

Cloud Native Innovation Labs

# AppMod Cloud Native Demonstration

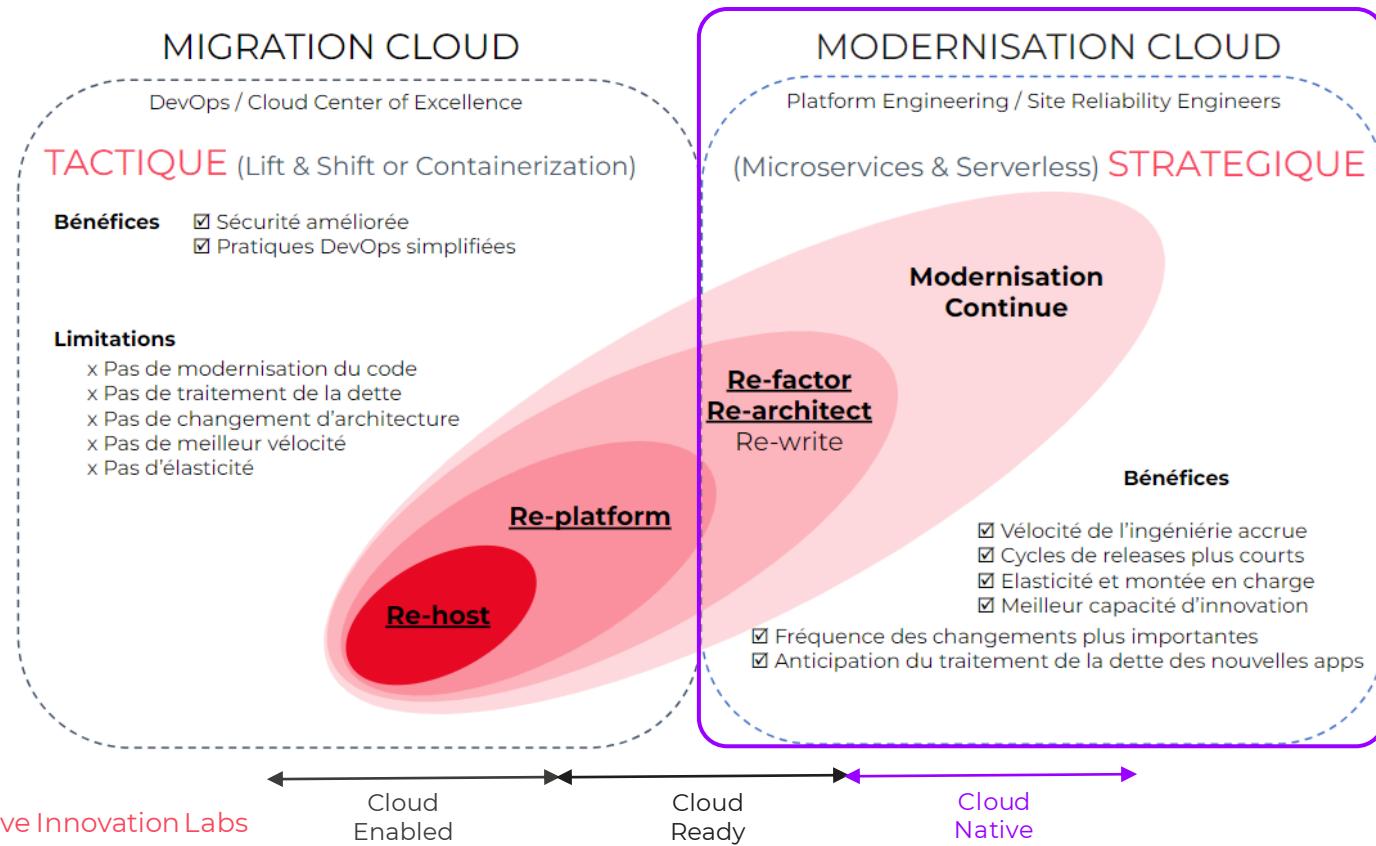
Introduction à la démonstration



Creative tech for Better Change

January 2024

# Moderniser son patrimoine applicatif plutôt que le migrer



# Les 5 types de modernisation possibles

## REHOST - "Big Pod" Lift & Shift

1

Les applications patrimoniales sont déplacées telles quelles, sans réels avantages liés à l'utilisation du Cloud Natif. Les conteneurs sont considérés comme des machines virtuelles ou des "Big Pods"

Une attention particulière est accordée pour garantir une forte cohérence des données lorsque le "Big Pod" échoue. La plupart du temps, chaque Big Pod fonctionne en tant qu'instance unique.

*Modernisation en CLOUD READY*  
*Modernisation en CLOUD NATIVE*

## REARCHITECT - Cloud Native Application

5

L'application est repensée et reconstruite en utilisant des microservices exposant des API via des protocoles communs tels que REST ou gRPC, nécessitant ainsi une couche de gestion des API

Il est probable que la plupart des applications seront en mode hybride jusqu'à une refonte complète, sauf si elles sont reconstruites entièrement. Atteindre ce niveau permettra d'obtenir un niveau élevé de résilience (> 5 nœufs). **Cloud Native Innovation Labs**

1

2

## REHOST - "Light Pod" Lift & Shift

Les applications sont déplacées avec l'ambition de tirer parti des services d'orchestration Cloud Native de base tels que le dimensionnement automatique (vertical ou horizontal).

La migration est réalisée en exploitant des facteurs de forme de conteneur optimisés (légers, idéalement gérés via des opérateurs...).

## REPLATFORM - Cloud Native Lift & Shift

Toutes les déploiements sont gérés en utilisant des pratiques Cloud Native telles que l'approche de déploiement GitOps ou les pratiques d'observabilité.

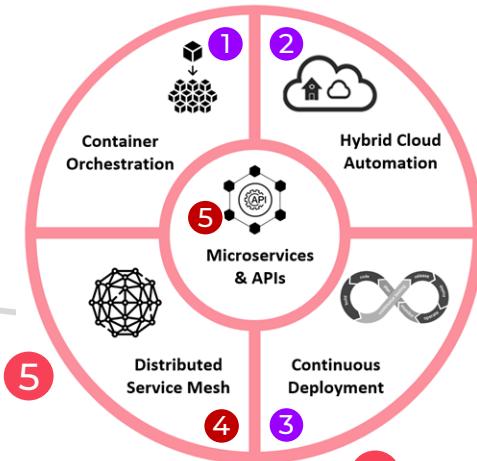
Des services PaaS gérés sont utilisés pour mutualiser les services backend tels que la base de données, le cache, la messagerie, etc.

## REFACTOR - Cloud Native Refactoring

4

De nouvelles capacités sont mises en œuvre pour garantir une approche Zero Trust au sein du déploiement de l'application en utilisant un ServiceMesh tel que Istio.

Cela permet également des approches de déploiement plus progressives telles qu'un déploiement Blue/Green ou A/B, en supposant que la structure de l'application le permette.



# L'application “Fil Rouge” : Order Management (Spring)

The screenshot shows the Postman interface with the following details:

- Collection:** My Workspace
- Request:**
  - Method:** GET
  - URL:** http://localhost:8080/oms-0.0.1-SNAPSHOT/service/inventory/multi-create
  - Body:** JSON (selected)
  - Body Content:**

```

1 [ 
2   {
3     "skuId": "SM-S20-BLK",
4     "storeId": "11",
5     "quantity": 20
6   },
7   {
8     "skuId": "SM-S20-WHT",
9     "storeId": "11",
10    "quantity": 25
11  },
12  {
13    "skuId": "IPHN-12-64-MINI-BLK",
14    "storeId": "11",
15    "quantity": 13
16  },
17 ]

```
- Headers:** (9 items)
- Tests:** Pre-request Script, Tests, Settings

## 3 OMS Application (Spring) 4 5

User Interaction: Headless (REST API)

Framework: Spring 5

# classes: ~ 30

# LoC in classes: 2.6k

AppServer: Apache Tomcat 9

Backend DB

Docker Container

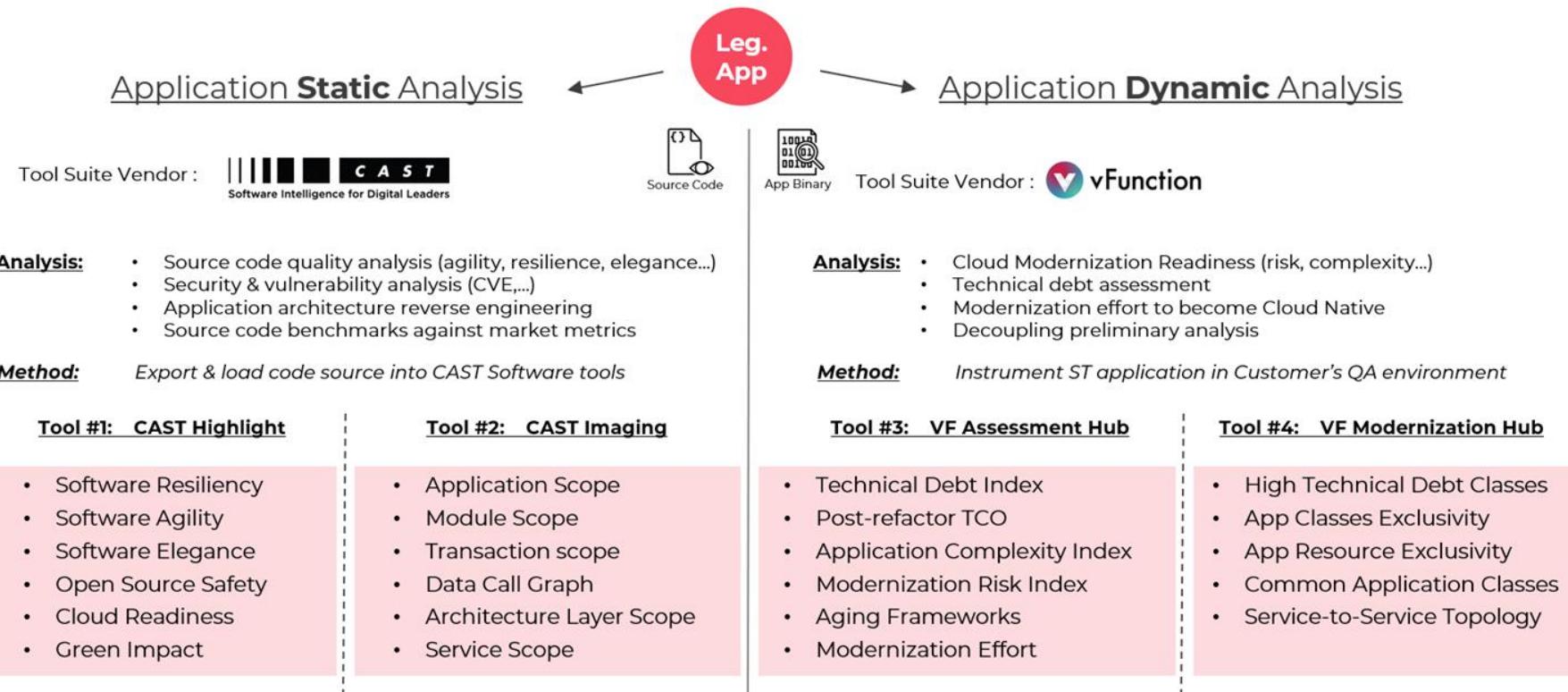
Modernization Type Cloud Native  
# uServices deployments 3

