THE COGNITIVE ERA
CASCON
2017 November 6-8

OpenAPI

Arthur De Magalhaes
API & Cloud Architect

IBM

November 8, 2017

Agenda

- Swagger & WebSphere Connect
- OpenAPI v3
- APIs & Cloud
- Open Liberty & MicroProfile
- Labs Intro

Swagger & WebSphere Connect

Swagger Introduction

- Industry leading specification for defining REST APIs.
- Supports both JSON and YAML formats.
- Large open source community with various projects on GitHub:
  - Client code generation (37 languages)
  - Server code generation (23 languages)
  - Online editor and GUI
  - Over 2000 related open-source repositories, with 15000 daily downloads
- Base specification for OpenAPI Initiative (https://openapis.org/), under Linux foundation.

Swagger Annotations

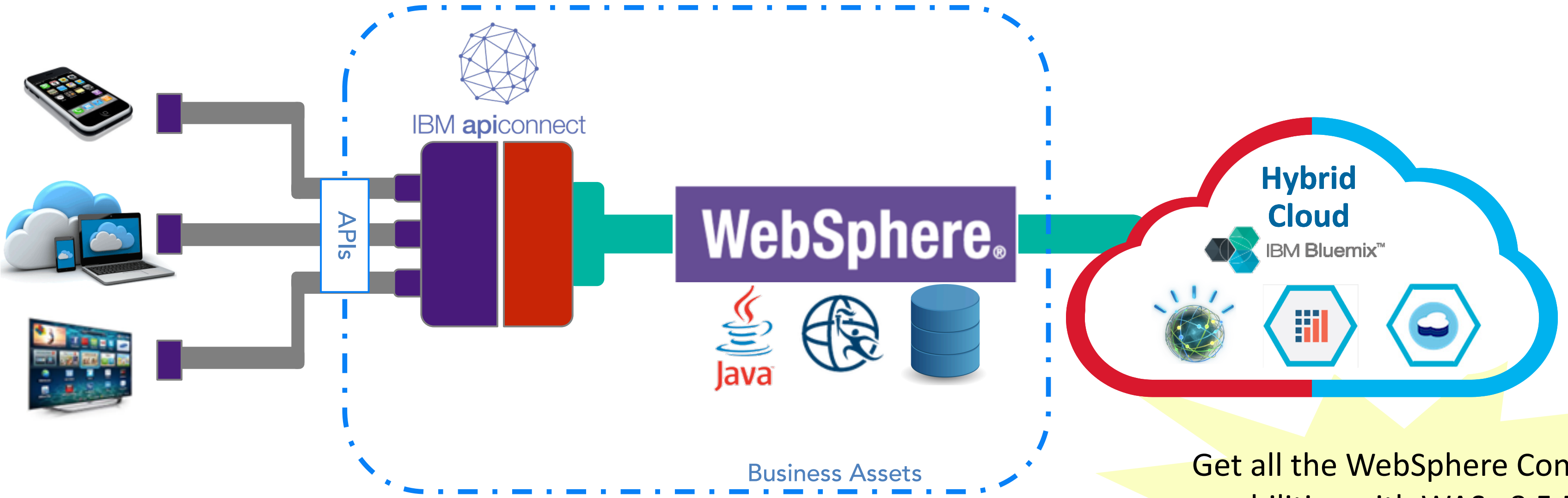
- Documents the RESTful endpoint.
- Always in-sync, since it’s together with the code.

```java
@POST
@Consumes({ "application/json", "application/xml" })
@Produces({ "application/json", "application/xml" })
@ApiOperation(value = "Add a new pet to the store",
              response = void.class,
              authorizations = {
                @Authorization(value = "petstore_auth", scopes = {
                    @AuthorizationScope(scope = "write:pets", description = "modify pets in your account")
                })
})
@ApiResponses(value = {
    @ApiResponse(code = 405, message = "Invalid input", response = void.class) })
public Response addPet(Pet body) {
```

Swagger Yaml Example

```
paths:
  /pet:
    post:
      summary: Add a new pet to the store
      consumes: <mime type>
      produces: <mime type>
      parameters:
        - in: body
          name: body
          description: Pet object that needs to be added to the store
          required: true
          schema: $ref: '#/definitions/Pet'
      responses:
        '405': description: Invalid input
      security:
        - petstore_auth:
            - 'write:pets'
```

WebSphere Connect Overview



Get all the WebSphere Connect capabilities with WAS v8.5.5 and WAS v9

Supported by both Liberty and traditional WAS

WebSphere Connect is…

- A collection of capabilities **built into WAS** that help you **turn backend WAS assets into APIs** and **connect to and from the hybrid cloud** to rapidly extend the value of their application investments
- WebSphere Application Server v8.5.5 and v9 (Liberty Core, Base, ND, z/OS) includes API Connect Essentials as a Supporting Program, adding IBM Support and extra API calls per month

Why "WebSphere Connect"?

We want to change the way you think about constructing applications…

…leveraging not only an ecosystem of APIs, but also a rich set of cloud-provided services…

…without having to perform significant re-engineering of existing applications…

…whether they reside on-premises or in the cloud.

WebSphere Liberty's API Discovery

apiDiscovery-1.0 feature

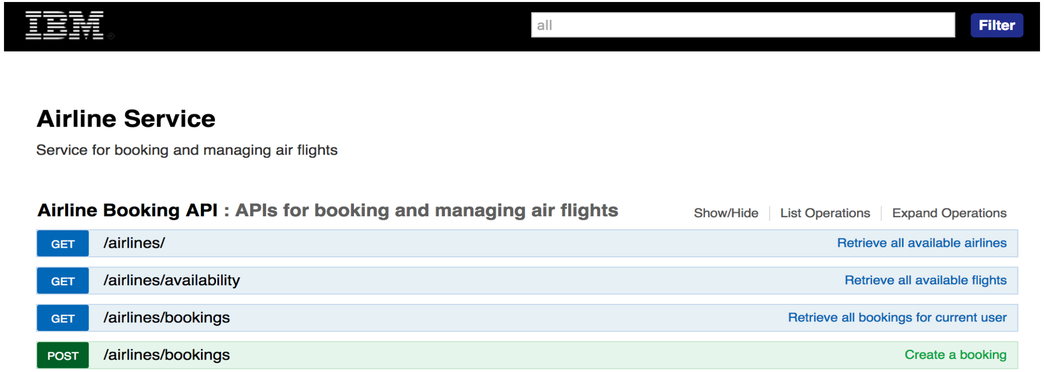
```java
@GET
@Produces("application/json")
public Response getFlights(@QueryParam("departureDate") String departureDate,
        @QueryParam("airportFrom") String airportFrom,
        @QueryParam("returningDate") String returningDate,
        @QueryParam("airportTo") String airportTo,
        @QueryParam("numberOfAdults") int numberOfAdults,
        @QueryParam("numberOfChildren") int numberOfChildren){
    return Response.ok().entity(findFlights(airportFrom, airportTo, departureDate, returningDate
}

private static List<Flight> findFlights (String airportFrom, String airportTo, String departureDa

    List<Flight> flights = new ArrayList<Flight>(6);
```

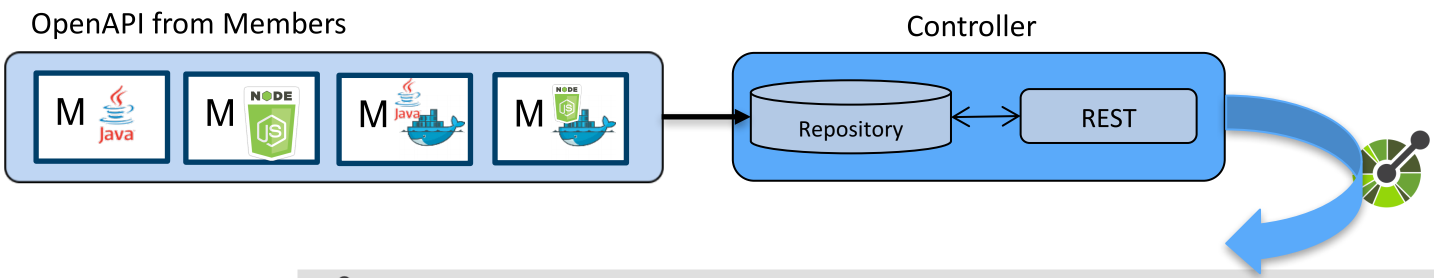
```java
@SwaggerDefinition(
        info = @Info(description = "Service for booking and managing air flights",
                     version = "1.0.1",
                     title = "Airline Service"),
        tags= {@Tag(name="Airline Booking API",
                    description="APIs for booking and managing air flights")})
public class JAXRSApp extends Application {
```

