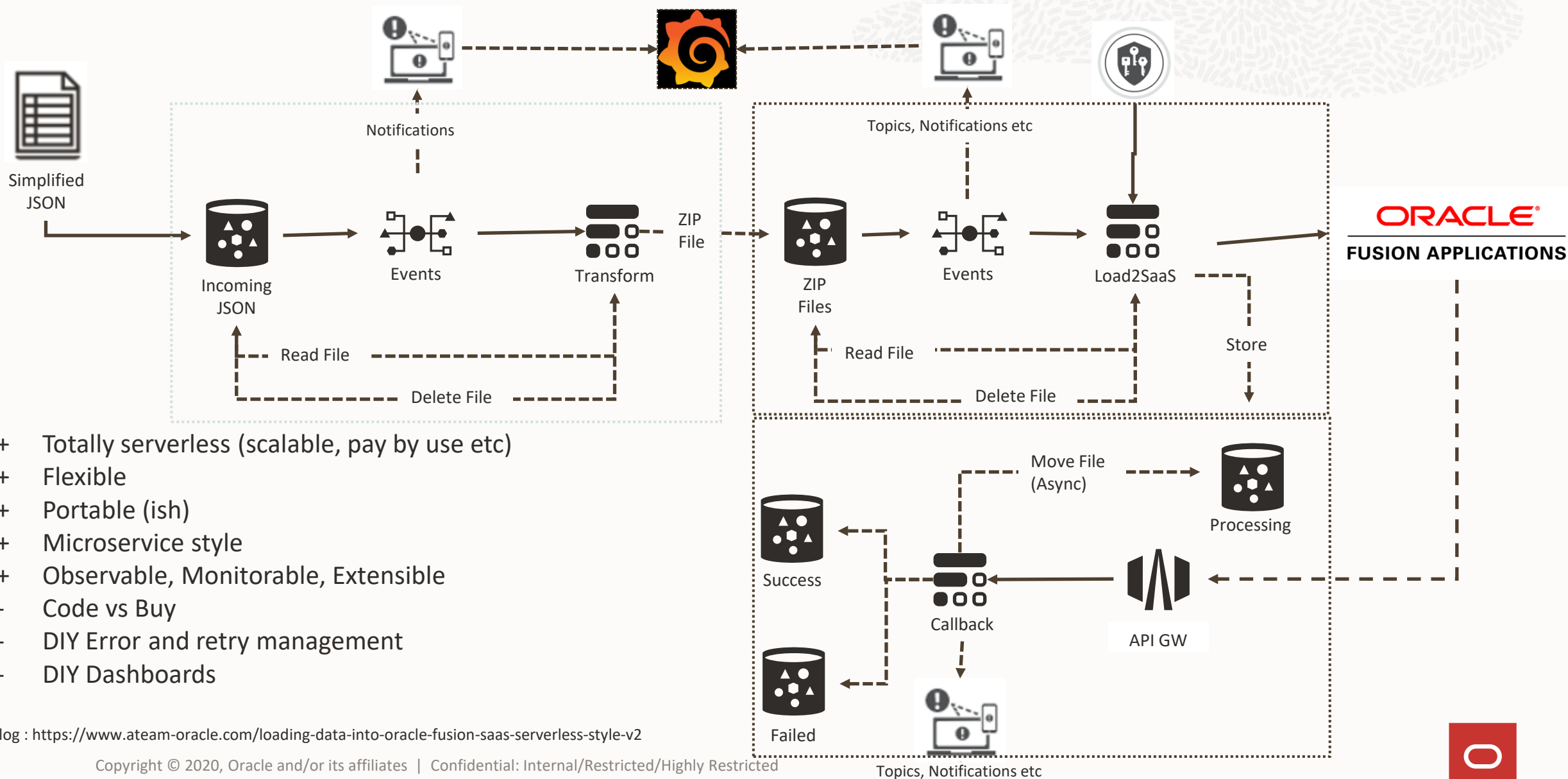
# Delivering Bulk Data to Mobile Devices

## The serverless way

1. Scheduled BICC runs store bulk extracted data on Storage Cloud
2. Storage cloud yields event which in turn calls Function
3. Function processes BICC Extract (JSON conversion + processing)
4. Resulting file stored on storage cloud for later download by mobile client
5. Function returns on-the-fly created pre-authenticated download URL (5a) for efficient download by mobile device (5b)