Target Framework Springboot

H2

Kubernetes

# Explication sur l'utilisation des outils de CAST et vFunction



# Aperçu de la plate-forme vFunction

 **Architectural Observability Manager**

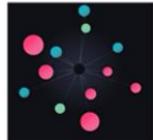
**OBSERVE**  
AI Analysis & Baseling (Detect Drift)

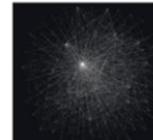




Config AI Parameters      Domain Identification/Baselining      Architectural Drift Thresholds

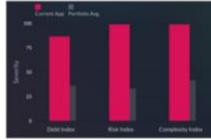
**LEARN**  
Static & Dynamic Analysis (Analyze Architecture)





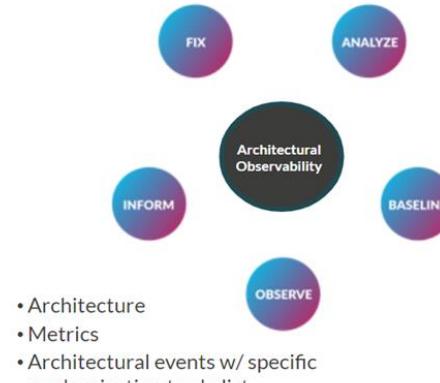
Resource Dependencies      Domain's Call Tree      Static Dependencies

**ANALYZE**  
Static Analysis  
e.g., CI/CD)





Tech Debt, Complexity, Risk      Measure Tech Debt Impact      Multi-App View



 **Refactoring Engine Add-on Module**

**Code Refactoring**



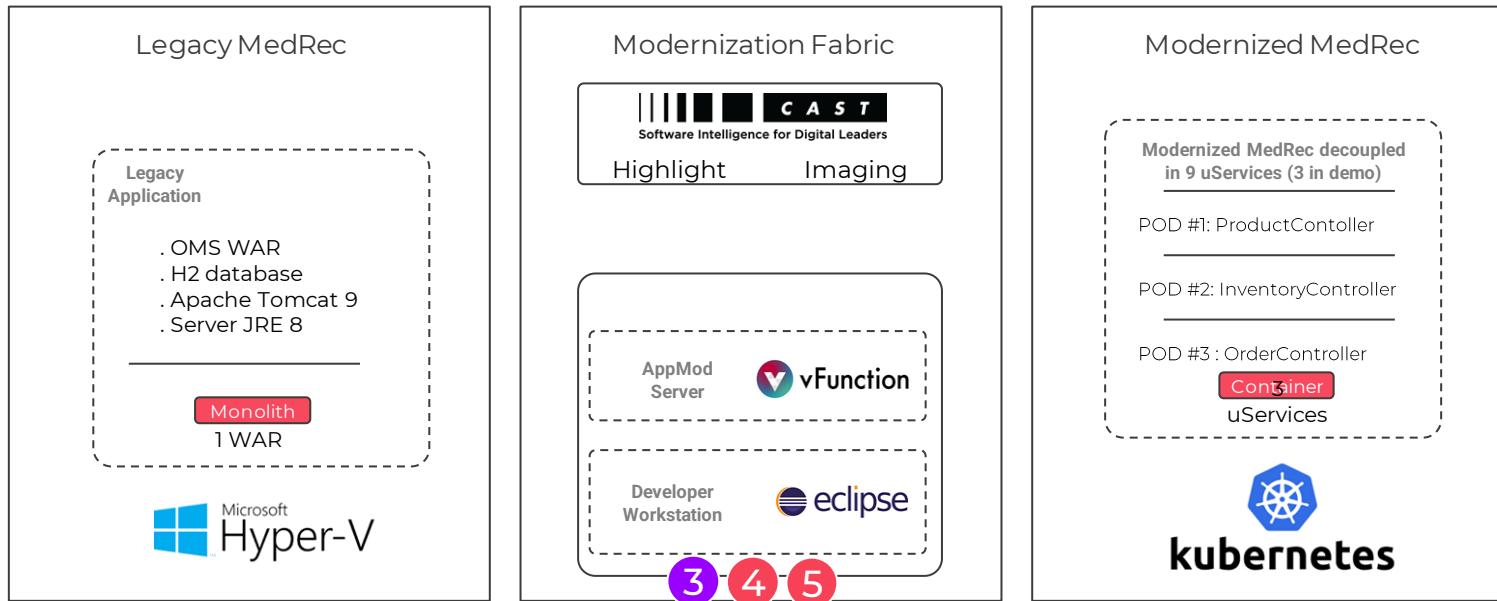
**Service creation:**  
Service Decomposition



- Decomposition automation (microservices code)

- Frameworks upgrade
- Restful APIs generation
- Client libraries generation

# Principes de modernisation de OMS



Cloud Native Innovation Labs

# AppMod Cloud Native Demonstration

Scénarios de démonstration

Creative tech for Better Change



## Files

main

Go to file

- > documentation
- > medrec-legacy
- > medrec-modernized
- oms-legacy**
- > script
- > src
- README.md
- oms-web-layers.drawio
- oms-web-layers.jpg
- pom.xml
- > oms-services
- .gitignore
- LICENSE
- README.md

vFunction / oms-legacy /

Add file

Name	Last commit message	Last commit date
..		
script	first commit	5 hours ago
src	first commit	5 hours ago
README.md	first commit	5 hours ago
oms-web-layers.drawio	first commit	5 hours ago
oms-web-layers.jpg	first commit	5 hours ago
pom.xml	first commit	5 hours ago

README.md



# Order Management System (OMS)

This is an example Spring application used by the tutorial of [vFunction](#).

## Prerequisites

- [Java 8](#) or above
- [Tomcat 9](#) or above
- [Maven 3.6.3](#) or above
- [Postman](#) (optional)

## Compiling

1. After cloning the repository, switch to oms-webmvc folder and compile using maven: `mvn clean install`

## Running (in Tomcat)



### Software Health

Programming best practices that increase resiliency, improve agility and reduce complexity

71.8

WORST

66.3

INDUSTRY

63.37

BEST

77.2



### Cloud Maturity

Software & organizational characteristics to optimize applications on the Cloud

64.1

WORST

59.7

INDUSTRY

62.68

BEST

68.4



### Open Source Safety

Use of 3rd-party components that comply with security, license and age requirements

51.8

WORST

49.6

INDUSTRY

73.03

BEST

54.0



### Green Impact

Programming practices and engineering principles that make software more environment-friendly

83.3

WORST

80.1

INDUSTRY

73.33

BEST

86.6

2/2

ONBOARDED APPLICATIONS

[SEE PORTFOLIO DETAILS](#)

0/0

CURRENTLY NO RUNNING CAMPAIGN

[SEE PORTFOLIO TRENDS](#)

11.0

TOTAL FTE

[SEE DETAILS](#)

17.2k

LINES OF CODE

[SEE DETAILS](#)

## Quick Portfolio Insights

### Top Applications to Invest Resources

Name	Segment
No data available in table	

No data available in table

### Top Applications to Refactor for Cloud

Name	Segment
MEDREC	REFACTOR

MEDREC

REFACTOR

### Top Applications with urgent OSS Risks

Name	Segment
MEDREC	IMMEDIATE ATTENTION

MEDREC

IMMEDIATE ATTENTION

### Top Applications to improve Green Impact

Name	Segment
No data available in table	

No data available in table

[See Software Maintenance](#)

[See Cloud Maturity](#)

[See Open Source Safety](#)

[See Green Impact](#)

## Top 20 Code Insights

## OMS

Dem\_DVT\_Vfunc\_OMS

PPT Report

Ajouter à ma Watchlist

Overview

Sondage

Distribution de la santé

Alertes

Tendances

Cloud Maturity

Green Impact

Benchmark

Composition du programme

Mot clé



Rang 1 / 2



Santé de l'application

77.2



Rang 1 / 2



Résilience Logicielle

85.6



Rang 1 / 2



Agilité du Code

79.9



Rang 1 / 2



Elégance du Code

66.2



Rang 1 / 2



Cloud Maturity

68.4



Rang 2 / 2



Green Impact

80.1



Rang 1 / 2



Santé Open Source

54.0



Rang 2 / 2



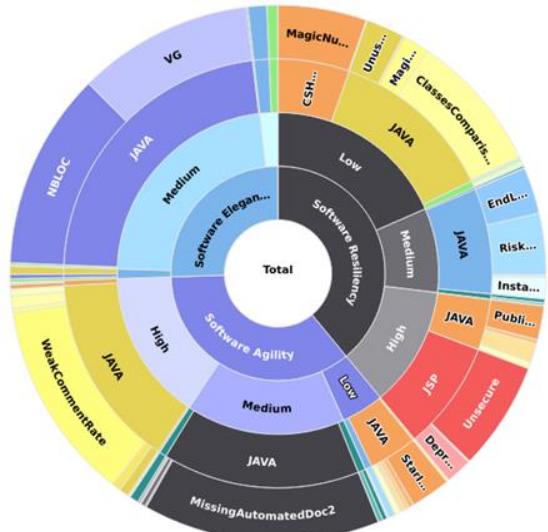
Sécurité des composants

2.0



Distribution

Portfolio Advisor for Technical Debt



Search:

Name	LOC	Files	BI	\$ Tech.Debt	\$ Tech.Debt/LOC	BFP	Total FTE	SR	SA	SE	Campaign Date
MEDREC	14.66k LOC	250	21	18.1 person-day	0.6 min/LOC	2k	8.00 FTE	75.29	70.07	53.53	01/30/24
OMS	2.56k LOC	89	34	4.2 person-day	0.8 min/LOC	48	3.00 FTE	85.60	79.89	66.19	01/14/24

## OMS

Dem\_DVT\_Vfunc\_OMS

PPT Report

Ajouter à ma Watchlist

Overview

Sondage

Distribution de la santé

Alertes

Tendances

Cloud Maturity

Green Impact

Benchmark

Composition du programme

Mot clé

Cloud Migration Advisor

Sondage Cloud

Analyse Cloud

Recommendations

Containerization

Multi-Cloud

Java

Exporter vers Excel

68.4

Cloud Maturity

59.1

Cloud Survey

77.8

Cloud Scan

0.0

Boosters

2.8

Blockers

2

Roadblocks

Exigence Cloud

Technology

Impact

Criticité

Contribution

Roadblocks

Est. Effort ?