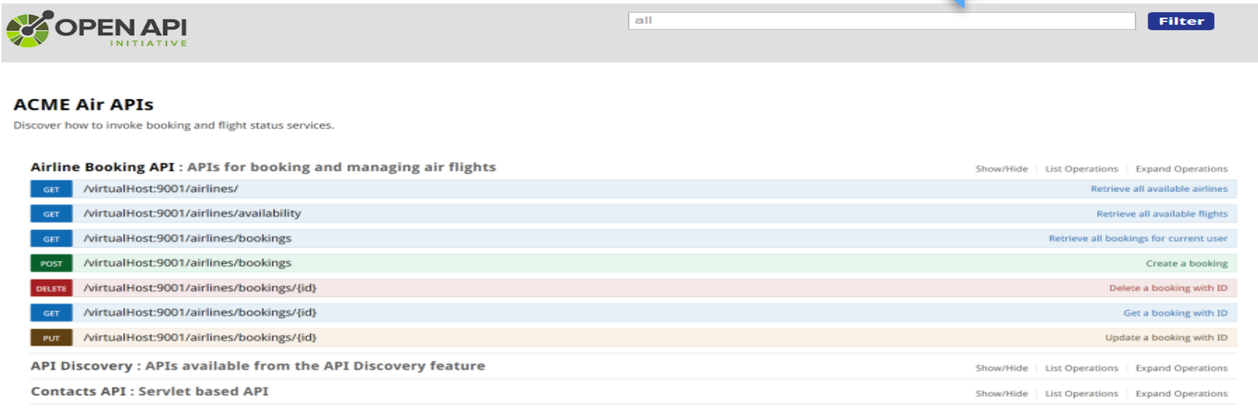

```json
{
   "swagger": "2.0",
   "info": {
      "description": "Service for booking and managing air flights",
      "version": "1.0.1",
      "title": "Airline Service"
   },
   "host": "libertyapi.mybluemix.net:443",
   "tags": [
      {
         "name": "Airline Booking API",
         "description": "APIs for booking and managing air flights"
      }
   ],
   "definitions": {
      "Airline": {
         "type": "object",
         "required": [
            "contactPhone",
            "name"
         ],
         "properties": {
            "name": {
```

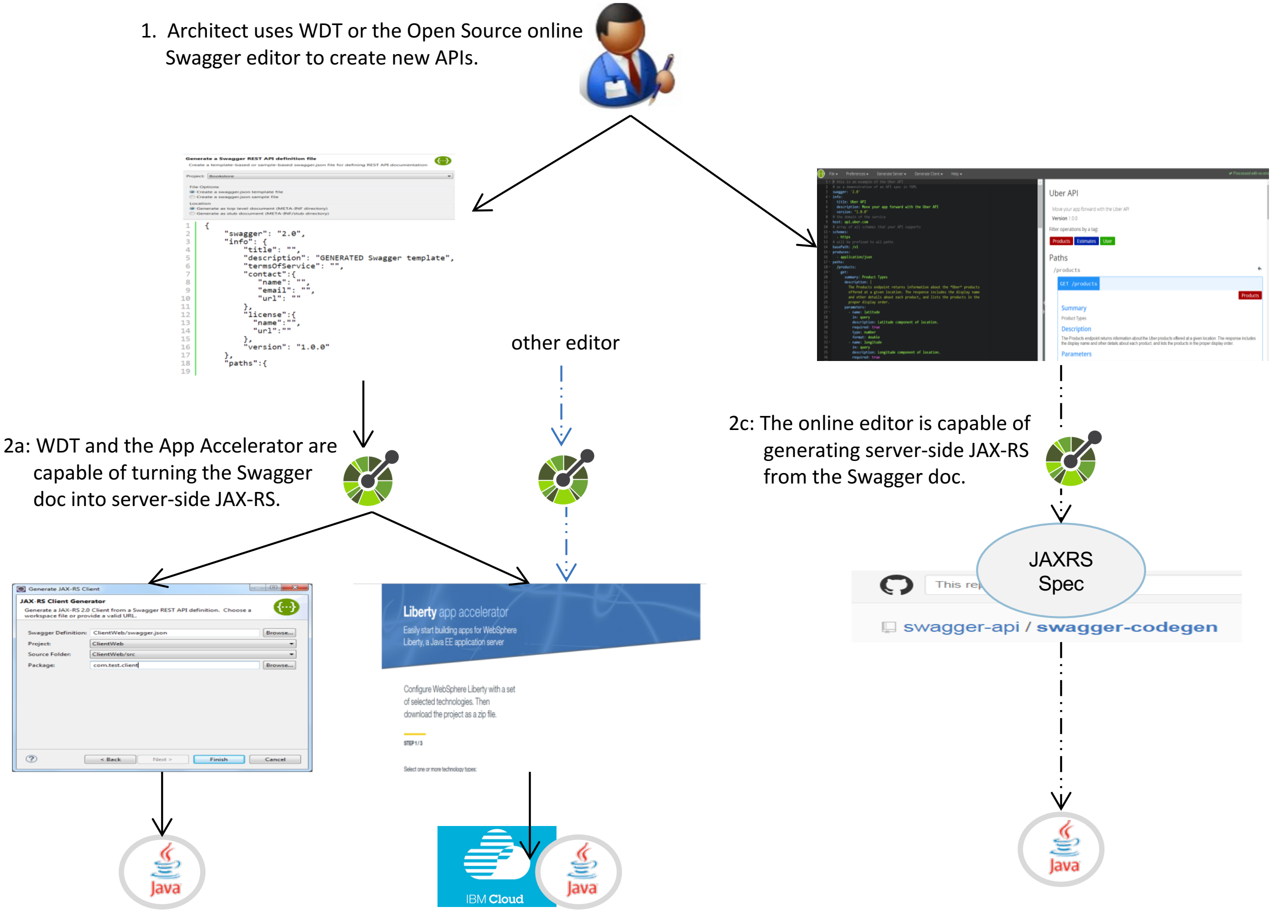
WebSphere Collective OpenAPI Support

- Adds dynamic routing and auto-scaling for Java and Node members
- Can be deployed inside Docker containers
- Aggregates OpenAPI for all clusters and collective members