# Serverless Data Loading using OCI



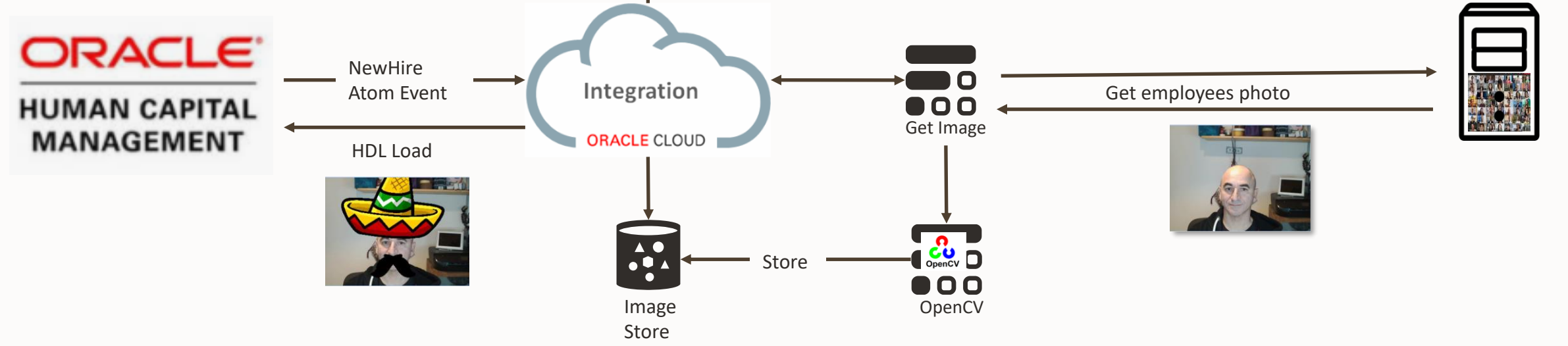
- + Totally serverless (scalable, pay by use etc)
- + Flexible
- + Portable (ish)
- + Microservice style
- + Observable, Monitorable, Extensible
- Code vs Buy
- DIY Error and retry management
- DIY Dashboards

Blog : <https://www.ateam-oracle.com/loading-data-into-oracle-fusion-saas-serverless-style-v2>



# HCM Atom Feed Example

If This Happens Then That

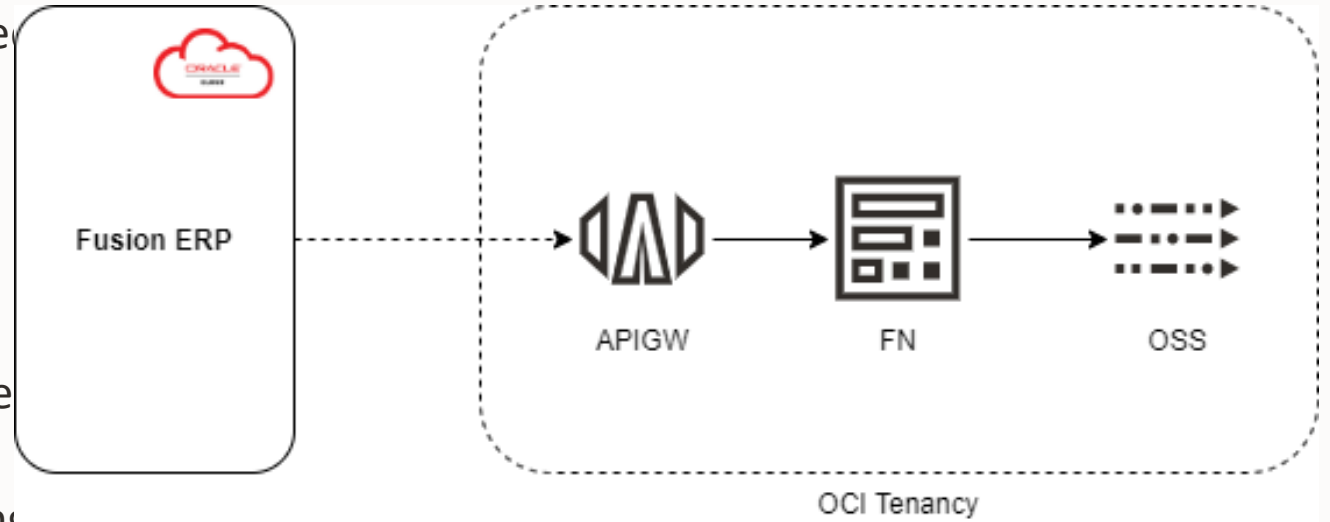


- User onboarded to HCM , image obtained from image server and company logo embossed
- Co-Existence of OIC and Serverless paradigms
- Classic use-case when a process is not needed often, why pay and maintain it