Access Control List : Using Access Control List	Java		Critical	+24.92 %	0	0.00 person-day
Application Settings Configuration : Using other configuration files than Web configuration	Java		Low	+0.31 %	0	0.00 person-day
Execution Environment : Access to environment variable	Java		Low	+2.49 %	0	0.00 person-day
Execution Environment : Application server dependencies	Java		Low	+4.67 %	0	0.00 person-day
Execution Environment : Using file system	Java		Medium	+4.98 %	0	0.00 person-day
Execution Environment : Using hardcoded network IP address (IPV4, IPV6)	Java		Low	+2.49 %	0	0.00 person-day
Inter Application Messaging : Using a middleware application	Java		Medium	+4.98 %	0	0.00 person-day
Persistent Files : Perform Directory Manipulation	Java		Medium	+2.49 %	0	0.00 person-day
Persistent Files : Perform File Manipulation	Java		Medium	+2.49 %	0	0.00 person-day
Persistent Files : Using stateful session (Servlet)	Java		High	+4.67 %	0	0.00 person-day



Portfolio Advisor

Cloud Migration Waves

Health Factors

Cloud Requirements

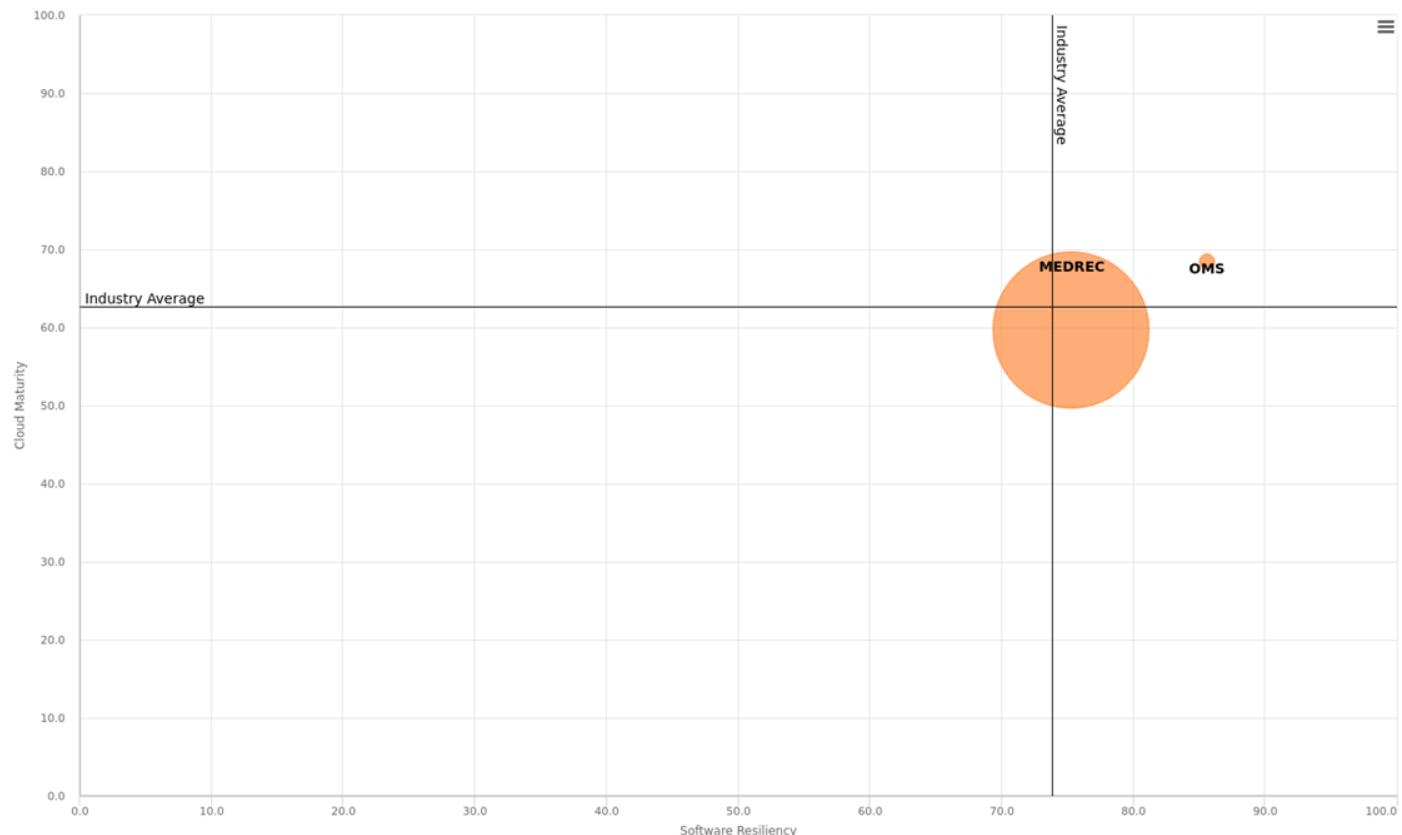
Recommendations

Containerization

Multi-Cloud



Software Resiliency Software Agility Software Elegance





OMS Dem\_DVT\_Vfunc\_OMS

PPT Report

Ajouter à ma Watchlist

Overview

Sondage

Distribution de la santé

Alertes

Tendances

Cloud Maturity

Green Impact

Benchmark

Composition du programme

Mot clé

## Composants Tierce Partie

Chargement du BOM

Explorer les dépendances

CAST consolide l'une des plus grandes bases de données sur les composants logiciels, avec des milliards de signatures de fichiers (empreintes digitales) calculées pour chaque version de composants open source et tiers. Sur la base de cette base de données unique, CAST Highlight compare l'empreinte digitale de vos fichiers d'application et regroupe les informations sur les composants, la version, la licence et la date de sortie au niveau du portefeuille et de l'application.

### Security Vulnerabilities

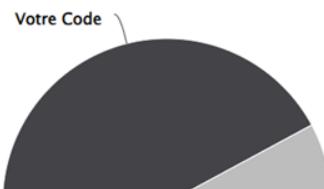
5 CRITICAL SEVERITY

7 HIGH SEVERITY

10 MEDIUM SEVERITY

0 LOW SEVERITY

1 ADVISORIES



### License Compliance

0 HIGH RISK

3 MEDIUM RISK

11 LOW RISK

1 Non défini

Composants Tierce Partie  
14

Plus Ancien  
Aug. 2017

Plus Récent  
Apr. 2022

[Vulnerabilities](#) [Licenses](#) [Obsolescence](#) [All](#)
[Profile de license | CAST Default](#)
[Bas](#) [Moyen](#) [Haut](#) [Non défini](#)
Search:  0/0

**OMS**

Dem\_DVT\_Vfunc\_OMS

PPT Report

Ajouter à ma Watchlist

Overview

Sondage

Distribution de la santé

Alertes

Tendances

Cloud Maturity

Green Impact

Benchmark

Composition du programme

Mot clé

**Benchmark Global****19922 applications**

Last Update: 01/01/2024

CAST Highlight Indicators of your application have been benchmarked against a population of anonymized applications. This population is updated on a monthly basis. For more information about the CAST Highlight Benchmark, please refer to the FAQ.

Sélectionner un axe de benchmark

## Applications filtrées

**19922 applications**

Dans votre portefeuilles

**2 applications****77.2**

Santé de l'application

**85.6**

Résilience Logicielle

**79.9**

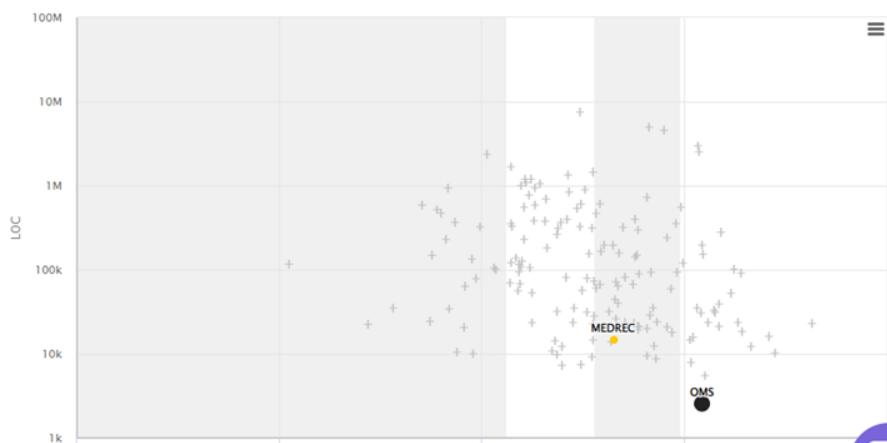
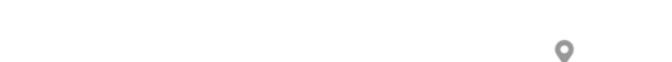
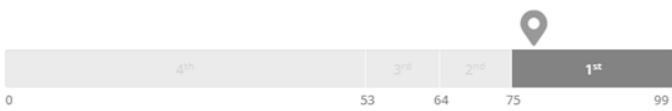
Agilité du Code

**66.2**

Elégance du Code

**68.4**

Cloud Maturity



+ Benchmark sample   Votre portefeuille applicatif   Votre applications benchmarquée



Portfolio Advisor

Cloud Migration Waves

Indicateur de santé

Règle Cloud

Recommendations

Containerization

Multi-Cloud

## Containerization Insights ?

The table below lists Cloud Maturity patterns that have been triggered and could block or slowdown containerization of applications

Règle Cloud	Technology	Impact	Criticité	Roadblocks	Est. Effort	Total app
Persistent Files : Using stateful session (Servlet) ?	Java	C	High	5	0.10 person-day	1
Persistent Files : Perform Directory Manipulation ?	Java	A	Medium	1	0.03 person-day	1
Execution Environment : Using file system ?	Java	A	Medium	18	0.56 person-day	1
Persistent Files : Perform File Manipulation ?	Java	A	Medium	5	0.16 person-day	1
Execution Environment : Using file system ?	C#	A	Medium	18	0.56 person-day	1
Execution Environment : Using file system ?	Python	A	Medium	1	0.03 person-day	1
Security & User Authentication : Use of unsecured network protocols (HTTP, FTP) ?	C#	C	Low	1	0.02 person-day	1
Security & User Authentication : Use of unsecured network protocols (HTTP, FTP) ?	Java	C	Low	1	0.02 person-day	1

# Cloud Migration Wave Advisor

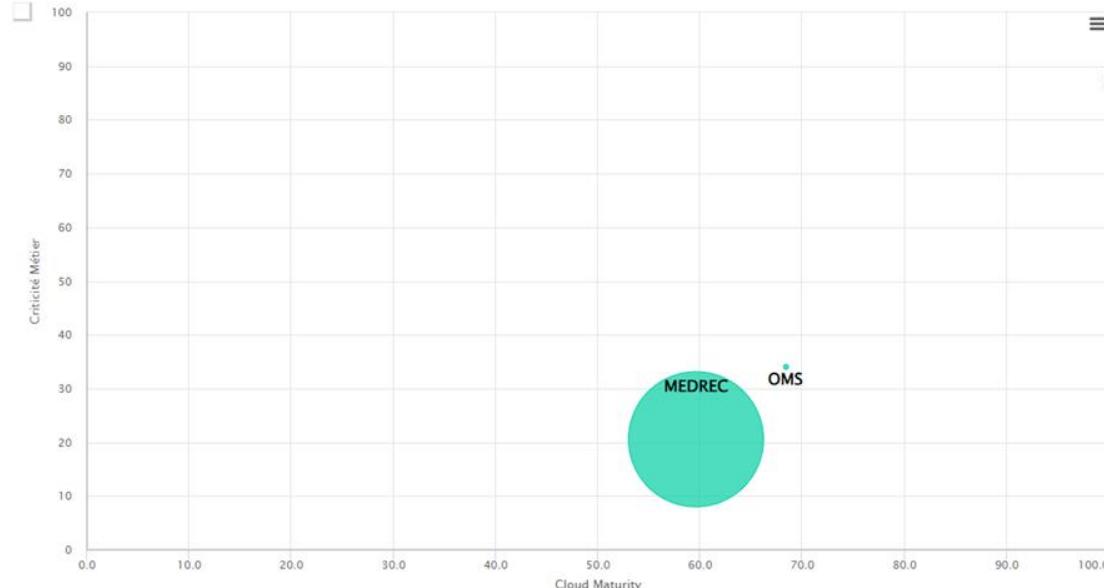
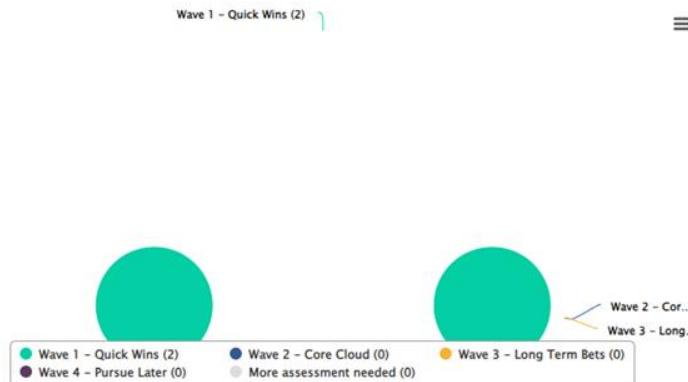
Export

This dashboard recommends the sequence of applications to migrate to the cloud for optimal results. Recommendations are based on various insights such as the Business Impact of applications, Cloud Maturity score, estimated effort, and segment (rehost, refactor, rearchitect, etc.) as well as detected technologies. Note that for optimal insights, the Business Impact survey should be completed.