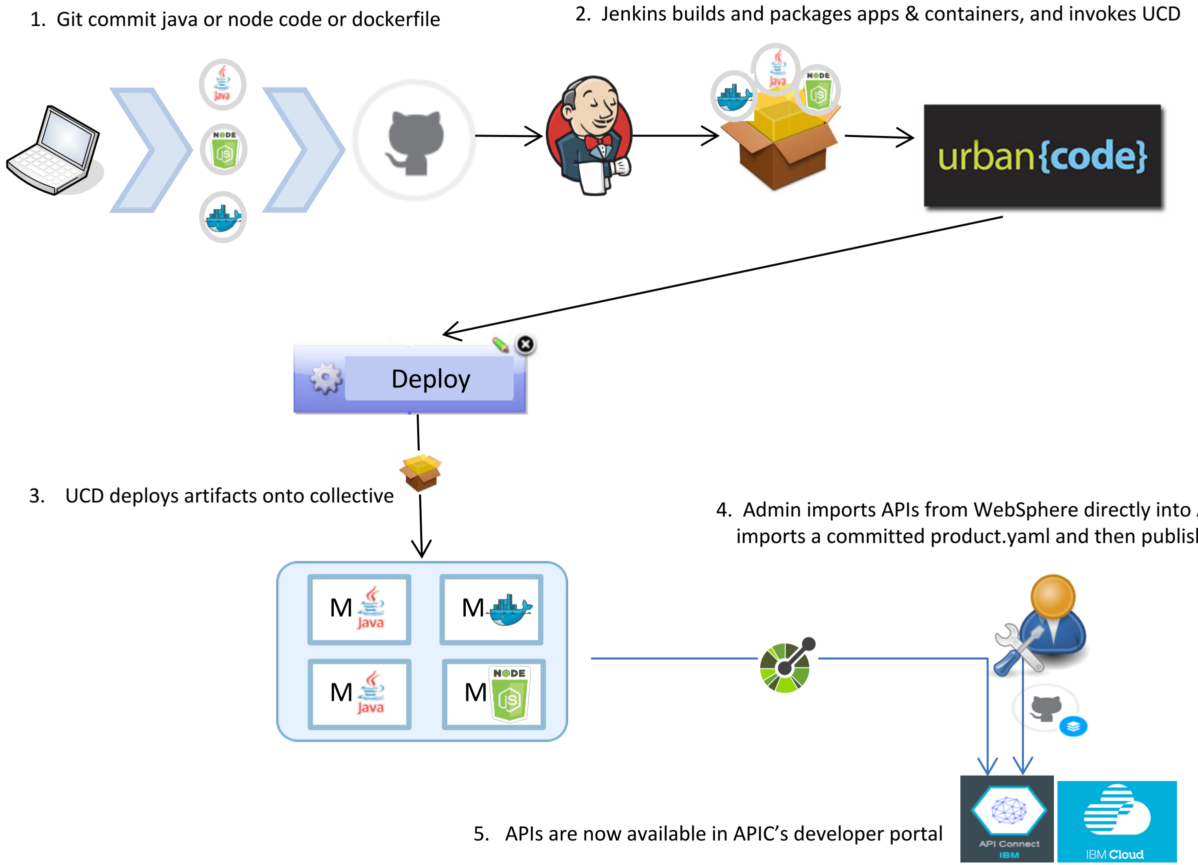




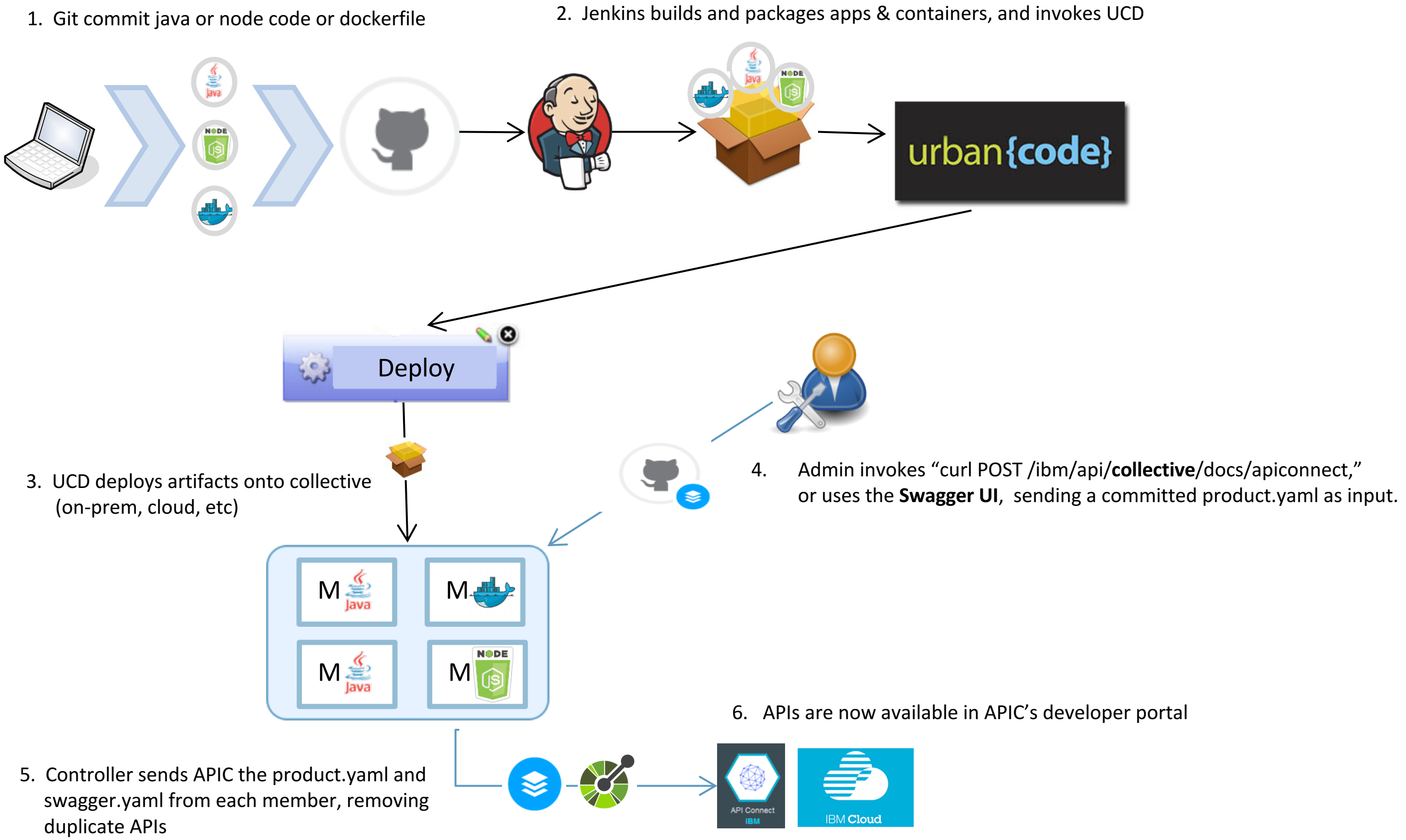
Design-first approaches



CI flow #1: manual pull of APIs after deployment



CI flow #2: manual push of APIs after deployment



OpenAPI v3

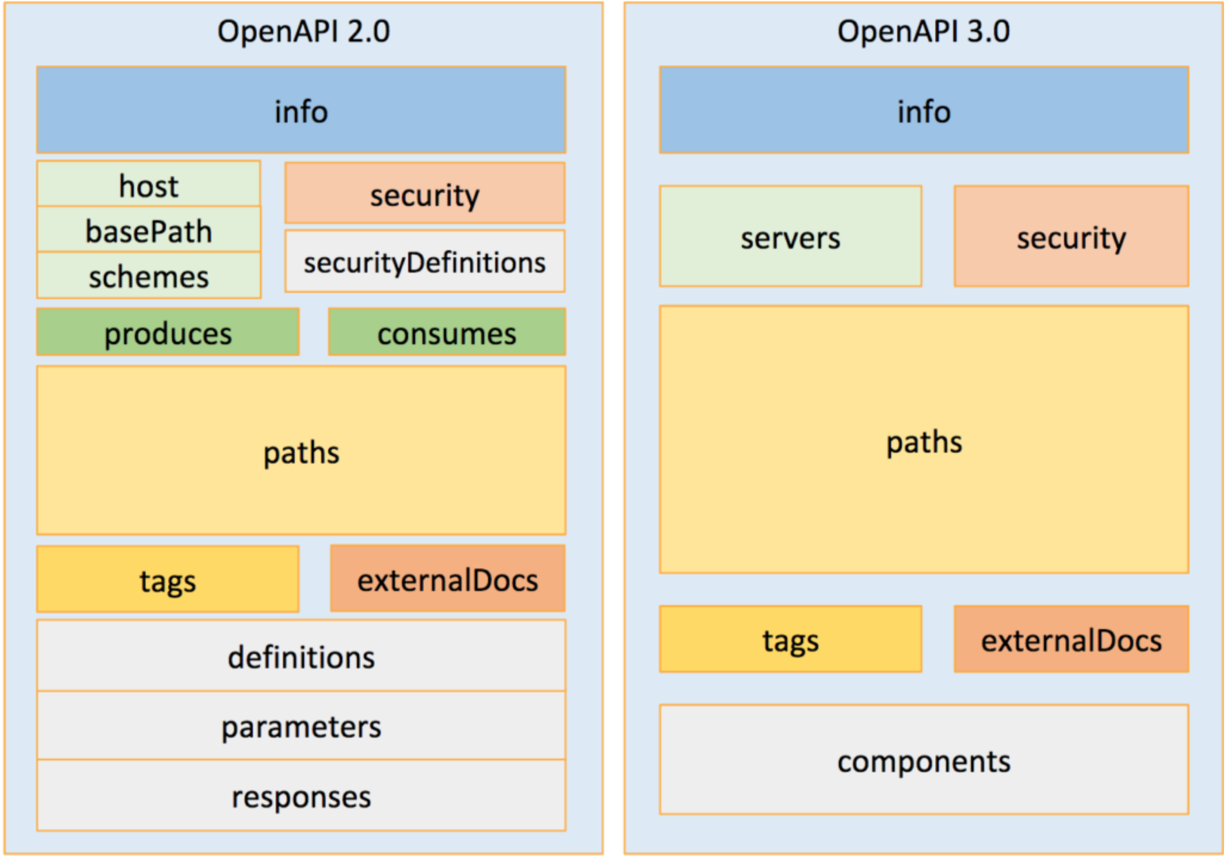
OpenAPI

- A simple, standard API description format
    - Started by Swagger, specification donated to Linux Foundation
    - Widely adopted, supported, vendor neutral
- Version 2 released 10/2014
- Version 3 released 07/2017

OAI Members

3scale BY RED HAT
42crunch
Adobe
{"logo":"API Evangelist"}
Atlassian
ca technologies
Capital One
Cloud Elements
Google
IBM
Intento
Isa
mashape
Microsoft
ORACLE + apiary
Restlet
salesforce
ARTIK Cloud
SMARTBEAR
software AG
Tyk.io
MuleSoft

OpenAPI V2 – V3




```
{
  “openapi” : “3.0.0”,
  “info” : {…},
  “servers” : {…},
  “paths” : {…},
  “components” : {…},
  “security” : {…},
  “tags” : {…},
  “externalDocs” : {…}
}
```