# Consume ERP events into OCI Streaming

- Create an OCI Function that listens to the required ERP events and pushes them to OCI Streaming Service.
- Expose the function to the internet using APIGW and register the APIGW endpoint to ERP specifying the events to expose
- Whenever an event occurs, a message is produced in OCI Streaming
- Other use-cases are possible as well, by modifying the Function





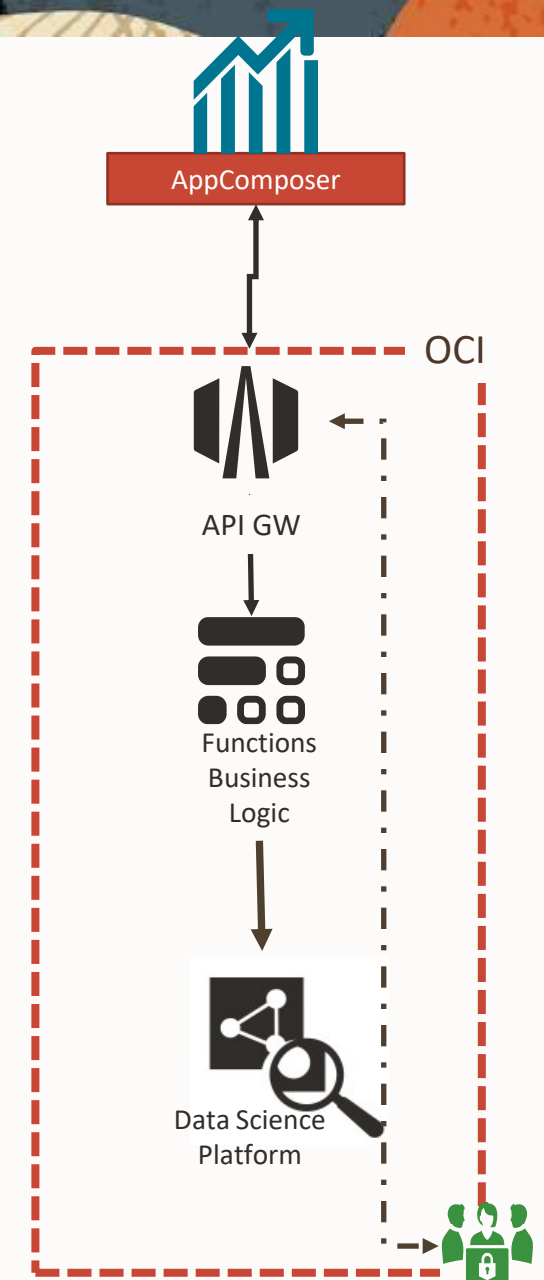
# Extension Usecases

---



# Extending SaaS business logic

- SaaS logic may be better executed outside of SaaS
  - E.g. State Machines, ML classifications, semantic analysis
- Still needs to be secured -> Use API Gateway
- Data Science Platform is preferred however you “could” do something custom in the function
- Functions isn't the only option you could use Kubernetes, VMs etc however they don't really fit the “SaaS” mantra of we manage nothing