↔ Criticité Métier ▾

↔ Cloud Maturity ▾

○ Cloud Effort ▾



2

**Wave 1 - Quick Wins**

0

**Wave 2 - Core Cloud**

0

**Wave 3 - Long Term Bets**

0

**Wave 4 - Pursue Later**

0

**More assessment needed**

# Flux de démonstration

1. Step 1: Montrer les 3 cas d'usage de création et de lecture des produits, Inventaires et Commandes via les API Rest qui seront ensuite réalisées via les microservices
2. Step 2: Présenter l'interface graphique de vFunction en utilisant leur instance de démo car + riche
3. Step 3: Basculer sur l'instance locale de vFunction et présenter OMS avant discovery + static analysis
4. Step 4: Lancer une découverte et exécuter un flux de 5 appels API Rest sur les 3 services ci-dessus
5. Step 5: Charger la découverte complète et présenter les bubbles avec focus sur la partie gestion des Inventaire, Produits et Commandes
6. Step 6: Charger le SNAP-002 et préciser que l'objectif est de passer de la vue éclatée à la vue consolidée (préciser les compétences nécessaires + le jus de cerveau nécessaire) + extraction des codes sources des 3 microservices avec la target Spring
7. Step 7: On bascule sur le studio Eclipse + explication de comment c'est obtenu + montrer rapidement les fichiers modifiés pour le microservice ProductController
8. Step 8: On termine en présentant la création des containers et en montrant le test des APIs REST
9. Step 9: OpenMind, expliquer comment cela aide à simplifier la modernisation, les nouvelles features, le cycle de release, les opérations ...



# Thank you.



Cloud Native Innotation Labs

# Step 1

Montrer les 3 cas d'usage de création et de lecture des produits,  
Inventaires et Commandes via les API Rest qui seront ensuite réalisées via les microservices

Activities Terminal janv. 29 21:15

```
franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-webmvc/script$ ./createInventories.sh
Creating multiple inventories
[{"skuId": "SM-S20-BLK", "storeId": "11", "quantity": 20}, {"skuId": "SM-S20-WHT", "storeId": "11", "quantity": 25}, {"skuId": "IPHN-12-64-MINI-BLK", "storeId": "11", "quantity": 13}, {"skuId": "IPHN-12-64-MINI-WHT", "storeId": "11", "quantity": 23}, {"skuId": "IPHN-12-64-MINI-GRN", "storeId": "11", "quantity": 12}]franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-webmvc/script$ ./createProducts.sh
Creating multiple products
[{"productId": "SM-S20-BLK", "name": "S20 Black", "description": "Samsung S20 Black", "manuf": "Samsung"}, {"productId": "SM-S20-WHT", "name": "S20 White", "description": "Samsung S20 White", "manuf": "Samsung"}, {"productId": "IPHN-12-64-MINI-BLK", "name": "iPhone 12 Mini 1", "description": "iPhone 12 Mini 64GB Black", "manuf": "Apple"}, {"productId": "IPHN-12-64-MINI-WHT", "name": "iPhone 12 Mini 2", "description": "iPhone 12 Mini 64GB White", "manuf": "Apple"}, {"productId": "IPHN-12-64-MINI-GRN", "name": "iPhone 12 Mini 3", "description": "iPhone 12 Mini 64GB Green", "manuf": "Apple"}]franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-webmvc/script$ ./createOrders.sh
Creating multiple orders
[{"customerOrderId": "1", "primaryPhone": "9525944805", "customerEmailId": "as@yahoo.com", "orderStatus": "SUBMITTED", "firstName": "Anand", "orderDate": null, "profileId": null, "lastName": null, "entryType": null, "billToAddressId": "101", "billToAddress": {"billToAddressId": "101", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "shipToAddress": {"shipToAddressId": "102", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "orderLines": [{"lineItemId": "100", "customerOrderId": "1", "shipNode": null, "shipNodeDescription": null, "levelOfService": null, "primeLineNumber": "1", "subLineNumber": "1", "customerSKU": "SM-S20-BLK", "skuDescription": null, "estimatedArrivalDate": null, "status": null, "reshippedBefore": null, "shipToAddress": {"billToAddressId": "102", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "cardType": "VISA", "authorizedAmount": 43.0, "collectedAmount": 5.0, "charges": {"chargeId": "601", "lineSubTotal": null, "totalCharges": 10.0}], "paymentInfo": {"paymentId": "234", "paymentStatus": "SUCCESS", "cardType": "VISA", "authorizedAmount": 43.0, "collectedAmount": 5.0, "charges": {"chargeId": "601", "lineSubTotal": null, "totalCharges": 10.0, "salesTax": null, "grandTotal": null}}, {"customerOrderId": "2", "primaryPhone": "9525944805", "customerEmailId": "as@yahoo.com", "orderStatus": "SUBMITTED", "firstName": "Anand", "orderDate": null, "profileId": null, "lastName": null, "entryType": null, "billToAddress": {"billToAddressId": "101", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "shipToAddress": {"shipToAddressId": "102", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "orderLines": [{"lineItemId": "200", "customerOrderId": "2", "shipNode": null, "shipNodeDescription": null, "levelOfService": null, "primeLineNumber": "1", "subLineNumber": "1", "customerSKU": "SM-S20-WHT", "skuDescription": null, "estimatedArrivalDate": null, "status": null, "reshippedBefore": null, "shipToAddress": {"billToAddressId": "102", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "cardType": "VISA", "authorizedAmount": 43.0, "collectedAmount": 5.0, "charges": {"chargeId": "602", "lineSubTotal": null, "totalCharges": 11.0, "salesTax": null, "grandTotal": null}}, {"customerOrderId": "3", "primaryPhone": "9525944805", "customerEmailId": "as@yahoo.com", "orderStatus": "SUBMITTED", "firstName": "Anand", "orderDate": null, "profileId": null, "lastName": null, "entryType": null, "billToAddress": {"billToAddressId": "101", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "shipToAddress": {"shipToAddressId": "102", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "orderLines": [{"lineItemId": "300", "customerOrderId": "3", "shipNode": null, "shipNodeDescription": null, "levelOfService": null, "primeLineNumber": "1", "subLineNumber": "1", "customerSKU": "IPHN-12-64-MINI-BLK", "skuDescription": null, "estimatedArrivalDate": null, "status": null, "reshippedBefore": null, "shipToAddress": {"billToAddressId": "102", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "cardType": "VISA", "authorizedAmount": 43.0, "collectedAmount": 5.0, "charges": {"chargeId": "603", "lineSubTotal": null, "totalCharges": 12.0, "salesTax": null, "grandTotal": null}]}]franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-webmvc/script$ franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-webmvc/script$ franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-webmvc/script$ franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-webmvc/script$
```

### Login

Saved Settings: Generic H2 (Embedded)

Setting Name: Generic H2 (Embedded)

---

Driver Class: org.h2.Driver

JDBC URL: jdbc:h2:mem:testdb

User Name: sa

Password:

| | |  Auto commit | | Max rows: 1000  |  | Auto complete Off  | Auto select On

jdbc:h2:mem:testdb
+  BILL_TO_ADDRESS
+  CHARGES
+  INVENTORY
+  LINE_CHARGE
+  ORDER_LINE
+  PAYMENT_INFO
+  PRODUCT_INFO
+  SALES_ORDER
+  SHIPPING
+  SHIP_TO_ADDRESS
+  INFORMATION_SCHEMA
+  Sequences
+  Users
(i) H2 1.4.200 (2019-10-14)

SQL statement:

```
SELECT * FROM INVENTORY
```

SELECT \* FROM INVENTORY;

SKU_ID	STORE_ID	QTY
13778	10	20
SM-S20-BLK	11	20
SM-S20-WHT	11	25
IPHН-12-64-MINI-BLK	11	13
IPHН-12-64-MINI-WHT	11	23
IPHН-12-64-MINI-GRN	11	12

(6 rows, 4 ms)

Cloud Native Innotation Labs

# Step 2

Présenter l'interface graphique de  
vFunction en utilisant leur instance  
de démo car + riche



4 Total Apps | Jan 22, 2024

Default View

Sort By: App Name ▾

Debt Level (1 - 100)

1 100

Accumulation of Debt:

- 0 +

App Size:

XS S M L XL

App Type:

Java .Net

Last Updated: ▾

Created: ▾

Has Analysis 

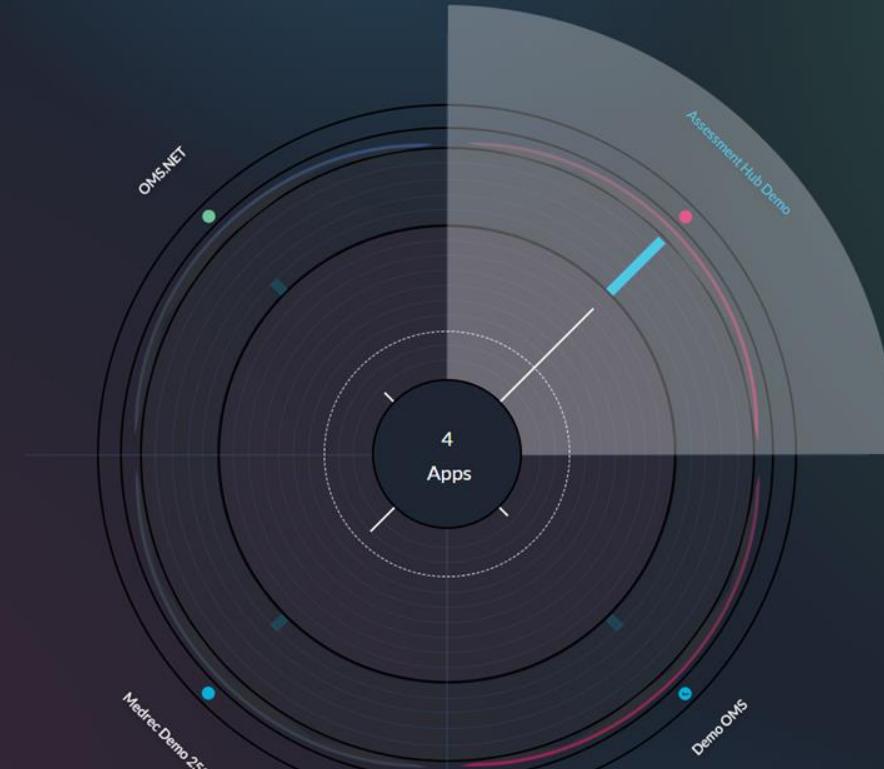
Architectural Complexity

1 2 3 4 5

 (Denotes Modernized)