Servers

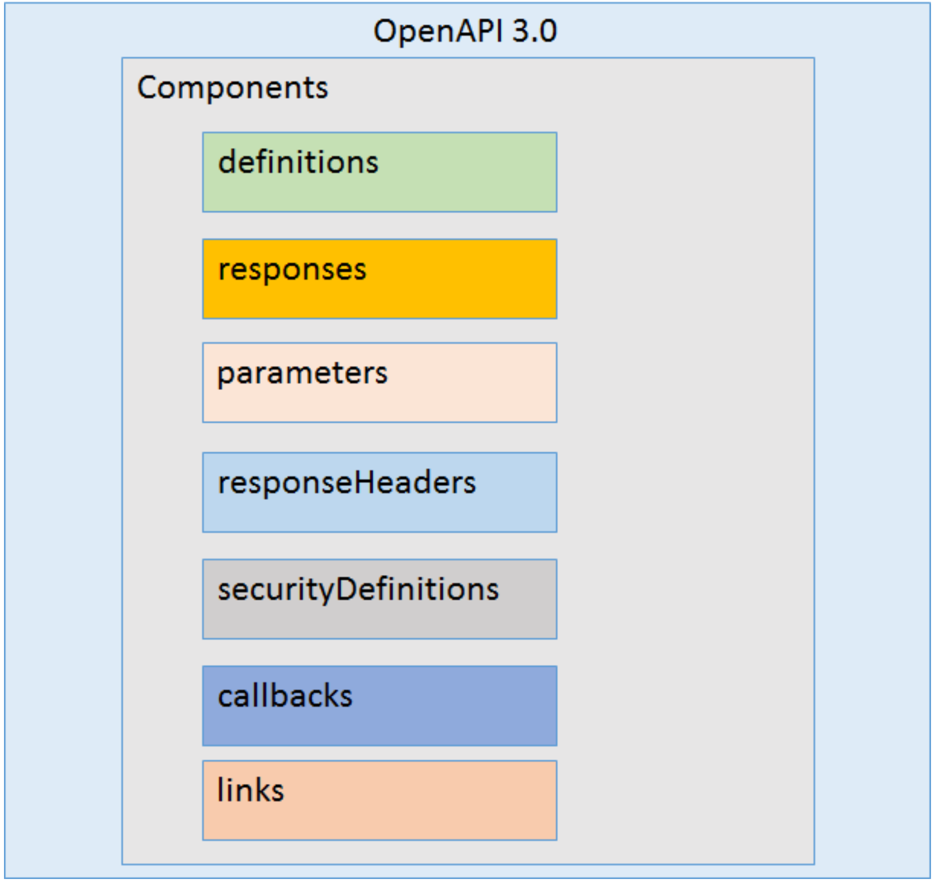
```yaml
servers:
- url: https://development.gigantic-server.com/v1
  description: Development server
- url: https://staging.gigantic-server.com/v1
  description: Staging server
- url: https://api.gigantic-server.com/v1
  description: Production server
```

- Operations can override their own servers object

Servers - Variables

```yaml
servers:
- url: https://{username}.gigantic-server.com:{port}/{basePath}
  description: The production API server
  variables:
    username:
      # note! no enum here means it is an open value
      default: demo
      description: this value is assigned by the service provider, in this example `gigantic-server.com`
    port:
      enum:
        - 8443
        - 443
      default: 8443
    basePath:
      # open meaning there is the opportunity to use special base paths as assigned by the provider, defaul
      default: v2
```

Components


OpenAPI 3.0
Components
definitions
responses
parameters
responseHeaders
securityDefinitions
callbacks
links

Callbacks

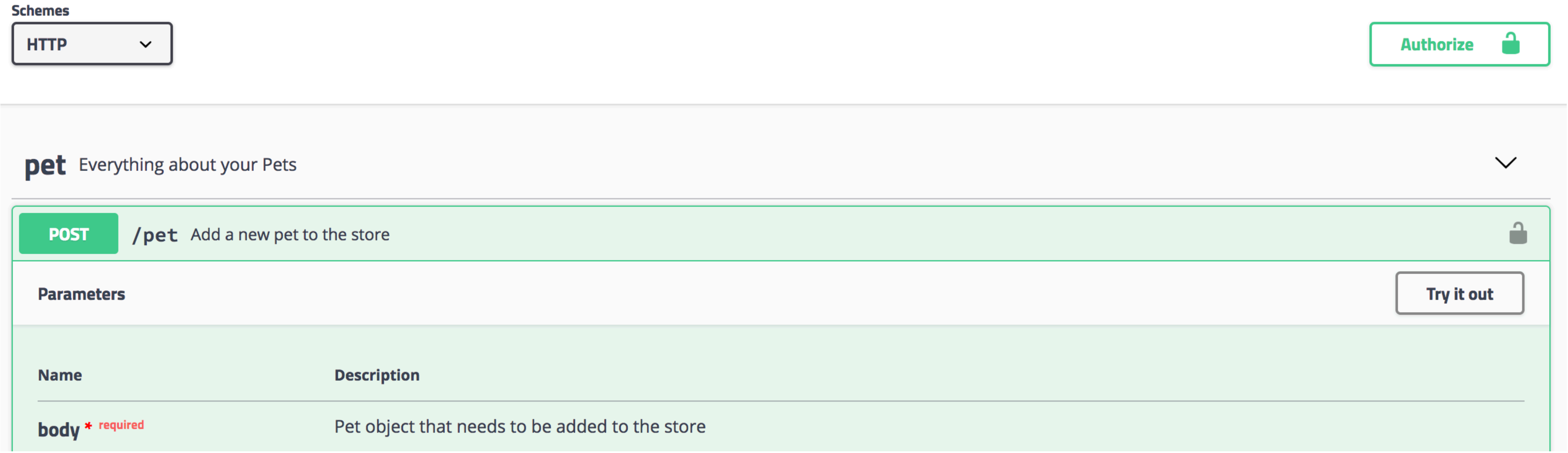
```
paths:
  /repos/{owner}/{repo}/hooks/{id}:
    post:
      body: |
        description: Repository service hooks (like email or Campfire) can have at most one
        Repositories can have multiple webhooks installed. Each webhook should have a uniqu
        schema:
          $ref: '#/components/definitions/WebhookRequest'
      responses:
        201:
          description: Webhook created
          schema:
            $ref: '#/components/definitions/WebhookResponse'
      callbacks:
        createWebhook:
          $ref: '#/components/callbacks/createWebhook'
```

Callbacks (2)

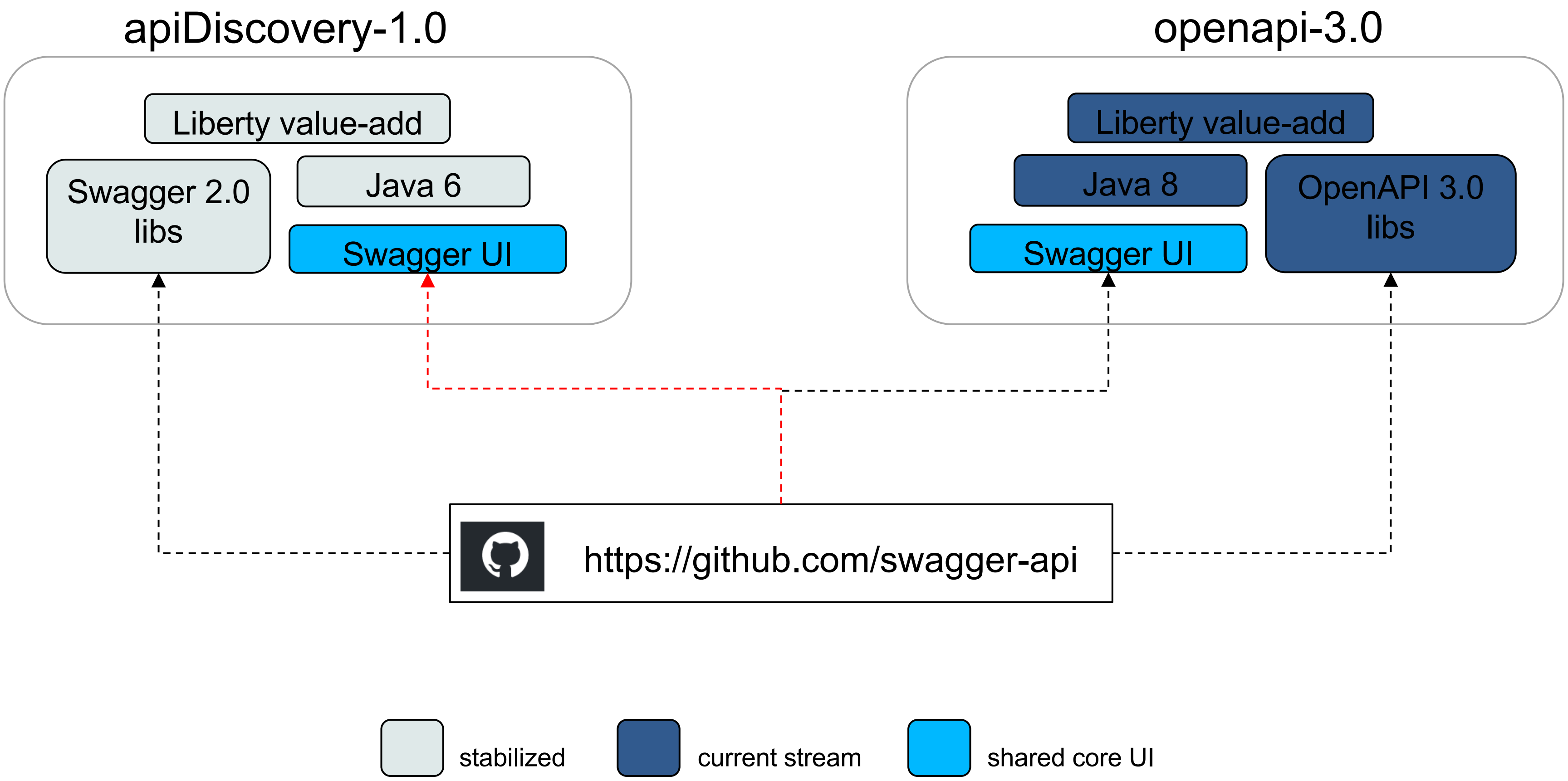
```
components:
  callbacks:
    createWebhook:
      # extract the url parameter from the request
      '$request.body.url':
        post:
          parameters:
          - in: header
            name: X-GitHub-Event
            type: string
            description: Name of the [event](https://developer.github.com/webhooks/#events)
            required: true
          - in: header
            name: X-Hub-Signature
            type: string
            description: HMAC hex digest of the payload, using the [the hook's `secret`](ht
          - in: header
```