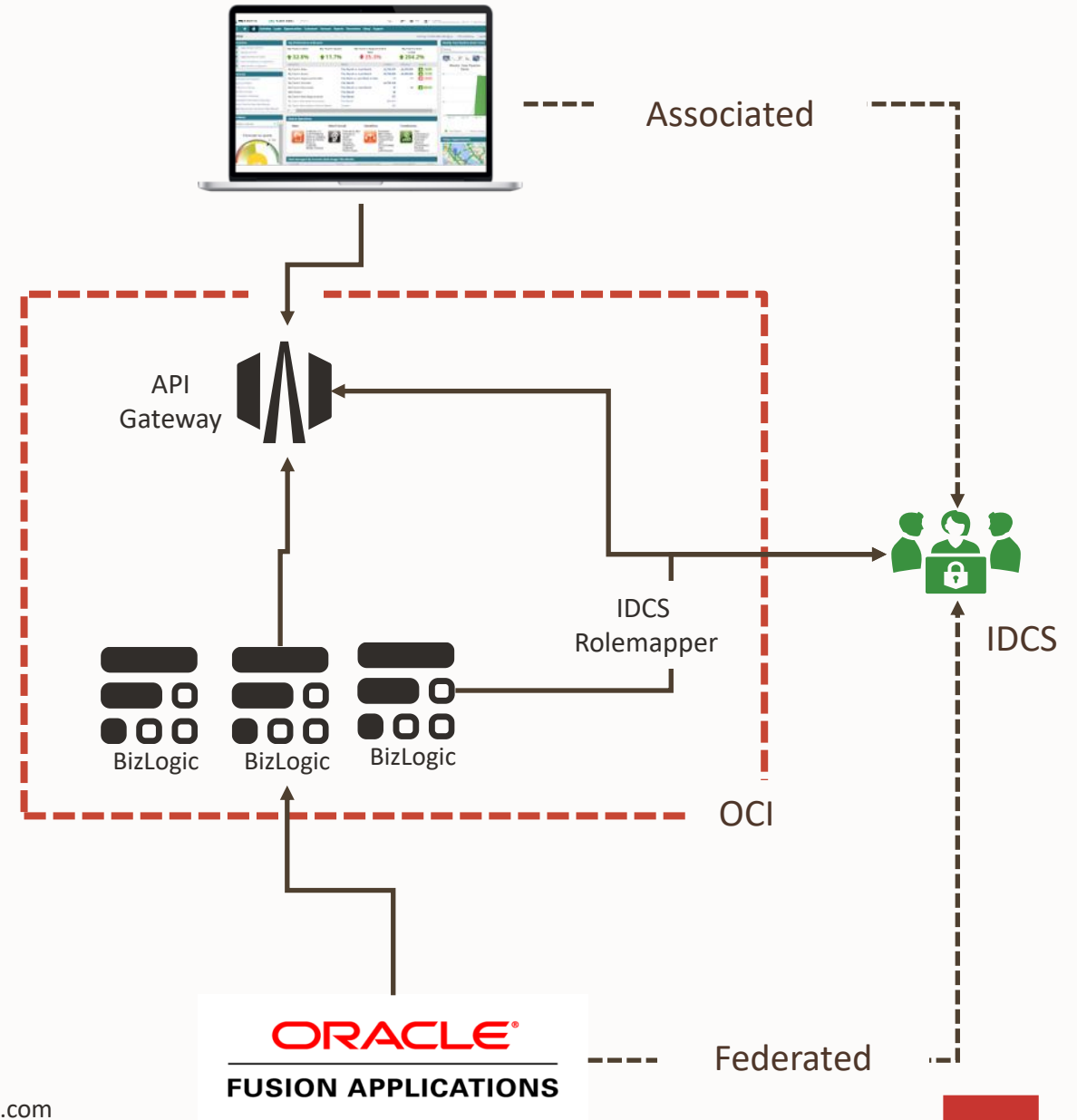
<https://blogs.oracle.com/datascience/deploying-a-machine-learning-model-with-oracle-functions>





# Using “Serverless” to Backend The UI

- Solution uses only serverless components , ideal for JCS/SX UI migrations
- Visual Builder for the UI
- API Gateway
  - Single API endpoint for the application (can be shared with other applications)
  - Maps HTTP REST verbs to Functions
  - Implements Authentication and SSL security
- Functions
  - Implements business logic, *optional* security token exchange
- Identity Cloud
  - Federated IDM between OCI & SaaS
- Key Management
  - Secure storage of secret data (e.g. oAuth Secret)



<https://github.com/oracle/cloud-asset-fusion-serverless-vbcs-sample> / [angelo.santagata@oracle.com](mailto:angelo.santagata@oracle.com)