# Application Tags ▾

Download Filtered Report



## Assessment Hub Demo

App Last updated: 09/01/2024

View Tech-Debt Report 

Debt	(Avg. All)
86.54%	32%
Accumulation of Debt	
Neutral	

Size of App
18359 classes
XLarge

Aging Framework
82%

Architectural Complexity
Very high

# Application Tags (0) 

TODOs	
0	
Manual TODOs	0 (0 Done)
Drift TODOs	0 (0 Done)

Hide Legend ▾

Debt Size (1-100)
Color denotes accumulation
- 0 +

App Size

XS S M L XL

4 Total Apps Jan 22, 2024

## Onboarding Status

INSTALLATION	LEARNING	ANALYSIS	OBSERVATION
0 Apps	1 App	2 Apps	1 App
0 Static Data Available	0 Actively Learning	0 Active Analysis	4 Schedule Runs: 0 Dynamic Class Exclusivity Alerts 1 Domain Changes Alert 0 Dead Code Alerts 0 High Debt Alerts 0 Common Library Alerts 1 Resource Exclusivity Alert 1 Architecture Complexity Alert
0 Controllers Installed	1 Analysis Ready	2 Baseline Ready	1 Missed Scope 0 Static Class Exclusivity Alerts 0 Library Circular Dependencies 1 Missed Trigger
0 Learning Ready			

## SERVICE CREATION

3 Apps  
8 Services Created

Licensed Classes	18,612
Learning Time	47h
Analysis Events	7,421



# Onboarding

**INSTALLATION**  
0 Apps**LEARNING**  
1 App**ANALYSIS**  
2 Apps**OBSERVATION**  
1 App in Observation Phase**SERVICE CREATION**  
3 AppsFavorites Installation **0**Learning **1**Analysis **2**Observation **1**Search Architecture Complexity:  1 2 3 4 5 **INSTALLATION**

Static Data Available (0)

Controllers Installed (0)

Learning Ready (0)

All (0)

No applications in this state

**LEARNING**

Actively Learning (0)

Analysis Ready (1)

All (1)

STOP ALL MEASUREMENTS

Sort By: Most Recent  Assessment Hub D...

Stopped

View Tech-Debt Report **ANALYSIS**

Active Analysis (0)

Baseline Ready (2)

All (2)

Architectural Complexity Sort By: Complexity High to Low  Medrec Demo 259 

Analysis Start- PB

7 Domains

99.97% Class Exclusivity

 OMS.NET 

Jeroen Domensino - 8 Dec 202...

8 Domains

86.31% Class Exclusivity



^ Filters Show All

Default View

Graph View

^ Date None

Week Month QTD

^ Status Show All None

TODO DONE Snoozed

Dismissed Acknowledged

^ Type Show All None

Drift Manual

^ Event Show All None

Architecture Complexity

Common Library Dead Code

Domain Changes

Dynamic Class Exclusivity

High Debt Class

Library Circular Dependencies

Resource Exclusivity

Static Class Exclusivity

^ Application Show All None

Q SEARCH

Assessment Hub Demo

Demo OMS Medrec Demo 259

OMS.NET

## TODOS - 25 of 59 (4 Filters Applied)

GROUP BY: Application ▾



Export CSV



		Date	Event	Details	Domains
▼	● D	10-Sep-2023 12:55	Domain Changes	InventoryController	
▼	● D	10-Sep-2023 12:55	Domain Changes	OrderController (+1)	
▼	● D	10-Sep-2023 12:55	Domain Changes	StoreSearchController	
▼	● D	10-Sep-2023 12:55	Resource Exclusivity	oms-log.txt (File)	InventoryController, OrderController (+1), ShippingPriceController and 1 more.
▼	● D	27-Aug-2023 17:15	Architectural Complexity Decreased	Architectural complexity has decreased from 3 to 1	
▼	● D	25-Aug-2023 20:55	Architectural Complexity Decreased	Architectural complexity has decreased from 3 to 2	
▼	● D	26-Sep-2023 16:40	Domain Changes	ModifyFulfillmentController	
▼	● D	26-Sep-2023 16:40	Resource Exclusivity	SALES_ORDER (Read-Write)	Application, ModifyFulfillmentController
▼	● D	26-Sep-2023 16:40	Resource Exclusivity	SHIP_TO_ADDRESS (Read-Only)	Application, ModifyFulfillmentController
▼	● D	26-Sep-2023 16:40	Resource Exclusivity	LINE_CHARGE (Read-Only)	Application, ModifyFulfillmentController
▼	● D	26-Sep-2023 16:40	Resource Exclusivity	PAYMENT_INFO (Read-Only)	Application, ModifyFulfillmentController
▼	● D	26-Sep-2023 16:40	Resource Exclusivity	BILL_TO_ADDRESS (Read-Only)	Application, ModifyFulfillmentController
▼	● D	26-Sep-2023 16:40	Resource Exclusivity	CHARGES (Read-Only)	Application, ModifyFulfillmentController
▼	● D	26-Sep-2023 16:40	Resource Exclusivity	ORDER_LINE (Read-Write)	Application, ModifyFulfillmentController
▼	● D	27-Aug-2023 18:05	Architectural Complexity Increased	Architectural complexity has increased from 1 to 2	

## ^ Medrec Demo 259 Baseline measurement Analysis Completed

TODO 4

DONE 0

		Date	Event	Details	Domains
▼	● M	26-Oct-2023 08:55	Dynamic Class Exclusivity	PatientServiceImpl	Authentication, PatientSearch, RecordsManagement and 1 more.
▼	● M	18-Oct-2023 14:43	Dynamic Class Exclusivity	PatientService	Authentication, Monolith, RecordsManagement and 1 more.
▼	● D	3-Oct-2023 06:41	High Debt Class	JmsClientImpl	
▼	● D	3-Oct-2023 06:41	High Debt Class	PatientServiceImpl	

## ^ OMS.NET Baseline measurement Cori - 19 Jun 2023 12:56

TODO 1

DONE 3

		Date	Event	Details	Domains
▼	● M	12-Oct-2023 14:09	Common Library	LineCharges	
▼	● M	15-Dec-2023 13:43	Dynamic Class Exclusivity	InventoryService	Found in 0 Domains
▼	● M	12-Oct-2023 14:11	Resource Exclusivity	dbo.OrderLines (Read-Write)	Found in 0 Domains
▼	● M	12-Oct-2023 14:06	Static Class Exclusivity	ShippingService	Found in 0 Domains

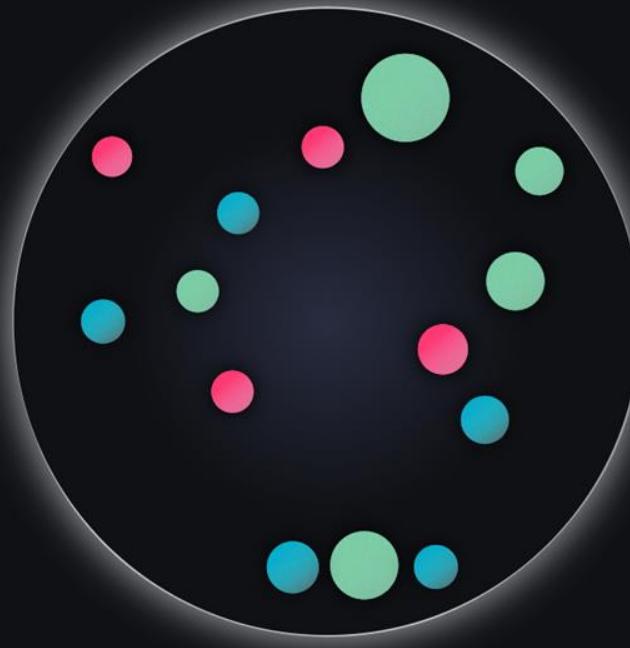


## Learning

For the system to learn and understand your application, it needs to analyze as many usage patterns as possible. It is recommended to either collect data while the application is running in production for a while or, if it's installed in pre-production, test the major flows of the application. It is recommended to run your regular set of acceptance tests or regression tests in order to get the best coverage.

### ACTIONS

- [View Tech-Debt Report](#)
- [Select Controllers](#)



START

^  
MEASUREMENTS

| DOMAINS  
14

| FUNCTIONS 18,580 | RESOURCES 48,479

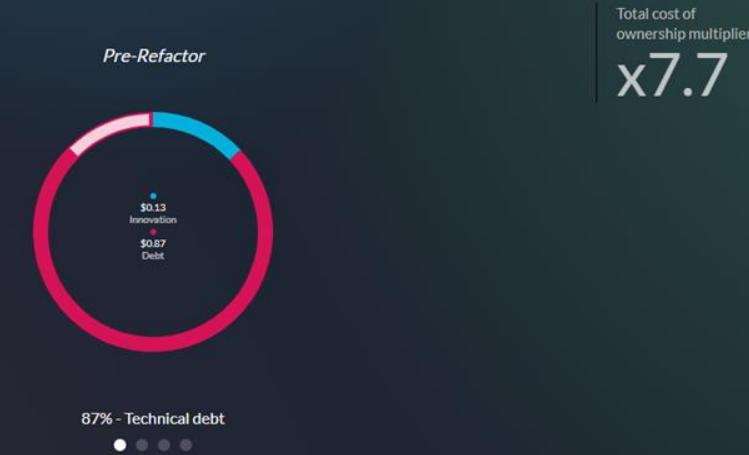
## Tech-Debt Report

[Cost of Innovation](#) [Tech Debt Metrics](#) [Aging Frameworks](#) [History](#)

The Tech-Debt Report shows the level of technical debt and architectural complexity in your application. Through static analysis, the report exposes how the dependencies between the classes affect the modularity of the code, the risk of impacting parts of the application and the overall level of technical debt. This level of debt affects the rate and cost of innovation and is a key business measure for the application.

This chart shows how the level of technical debt affects the cost of innovation for the application. It tries to assess if one would spend \$1 on this application, how much of it will actually be put to innovation, and how much to repay the technical debt within the application.