OpenAPI User Interface

- New UI: http://petstore.swagger.io/



API support in WebSphere (Liberty 17.0.0.3)



APIs & Cloud

Bluemix is now IBM Cloud

https://www.ibm.com/blogs/bluemix/2017/10/bluemix-is-now-ibm-cloud/

IBM Cloud - Runtimes on CloudFoundry

Buildpacks

Liberty for Java™

Develop, deploy, and scale Java web apps with ease. IBM WebSphere

IBM

SDK for Node.js™

Develop, deploy, and scale server-side JavaScript® apps wit

IBM

Boilerplates

Java Cloudant Web Starter

Use the Cloudant NoSQL DB service with the 'Liberty for Java'

IBM

Java Workload Scheduler Web Starter

This application demonstrates how to use the Workload Scheduler

IBM

LoopBack Starter

This is a sample StrongLoop LoopBack Node.js application,

IBM

MobileFirst Services Starter

Start building your next mobile app with mobile services on Bluemix.

IBM

Node.js Cloudant DB Web Starter

Use the Cloudant NoSQL DB service with the 'SDK for Node.js

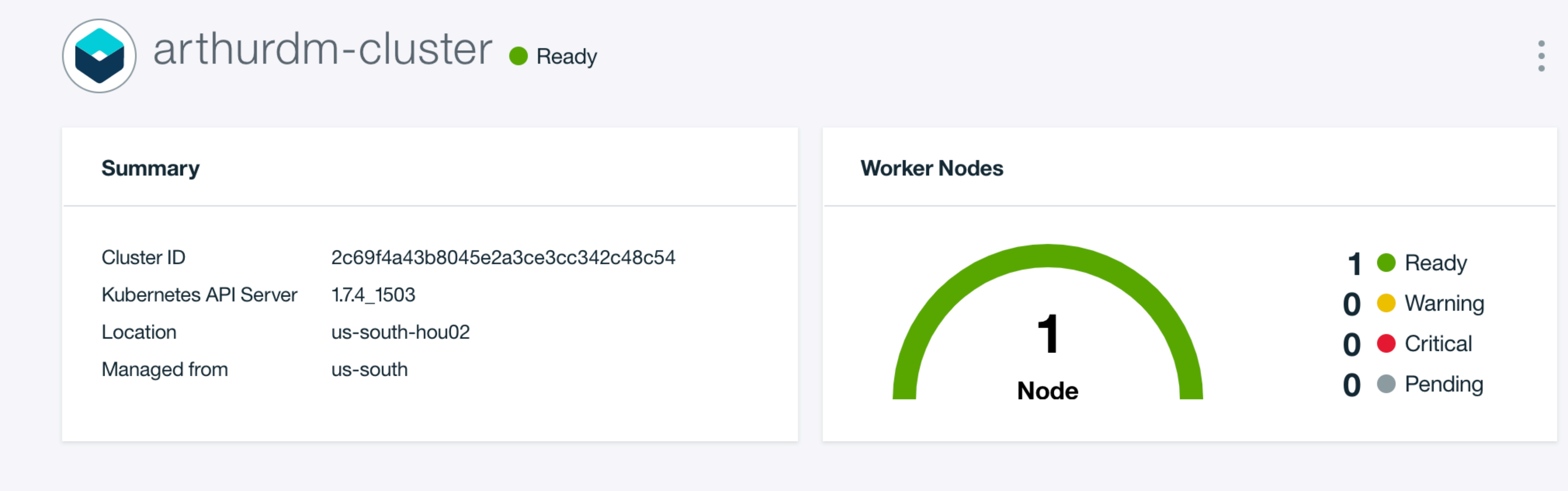
IBM

Personality Insights Java Web Starter

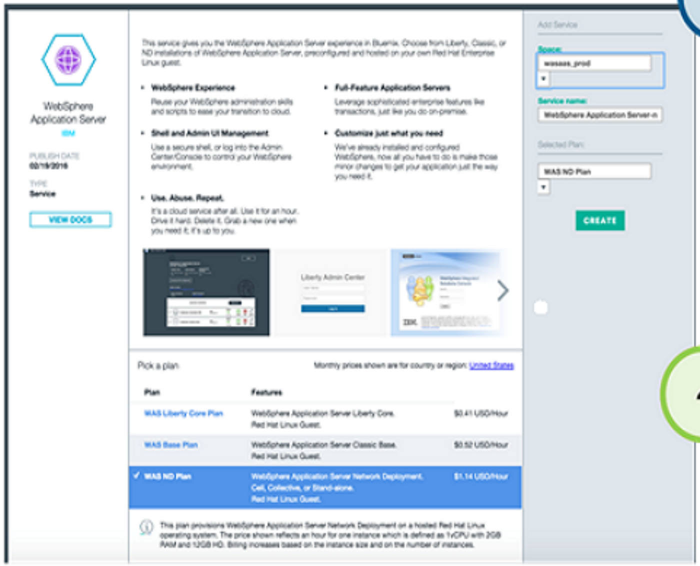
A simple Java app that uses the Personality Insights service to

IBM

IBM Cloud - Container Service



IBM Cloud – WAS on Cloud



1

WebSphere Application Server workloads

Deploy *Virtual Machines* based on – WAS ND, Base, & Liberty Core

2

Integrated Bluemix UI Portal

Includes *Simple* administration, management, monitoring, connectivity info, ..etc

3

Single-Tenant Infrastructure

Deploy WAS for Bluemix VM on *Private Bare metal hardware* in which the Physical Cores, Memory, and Storage is Isolated to one customer.

4

Additional Cloud Services

Intelligent workload placement, VPN, Firewall, DirectLink High Speed Connection up to 10Gps, & Softlayer VLAN Spanning to expand to additional customer solutions

IBM Cloud Private

http://ibm.biz/IBMCloudPrivate

Community

Open Source*

Toolchains & Runtimes
Jenkins
Apache Tomcat
Open Liberty

Messaging
RabbitMQ

Data Services
MongoDB
PostgreSQL
Redis

Clustering
Galera

Http Servers
Nginx

Terminal Access
Web Terminal

IBM Software

Toolchain & Runtimes
IBM Microservice Builder
IBM WebSphere Liberty for Developers
IBM SDK for Node.js

Messaging
MQ Advanced for Developers

Data Services
IBM Db2 Dev-C
IBM Db2 Warehouse Dev-C
IBM Data Server Manager (for Db2 Dev-C)
IBM Cloudant Developer Edition

Data Science
IBM Data Science Experience Developer Ed.

Integration
IBM Integration Bus for Developers
IBM DataPower Gateway for Developers

App Modernization Tooling
IBM Transformation Advisor

Monitoring
IBM Cloud APM for DevOps (Beta)

HPC
IBM Spectrum LSF Community Edition

**Open source is not warranted by IBM unless otherwise specified*

Cloud Native

Access to Community content, plus...

Toolchain & Runtimes
IBM Microservice Builder
IBM WebSphere Liberty
IBM SDK for Node.js

Multi-Cloud Management
IBM Cloud Automation Manager

Cloud Foundry Buildpacks (add-on)
IBM WebSphere Liberty,
Node.js, Swift, .Net

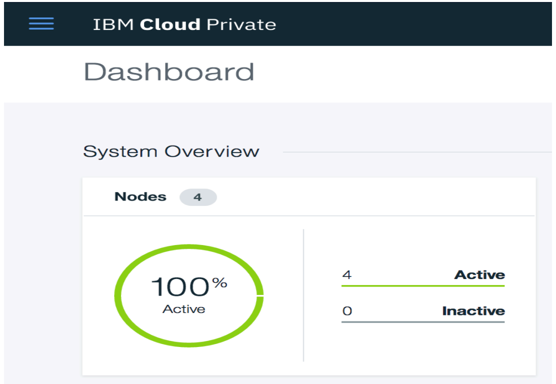
Enterprise

All Cloud Native content, plus...