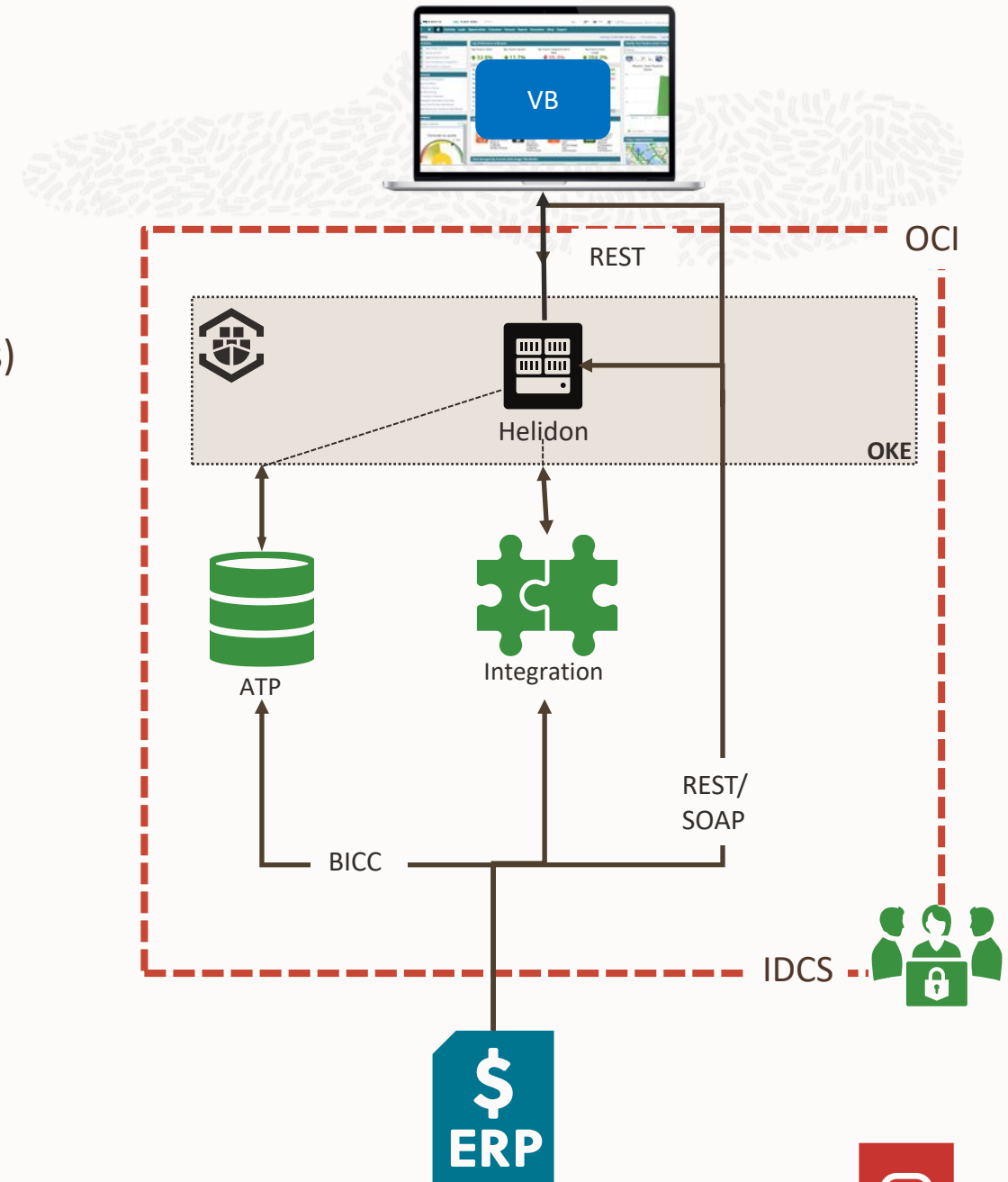
<https://docs.oracle.com/en/solutions/extend-saas-cloud-native/index.html>



# Need more than Serverless

## ERP Example using UI and Integration

- Customer Service UI extension (ePOS , order mgmt. etc)
  - Large number of users projected (6000 conc, ~150req/s)
  - Must protect ERP SaaS from end user workloads
  - Long running workloads
- Advantages
  - Elastic, servers can grow and shrink based on demand
  - Agile deployments- start small , grow big and be independent
  - Performance and Full control of cluster/mesh but no need to manage control plane
  - Can use advanced deployment techniques such A/B testing, canary deployments etc
- Challenges
  - Customer Managed
  - Full control = Full management and responsibility
  - Complex designs, do you use Istio/Linkerd?



# Conclusion



- Oracle Cloud Platform provides services which meet many SaaS customer needs
  - Cloud Native
  - Serverless
  - PaaS
- Strategically PaaS services offer the best entry point for most customer use cases
  - Or a mix!
- OCI Services now provide us with a new level of agility



A professional office setting where a woman in a blue blazer is listening attentively to a man in a grey suit who is gesturing with his hands. A laptop and a coffee cup are on the table in front of them. The background shows a modern office with large windows.

Questions?



# Oracle Team



## Your Local Partner Manager

Contact the OPN team to identify  
Your partner manager

[www.tinyurl.com/OracleOPN](http://www.tinyurl.com/OracleOPN)



## Angelo Santagata

Architect A-Team

E-Mail [angelo.santagata@oracle.com](mailto:angelo.santagata@oracle.com)

LinkedIn [www.linkedin.com/in/angelo-santagata-a64344](http://www.linkedin.com/in/angelo-santagata-a64344)



## Jürgen Kress

OIC & ODA Partner Adoption

E-Mail [Juergen.Kress@oracle.com](mailto:Juergen.Kress@oracle.com)

LinkedIn [www.linkedin.com/in/kress](http://www.linkedin.com/in/kress)

Twitter [@soacommunity](https://twitter.com/soacommunity)

Community [tiny.url.com/OraclePC](http://tiny.url.com/OraclePC)



**Thank You**