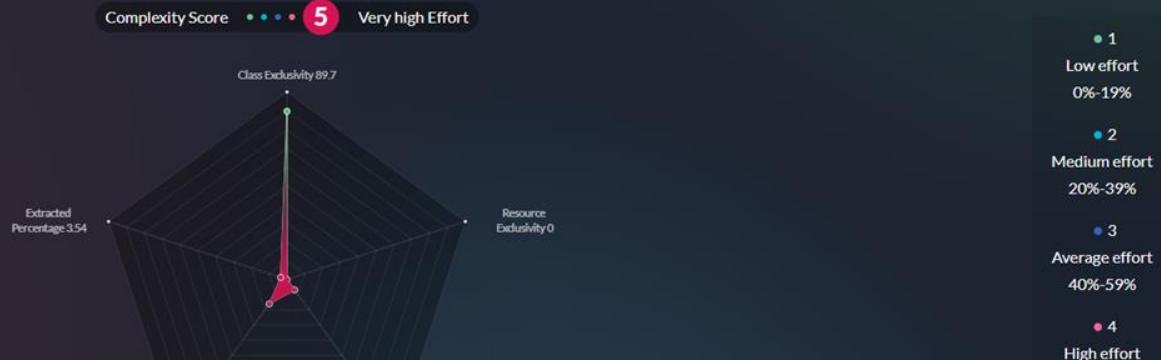
#Application Tags (0)



## Architectural Complexity

The radar chart provides a visual representation of the key architectural aspects of your application. While these parameters do not indicate the complexity of the application, they do indicate the level of effort that is required to re-architect the application. A larger covered area signifies lower effort to re-architect the application.

This calculation is based on a specific measurement, and depends on the analysis that was done on it.



## Tech-Debt Report

[Cost of Innovation](#) **Tech Debt Metrics** [Aging Frameworks](#) [History](#)

The Tech-Debt Report shows the level of technical debt and architectural complexity in your application. Through static analysis, the report exposes how the dependencies between the classes affect the modularity of the code, the risk of impacting parts of the application and the overall level of technical debt. This level of debt affects the rate and cost of innovation and is a key business measure for the application.

Complexity is the degree to which class dependencies are entangled between themselves, reducing the level of modularity of the code. Risk is correlated to the length of the dependencies. How likely is a change in the application to affect a seemingly unrelated part of it. The debt is an overall calculation, taking into account complexity, risk and other factors to provide a single metric to prioritize the application for modernization.

The average of these three metrics in the portfolio of applications is presented to further help with the prioritization.

#Application Tags (0)



### Debt Index

Score	86.54
Portfolio Avg	32
Difference	+168%

### Risk Index

Score	99
Portfolio Avg	32
Difference	+207%

### Complexity Index

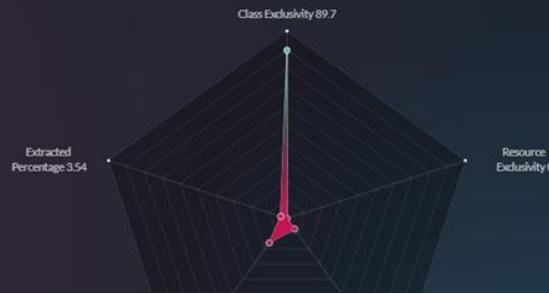
Score	99
Portfolio Avg	31
Difference	+217%

## Architectural Complexity

The radar chart provides a visual representation of the key architectural aspects of your application. While these parameters do not indicate the complexity of the application, they do indicate the level of effort that is required to re-architect the application. A larger covered area signifies lower effort to re-architect the application.

This calculation is based on a specific measurement, and depends on the analysis that was done on it.

Complexity Score 5 Very high Effort



- 1 Low effort 0%-19%
- 2 Medium effort 20%-39%
- 3 Average effort 40%-59%
- 4 High effort

## Tech-Debt Report

An important part of technical debt is the aging frameworks that are used in the application. Aging frameworks are a source of security risk, and indicate an increased effort to upgrade the frameworks to their latest versions. The use of some aging frameworks will also block the application from being considered cloud-native.

This table shows the names of the jars of the frameworks that are used in the application.

Each framework, based on its current and newest version receives a tag of either modern, aging, or unknown, in case the specific jar was not found in the vFunction database.

It also shows the total number of classes that were scanned by the static analysis, and the minimal Java version that these classes were compiled with.

#Application Tags (0)

Aging Frameworks Threshold 0%			
Framework Validity • Modern • Unknown • Aging			
Name	Current Version	Last Version	Jar
Apache Ant Core	1.7.1	1.10.14	• ant-1.7.1.jar *
ANTLR 3 Tool	2.7.7	3.5.3	• antlr-2.7.7.jar *
ANTLR 4 Runtime	4.9.1	4.13.1	• antlr4-runtime-4.9.1.jar
Avalon Framework API	4.2.0	4.3.1	• avalon-framework-api-4.2...
Avalon Framework Imple...	4.2.0	4.3.1	• avalon-framework-impl-4...
Barcode4J	2.1	2.2.1	• barcode4j-2.1.jar
Bouncy Castle Provider	1.68	1.70.GM	• bcprov-jdk15on-1.68.jar *
bl	1.0.11-SNAPSHOT	5.1.0	• bl-1.0.11-SNAPSHOT.jar
bl	1.0.29-SNAPSHOT	5.1.0	• bl-1.0.29-SNAPSHOT.jar
marker-and	4.4.4	4.7.1	• marker-and-4.4.4.jar *



## Architectural Complexity

The radar chart provides a visual representation of the key architectural aspects of your application. While these parameters do not indicate the complexity of the application, they do indicate the level of effort that is required to re-architect the application. A larger covered area signifies lower effort to re-architect the application.

This calculation is based on a specific measurement, and depends on the analysis that was done on it.



- 1 Low effort 0%-19%
- 2 Medium effort 20%-39%
- 3 Average effort 40%-59%

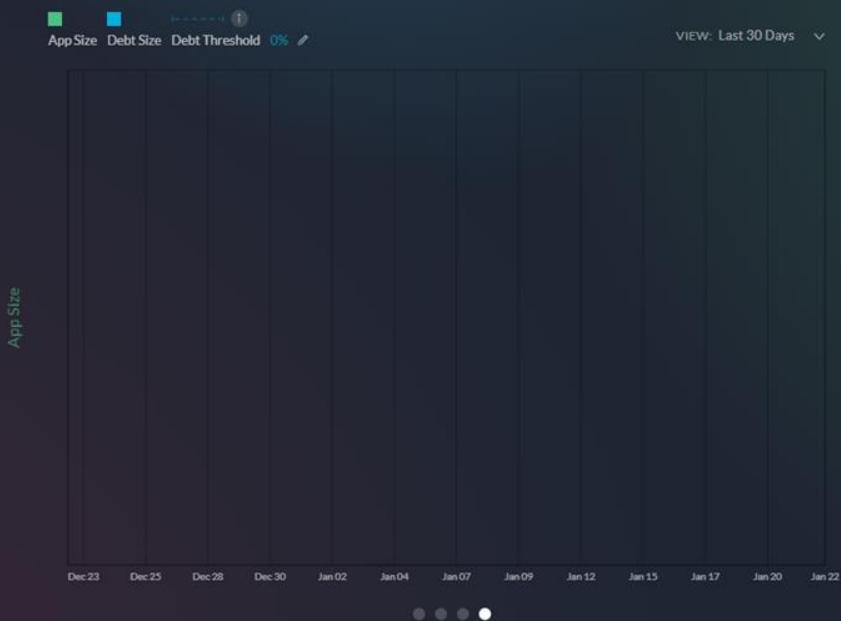
## Tech-Debt Report

The size of the application and its technical debt are two key factors to consider when it comes to observability. The size of an application is related to the amount of code and resources that make up the application. A larger size can make it more difficult and costly to maintain and update. Technical debt, on the other hand, refers to the cost of maintaining and updating an application over time due to shortcuts or shortcuts taken during the development process. It can also refer to the cost of fixing problems or bugs that arise as a result of these shortcuts. A higher technical debt indicates that the application is more challenging to modernize.

The graph shows how the size and technical debt of the application changes over time. This information helps to understand the current state of the application and the potential challenges.

To help manage technical debt and ensure that it does not become too high, set a technical debt threshold for the application. A technical debt threshold is a specific level of technical debt that has been established as acceptable. The history graph's technical debt threshold tracks and monitors technical debt, and sends notifications when the technical debt of the application exceeds it. Staying aware of the technical debt of an application allows proactive measures to be taken to address it before it becomes a significant problem.

#Application Tags (0)



## Architectural Complexity

The radar chart provides a visual representation of the key architectural aspects of your application. While these parameters do not indicate the complexity of the application, they do indicate the level of effort that is required to re-architect the application. A larger covered area signifies lower effort to re-architect the application.

Complexity Score 5 Very high Effort

Class Exclusivity 89.7

• 1	Low effort
0%-19%	
• 2	Medium effort
20%-39%	

Cloud Native Innotation Labs

# Step 3

Basculer sur l'instance locale de  
vFunction et présenter OMS avant  
discovery + static analysis





## Assess & Learn

Name: oms-java-local  
Status: Controller is down  
Version: 3.3.1251  
CPU Utilization: 16%  
ID: a5681e08-7be0-4d40-a40d-6ac2646927d8  
Created Date: 04 Jan 2024 11:42:45

IP

WARNING

DOWN

Filter By Created Date: Ascending ▾

Select a controller to either  
initiate the learning process or  
generate an tech-debt Report.



DELETE

The tech-debt report evaluates  
the level of technical debt  
present in the application,  
providing crucial insights that  
inform your continuous  
modernization strategy.

The learning process allows  
vFunction to uncover the  
intricacies of your application  
while it's in operation. This step  
is essential for recognizing  
distinct domain and domains  
within the application. This  
exploration contributes to  
architectural observability,

### ACTIONS

- Install Controller
- Configure Packages
- Table View

### CONFIGURATIONS

- Batch measurement
- Refresh viper

SELECT

VIEW TECH-DEBT REPORT

### MEASUREMENTS

Hide Legend ▾

Controller Up

Attention Required

Controller Down

## Tech-Debt Report

[Cost of Innovation](#)[Tech Debt Metrics](#)[Aging Frameworks](#)[History](#)

The Tech-Debt Report shows the level of technical debt and architectural complexity in your application. Through static analysis, the report exposes how the dependencies between the classes affect the modularity of the code, the risk of impacting parts of the application and the overall level of technical debt. This level of debt affects the rate and cost of innovation and is a key business measure for the application.

This chart shows how the level of technical debt affects the cost of innovation for the application. It tries to assess if one would spend \$1 on this application, how much of it will actually be put to innovation, and how much to repay the technical debt within the application.

#Application Tags (0)



Total cost of ownership multiplier

**x1.1**

9% - Technical debt



## Tech-Debt Report

Cost of Innovation    Tech Debt Metrics    Aging Frameworks    History

The Tech-Debt Report shows the level of technical debt and architectural complexity in your application. Through static analysis, the report exposes how the dependencies between the classes affect the modularity of the code, the risk of impacting parts of the application and the overall level of technical debt. This level of debt affects the rate and cost of innovation and is a key business measure for the application.

Complexity is the degree to which class dependencies are entangled between themselves, reducing the level of modularity of the code. Risk is correlated to the length of the dependencies. How likely is a change in the application to affect a seemingly unrelated part of it. The debt is an overall calculation, taking into account complexity, risk and other factors to provide a single metric to prioritize the application for modernization.

The average of these three metrics in the portfolio of applications is presented to further help with the prioritization.