Toolchain & Runtimes
IBM WebSphere Application Server ND [1]
IBM UrbanCode Deploy (add-on charge) [1]

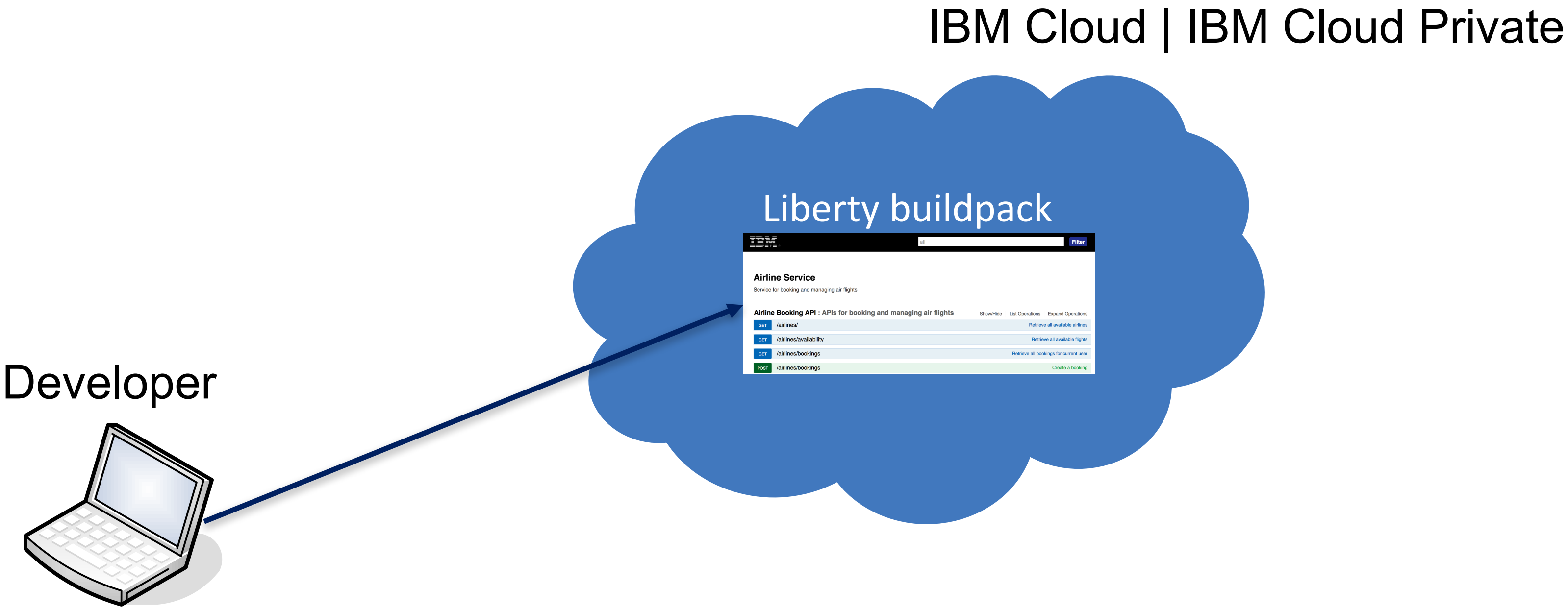
Integration and Messaging
IBM MQ Advanced
IBM API Connect [1]

Data Services
IBM Db2 Direct Advanced Edition with
Data Server Manager (add-on charge)



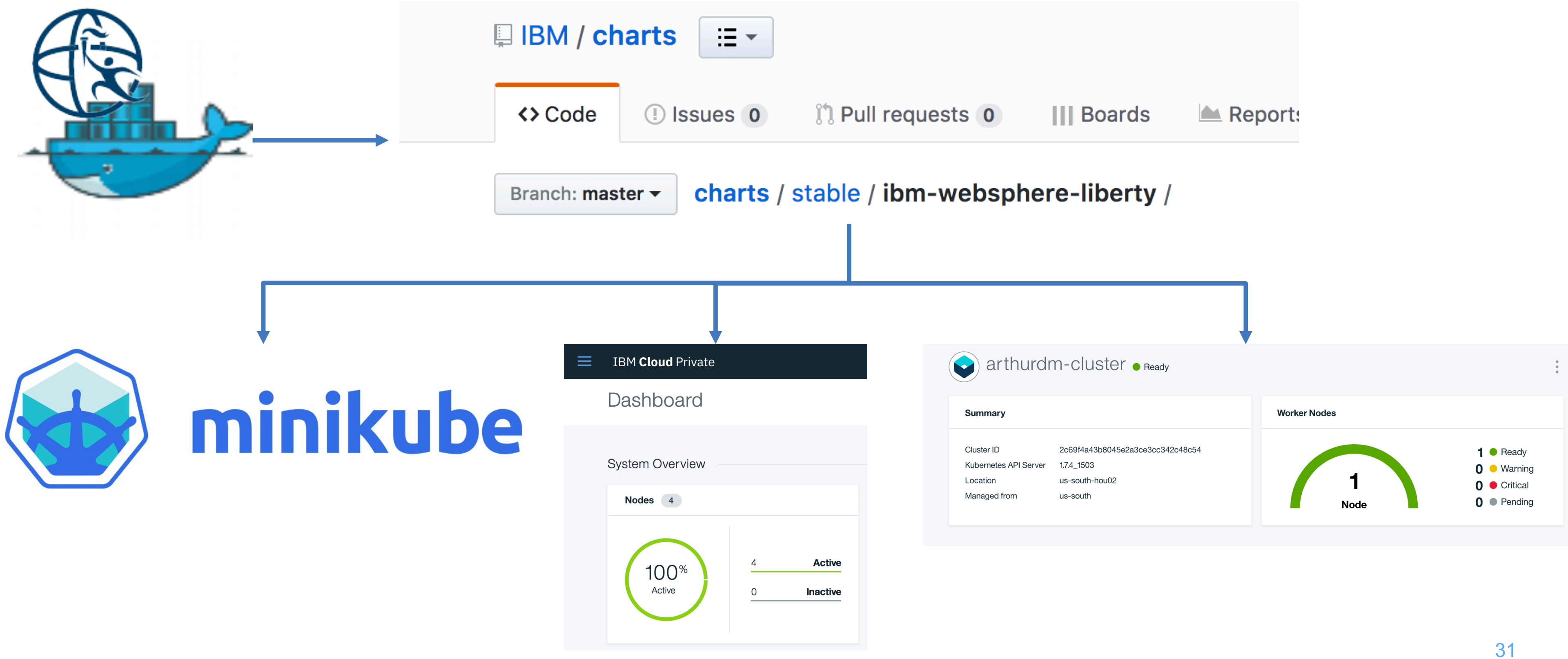
API Discovery on IBM Cloud

- IBM Cloud allow server packages to be pushed.
- Creates an auto-discoverable container in the cloud.