#Application Tags (0)



### Debt Index

Score	9.45
Portfolio Avg	16
Difference	-41%

### Risk Index

Score	7.88
Portfolio Avg	10
Difference	-20%

### Complexity Index

Score	1
Portfolio Avg	13
Difference	-92%

## Tech-Debt Report

[Cost of Innovation](#)[Tech Debt Metrics](#)[Aging Frameworks](#)[History](#)

An important part of technical debt is the aging frameworks that are used in the application. Aging frameworks are a source of security risk, and indicate an increased effort to upgrade the frameworks to their latest versions. The use of some aging frameworks will also block the application from being considered cloud-native.

This table shows the names of the jars of the frameworks that are used in the application.

Each framework, based on its current and newest version receives a tag of either modern, aging, or unknown, in case the specific jar was not found in the vFunction database.

It also shows the total number of classes that were scanned by the static analysis, and the minimal Java version that these classes were compiled with.

#Application Tags (0)

Aging Frameworks Threshold 0%

Framework Validity • Modern • Unknown • Aging

Name	Current Version	Last Version	Jar

• • • •

0%

Modern Framework

0%

Unknown Framework

0%

Aging Framework

Number of Classes

55

Min Compile Version

1.8

Cloud Native Innotation Labs

# Step 4

Lancer une découverte et exécuter un flux de 5 appels API Rest sur les 3 services ci-dessus

Activities Postman janv. 29 21:16 Create Inventory entries - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import No Environment

Collections Environments History

OMS WebMVC / Inventory / Create Inventory entries

GET http://localhost:38080/oms-0.0.1-SNAPSHOT/service/inventory/multi-create

Params Authorization Headers (9) Body Tests Settings Cookies Beautify

Body (none) form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 [  
2   {  
3     "skuId": "SM-S20-BLK",  
4     "storeId": "11",  
5     "quantity": 20  
6   },  
7   {  
8     "skuId": "SM-S20-WHT",  
9     "storeId": "11",  
10    "quantity": 25  
11   },  
12 ]
```

Status: 200 OK Time: 45 ms Size: 452 B Save as example

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1 [{"skuId": "SM-S20-BLK", "storeId": "11", "quantity": 20}, {"skuId": "SM-S20-WHT", "storeId": "11", "quantity": 25}, {"skuId": "IPHN-12-64-MINI-BLK", "storeId": "11", "quantity": 13}, {"skuId": "IPHN-12-64-MINI-WHT", "storeId": "11", "quantity": 23}, {"skuId": "IPHN-12-64-MINI-GRN", "storeId": "11", "quantity": 12}]
```

Online Find and replace Console Postbot Runner Capture requests Cookies Trash

Activities Postman janv. 29 21:17 Create Product entries - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import No Environment

Collections Environments History

OMS WebMVC / Product / Create Product entries

GET http://localhost:38080/oms-0.0.1-SNAPSHOT/service/product/register-list

Params Authorization Headers (9) Body Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send Cookies Beautify

1 [  
2 {  
3 "productId": "SM-S20-BLK",  
4 "name": "S20 Black",  
5 "description": "Samsung S20 Black",  
6 "manuf": "Samsung"  
7 },  
8 {  
9 "productId": "SM-S20-WHT",  
10 "name": "S20 White",  
11 "description": "Samsung S20 White",  
12 "manuf": "Samsung"  
13 },  
14 {  
15 "productId": "IPHN-12-64-MINI-BLK",  
16 "name": "iPhone 12 Mini 1",  
17 "description": "iPhone 12 Mini 64GB Black",  
18 "manuf": "Apple"  
19 },  
20 {  
21 "productId": "IPHN-12-64-MINI-WHT",  
22 "name": "iPhone 12 Mini 2",  
23 "description": "iPhone 12 Mini 64GB White",  
24 "manuf": "Apple"  
25 }]

Status: 200 OK Time: 19 ms Size: 721 B Save as example

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

Online Find and replace Console Postbot Runner Capture requests Cookies Trash

Activities Postman

janv. 29 21:18

### Create Order entries - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import No Environment

Collections Environments History

MedRec Refactored OMS Services OMS WebMVC

Inventory Order

Product Shipping ModifyFulfillment StoreSearch

GET Create Order entries

GET Create Inventory entries GET Create Product entries GET Create Order entries

HTTP OMS WebMVC / Order / Create Order entries

GET http://localhost:38080/oms-0.0.1-SNAPSHOT/service/order/multi

Params Authorization Headers (9) Body Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send Cookies Beautify

1 [  
2 {  
3 "customerOrderId": "1",  
4 "primaryPhone": "9525944885",  
5 "customerEmailId": "as@yahoo.com",  
6 "orderStatus": "SUBMITTED",  
7 "firstName": "Anand",  
8 "orderDate": null,  
9 "profileId": null,  
10 "lastName": null,  
11 "entryType": null,

Body Cookies Headers (5) Test Results Status: 200 OK Time: 78 ms Size: 3.87 KB Save as example

Pretty Raw Preview Visualize JSON

1 [  
2 {  
3 "customerOrderId": "1",  
4 "primaryPhone": "9525944885",  
5 "customerEmailId": "as@yahoo.com",  
6 "orderStatus": "SUBMITTED",  
7 "firstName": "Anand",  
8 "orderDate": null,  
9 "profileId": null,  
10 "lastName": null,  
11 "entryType": null,  
12 "billToAddress": {  
13 "billToAddressId": "101",  
14 "firstName": "Anand",  
15 "lastName": "sudhakar",  
16 "address1": null,  
17 "address2": null,  
18 "city": "BLOOMINGTON",  
19 "state": "MN",  
20 "zipCode": "55344"  
21 },  
22 "shipToAddress": {  
23 }

Postbot Runner Capture requests Cookies Trash



## Learning

For the system to learn and understand your application, it needs to analyze as many usage patterns as possible. It is recommended to either collect data while the application is running in production for a while or, if it's installed in pre-production, test the major flows of the application. It is recommended to run your regular set of acceptance tests or regression tests in order to get the best coverage.

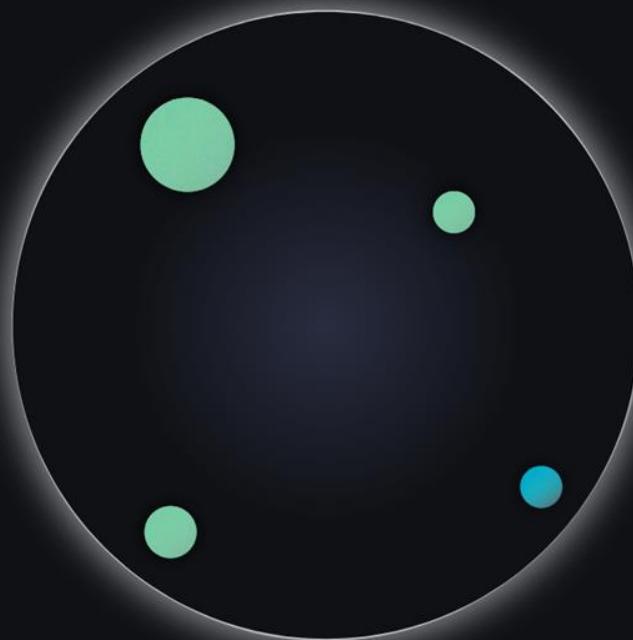
### ACTIONS

View Tech-Debt Report

Select Controllers

oms-java-local

Choose a different controller



START

*Batch Mode*

MEASUREMENTS

DOMAINS  
4

FUNCTIONS  
85

RESOURCES  
19

Cloud Native Innotation Labs

# Step 5

Charger la découverte complète et présenter les bubbles avec focus sur la partie gestion des Inventaire, Produits et Commandes





## Learning

For the system to learn and understand your application, it needs to analyze as many usage patterns as possible. It is recommended to either collect data while the application is running in production for a while or, if it's installed in pre-production, test the major flows of the application. It is recommended to run your regular set of acceptance tests or regression tests in order to get the best coverage.

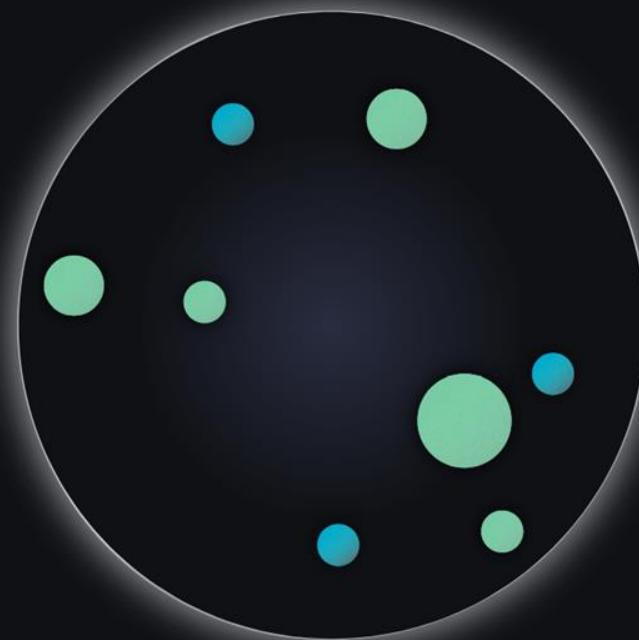
### ACTIONS

View Tech-Debt Report

Select Controllers

oms-java-local

Choose a different controller



START

*Batch Mode*

MEASUREMENTS  
^

DOMAINS  
8

FUNCTIONS  
96

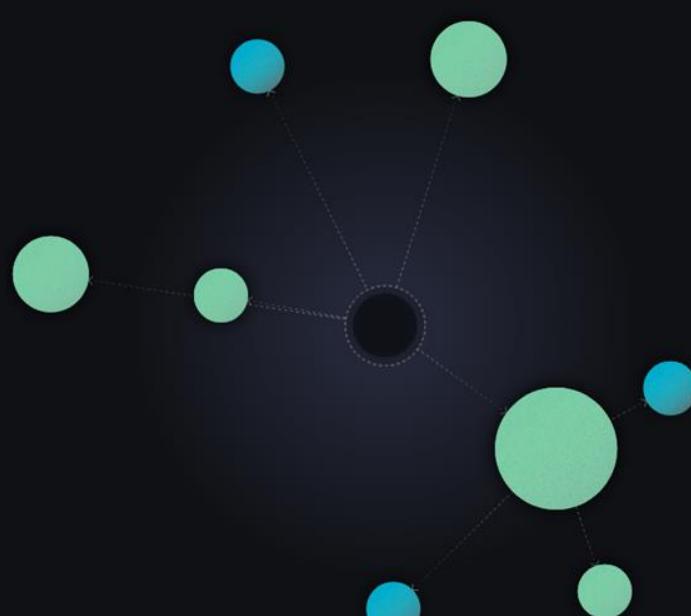
RESOURCES  
40



FILTER

SORT BY: Domain name ▾ FILTERS (4) 🔍

DinersPaymentService	(2)
EmailService	(2)
InventoryController	(3)
OrderController (+1)	(5)
PaymentService	(2)
ProductController	(3)
ShippingPriceController	(2)
StoreSearchController	(2)

| Application | (0) |
| Common Library | (0) |


## ANALYSIS 11 DAYS

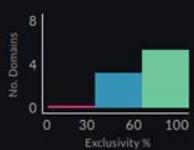
Total Classes	55
Domains	8
Entry Points	25
Classes	18
High Debt Classes	1
Common Classes	4
Resource Exclusivity	75%
Class Exclusivity	83%
Extracted Percentage	100%
Time in Learning	38m
Ignored Classes	0