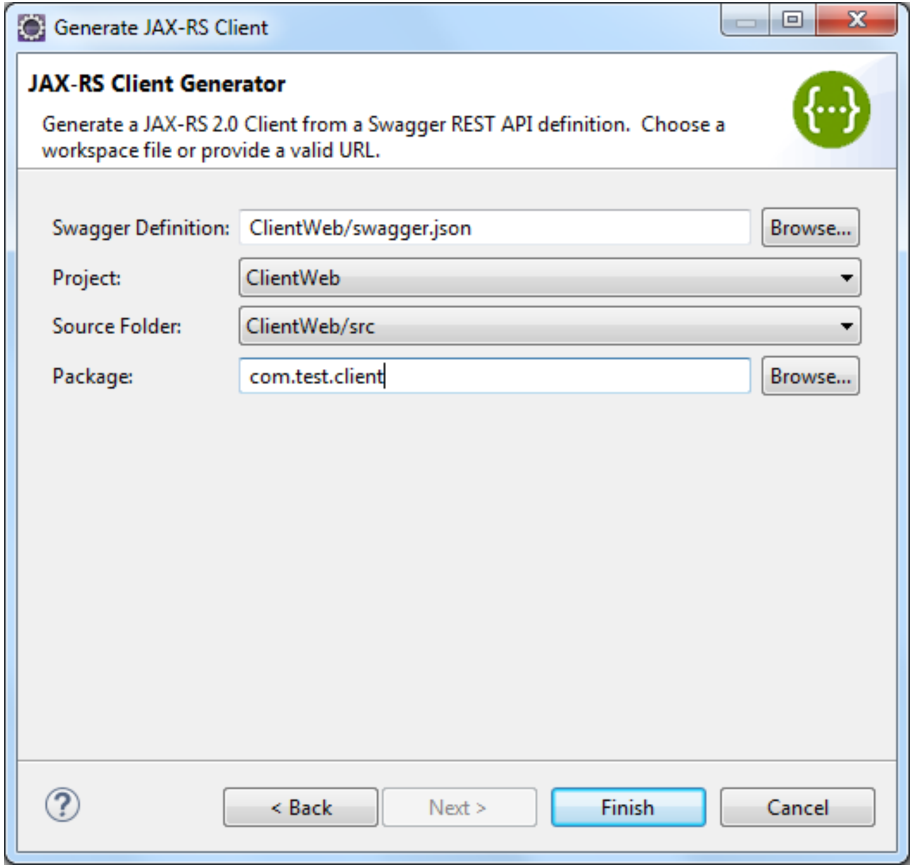
WebSphere Liberty Helm chart

github.com/IBM/charts

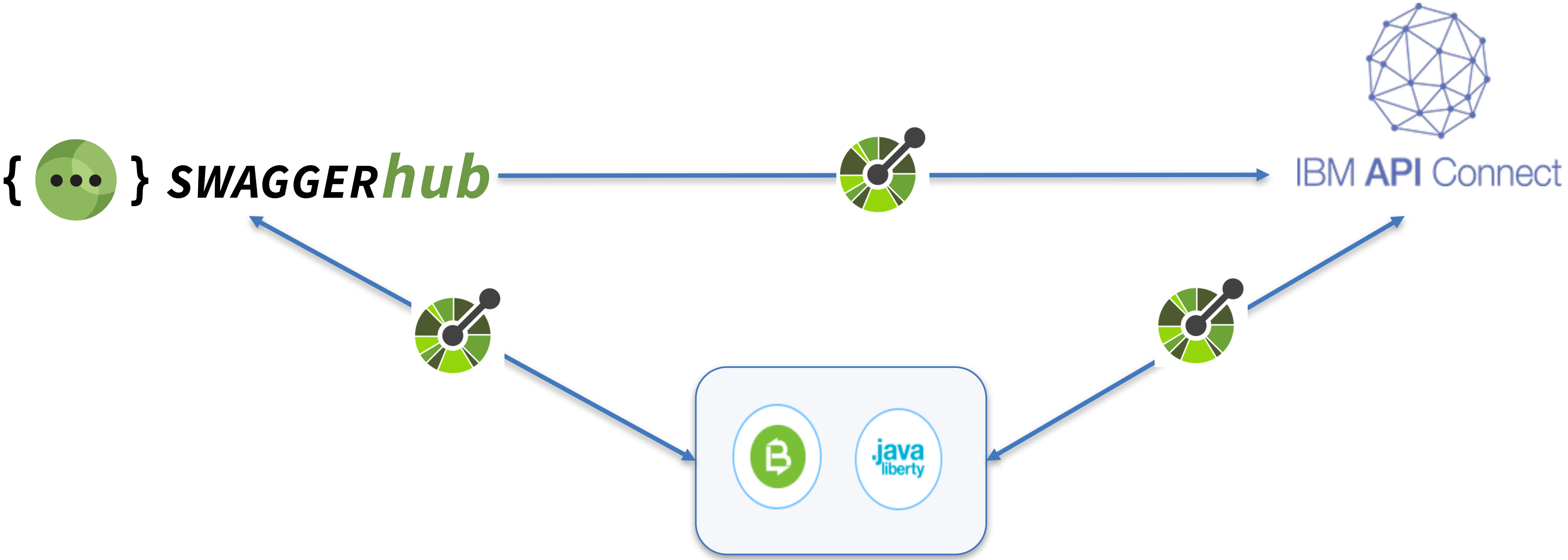


Connect to cloud services

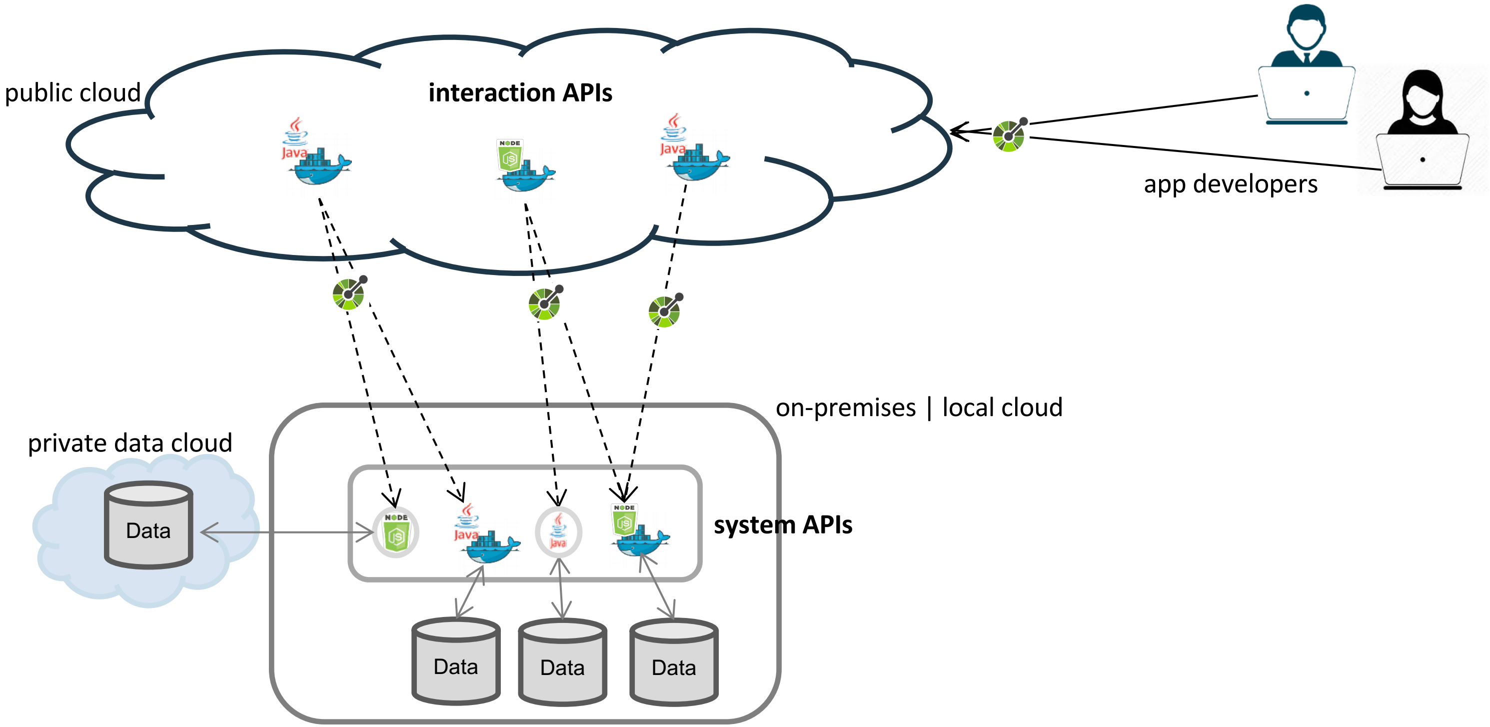
- Call any Swagger REST API
- Enhance your app with value-adds from a variety of microservices
- Add logic and deploy from within the same eclipse instance.



OpenAPI lifecycle



Hybrid cloud - via OpenAPI



Open Liberty & MicroProfile

OpenAPI v3 Community landscape

SMARTBEAR IBM

OPEN API INITIATIVE

https://github.com/swagger-api

27 OSS projects
Annotations, Parser, Codegen, UI, Node, etc

Swagger Spec
~~1.0, 1.1, 1.2~~, 2.0

OpenAPI Spec 3.0

supports

supports

Community Collaboration

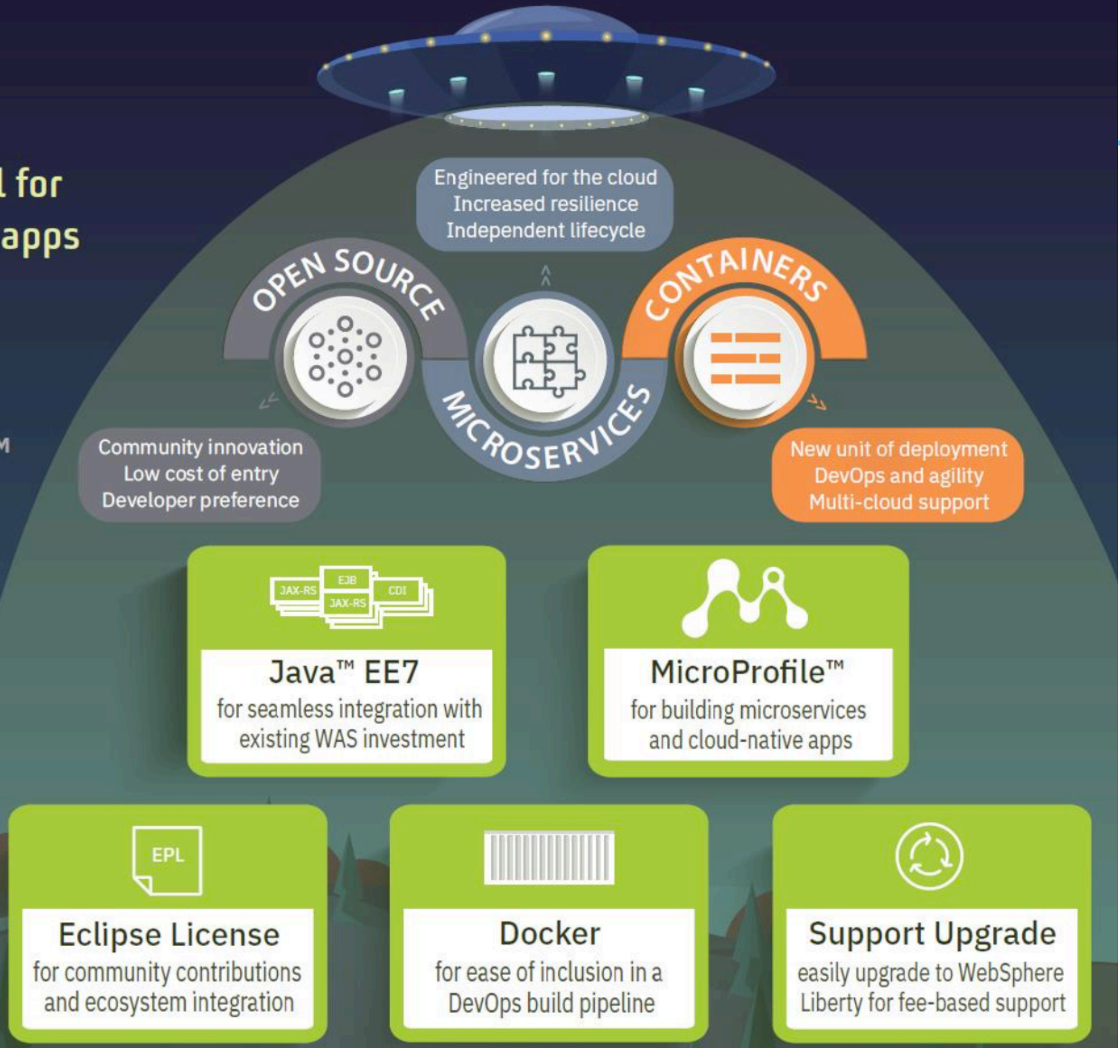
- Stage 1: collaborate with SmartBear on Open Source annotations, parser, converter, jaxrs, etc
- Stage 2: collaborate with Red Hat, SmartBear and MP community on plan for including the Java programming model into MicroProfile

Open Liberty

A lightweight open source server runtime ideal for building Java™ microservices and cloud-native apps

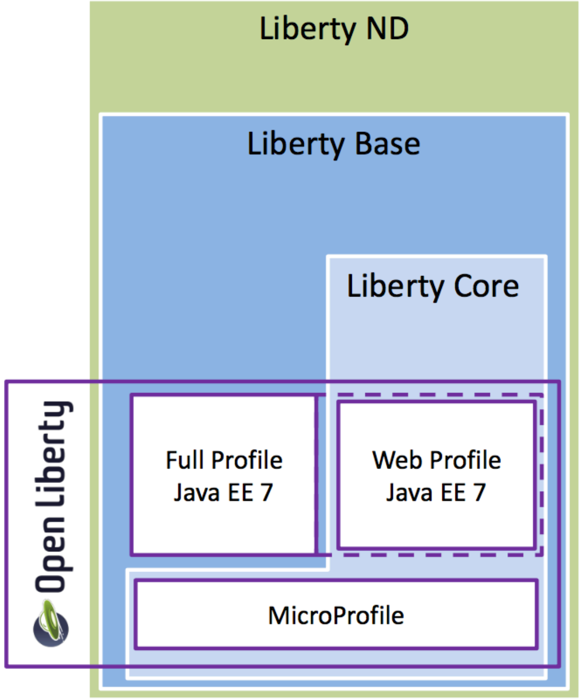
- Easy to consume
- Deploy on any cloud for Java™
- Seamlessly transition to WebSphere

https://openliberty.io/

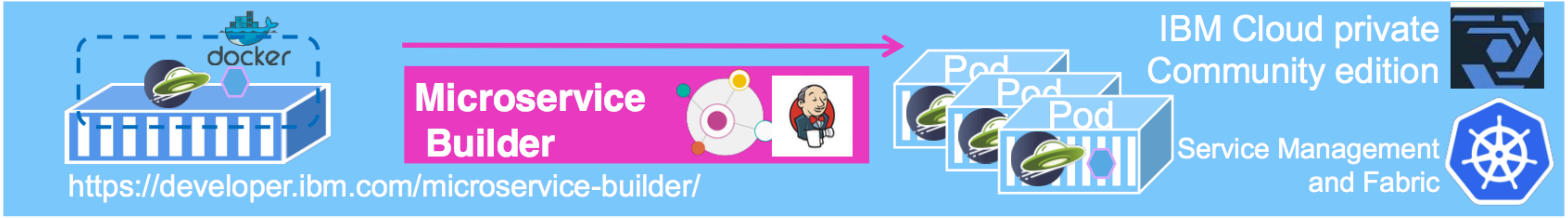


Open Liberty

MicroProfile	WebProfile 7	Java EE 7	Java EE 8 (WIP)	non-API
cdi-1.2	servlet-3.0	jaxws-2.2	servlet-4.0	restConnector-2.0
jsonp-1.0	beanValidation-1.1	jms-2.0	jaxrs-2.1	passwordUtilities-1.0
jaxrs-2.0	cdi-1.2	jca-1.7	jsonb-1.0	federatedRepository-1.0
mpConfig-1.0	jpa-2.1	jaspic-1.1	jsonp-1.1	ldapRegistry-3.0
mpFaultTolerance-1.0	el-3.0	jacc-1.5	javaMail-1.6	monitor-1.0
mpHealth-1.0	jaxrs-2.0	concurrent-1.0	jsf-2.3	bells-1.0
mpMetrics-1.0	managedBeans-1.0	ejb-3.2		appSecurity-2.0
mpJwt-1.0	websocket-1.1	batch-1.0		transportSecurity-1.0
	jsonp-1.0	j2eeManagement-1.1		
	servlet-3.1	javaMail-1.5		
	jsp-2.3			
	jsf-2.2			
	ejbLite-3.2			



An Open Cloud Java Stack



Open Liberty

WebSphere Liberty's open source Java EE and MicroProfile runtime for Java microservices

https://openliberty.io

Java Enterprise Edition

MICROPROFILE

Eclipse OpenJ9

The contribution of IBM's enterprise grade, open source, JVM
Prebuilt OpenJDK 9 with OpenJ9 VM from AdoptOpenJDK

http://www.eclipse.org/openj9/

Java

OpenJ9

Let’s collaborate!

GITTER https://gitter.im/eclipse/microprofile-open-api

github.com/eclipse/microprofile-open-api
github.com/microservices-api
-- github.com/microservices-api/websphere-connect/wiki

webspherereconnect.slack.com

email: **arthurdm@ca.ibm.com**

Labs Intro

Labs

- **Lab 1: OpenAPI Support in WebSphere Liberty**
  – Create a Liberty Server and deploy a JAX-RS application
  – Explore APIs using OpenAPI UI
  – Examine JAX-RS application source including OpenAPI annotations

- **Lab 2: Push APIs to the Cloud**
  – Explore the Liberty Server package
  – Push the server package to IBM Cloud
  – Inspect the APIs in IBM Cloud

Labs

- **Lab 3: OpenAPI Programming Models**
  – Understand OpenAPI Configuration and Builder
  – Programmatically build APIs using OpenAPI Models
  – Implement a custom OpenAPI Scanner

- **Lab 4: A Design-First Approach to Building APIs**
  – Design APIs Using Swagger Editor
  – Include the OpenAPI document within an application
  – Explore the APIs

Bonus Content

- **Bonus Lab: Converting Swagger 2.0 documents to OpenAPI 3.0**
  - Explore SwaggerHub
  - Explore changes from Swagger 2.0 to OpenAPI 3.0

- **Open Liberty Guides**
  - Creating a RESTful web service
  - Creating a MicroProfile application

Getting Started

- The Labs are available on GitHub
  https://github.com/microservices-api/oas3-lab

- **https://libertyswagger.mybluemix.net/api/explorer**
- Local Resources
  – **/home/student/cascon/**
    – eclipse → *(Contains eclipse with WDT plugin)*
    – wlp → *(Liberty Server installation directory)*
    – OpenLiberty → *(Open Liberty Guides)*
    – labs → *(Draft version of labs – as backup)*