## ACTIONS

- [View Tech-Debt Report](#)
- [Show History](#)
- [Show Resources Report](#)
- [Configure Parameters](#)
- [Reset Analysis](#)
- [Undo Last Action](#)
- [Reset Graph](#)
- [Reset Ignored Items](#)

## LEGEND

Size = No. of Classes





## ShippingPriceController

## FUNCTION STATES

- Third Party 11
- Common Class 1
- Domain Class 1
- Potential Entry Points 2
- Entry Points 1

## ENTRY POINTS (2)

Merge

## SEARCH

- ShippingPriceController.createShipping()

MOVE REMOVE

- ShippingPriceController.fetchSalesOrder()

MOVE REMOVE

## HISTORY (2)





## ShippingPriceController

## FUNCTION STATES

- Third Party 12
- Common Class 12
- Domain Class 10
- Potential Entry Points 49
- Entry Points 20

## ENTRY POINTS (2)

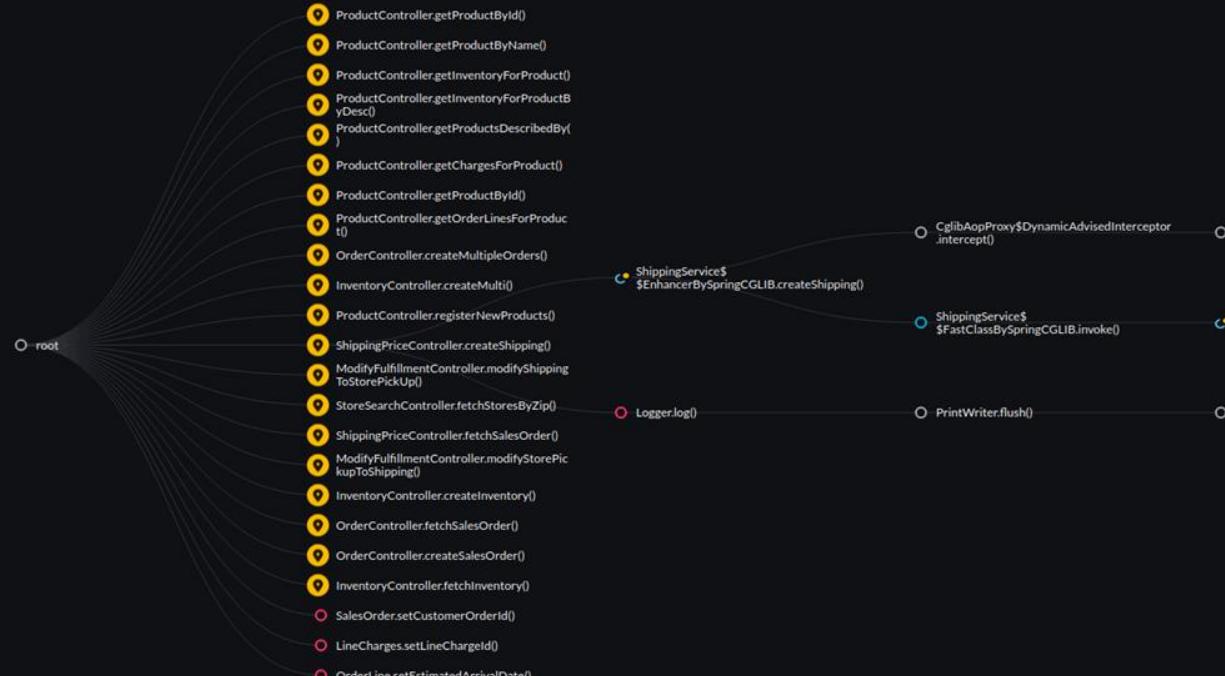
Merge

## SEARCH

- ShippingPriceController.createShipping()

- ShippingPriceController.fetchSalesOrder()

## HISTORY (3)

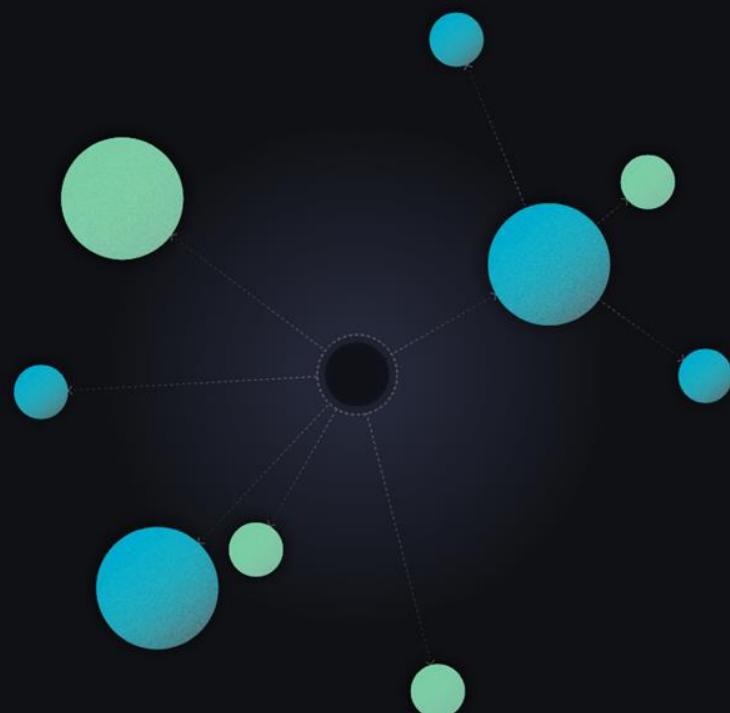




FILTER

SORT BY: Domain name ▾ FILTERS (4) ▾

- DinersPaymentService (2)
  - EmailService (2)
  - InventoryController (4)
  - ModifyFulfillmentController (4)
  - OrderController (2)
  - PaymentService (2)
  - ProductController (4)
  - ShippingPriceController (2)
  - StoreSearchController (2)
- Application (0)
  - Common Library (0)



## ANALYSIS

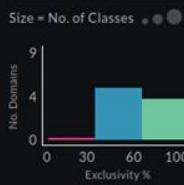
20 DAYS

Total Classes	55
Domains	9
Entry Points	25
Classes	20
High Debt Classes	1
Common Classes	1
Resource Exclusivity	35%
Class Exclusivity	80%
Extracted Percentage	100%
Ignored Classes	0

## ACTIONS

- View Tech-Debt Report
- Show History
- Show Resources Report
- Configure Parameters
- Reset Analysis
- Undo Last Action
- Reset Graph
- Reset Ignored Items

## LEGEND



# Step 6

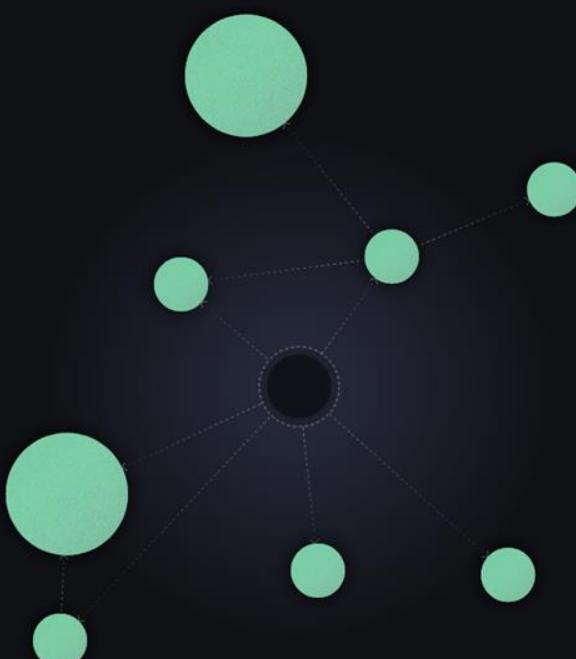
Charger le SNAP-002 et préciser que l'objectif est de passer de la vue éclatée à la vue consolidée (préciser les compétences nécessaires + le jus de cerveau nécessaire) + extraction des codes sources des 3 microservices avec la target Spring



FILTER

SORT BY: Domain name ▾ FILTERS (4) 🔍

- EmailService \* (2)
  - InventoryController \* (3)
  - ModifyFulfillmentController \* (2)
  - OrderController (2)
  - PaymentService \* (3)
  - ProductController \* (2)
  - ShippingPriceController \* (2)
  - StoreSearchController (2)
- Application (0)
  - Common Library (0)



## ANALYSIS

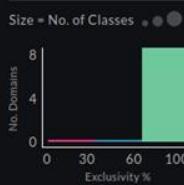
21 DAYS

Total Classes	55
Domains	8
Entry Points	27
Classes	18
High Debt Classes	1
Common Classes	3
Resource Exclusivity	37%
Class Exclusivity	100%
Extracted Percentage	100%
Ignored Classes	0

## ACTIONS

- View Tech-Debt Report
- Show History
- Show Resources Report
- Configure Parameters
- Reset Analysis
- Undo Last Action
- Reset Graph
- Reset Ignored Items

## LEGEND

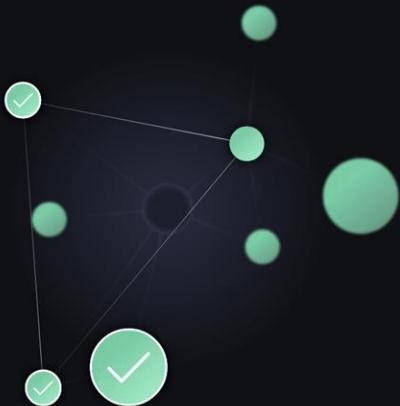


ANALYSIS		20 MINUTES AGO
Services	8	
Entry Points	27	
Classes	18	
Common Classes	3	
Resource Exclusivity	68%	
Class Exclusivity	100%	
Extracted Percentage	100%	
Ignored Classes	0	
Total Classes	55	

ACTIONS

- [View Assessment Report](#)
- [Show History](#)
- [Show Resources Report](#)
- [Configure Parameters](#)
- [Create Snapshot](#)
- [Reset Analysis](#)
- [Undo Last Action](#)
- [Reset Graph](#)
- [Reset Ignored Items](#)
- [Download Measurement](#)

LEGEND



Candidates for extraction (4)

All Services (8)

SORT BY: Service name ▾ FILTERS (4) 🔍

- 5 Static classes Excl 100% [VIEW >](#)
- 12 Resources Excl 100% [VIEW >](#)
- 0 Dead-code classes [VIEW >](#)

Runtime 15.49% [SHOW DEPENDENCIES >](#)

OrderController (2)

- 3 Entry points [EXPLORE TREE >](#)
- 2 Dynamic classes Excl 100% [VIEW >](#)
- 3 Static classes Excl 66% [VIEW >](#)
- 17 Resources Excl 58% [VIEW >](#)
- 0 Dead-code classes [VIEW >](#)

ProductController (2)

- 9 Entry points [EXPLORE TREE >](#)
- 2 Dynamic classes Excl 100% [VIEW >](#)
- 4 Static classes Excl 100% [VIEW >](#)
- 26 Resources Excl 88% [VIEW >](#)

LEARNING ANALYSIS SERVICE CREATION MODERNIZING

### Service Configuration

Group ID	com.oms
Target Platform	SPRINGBOOT
Target Version	JAVA_8
Dependency Repository Type	MAVEN

Generate endpoints

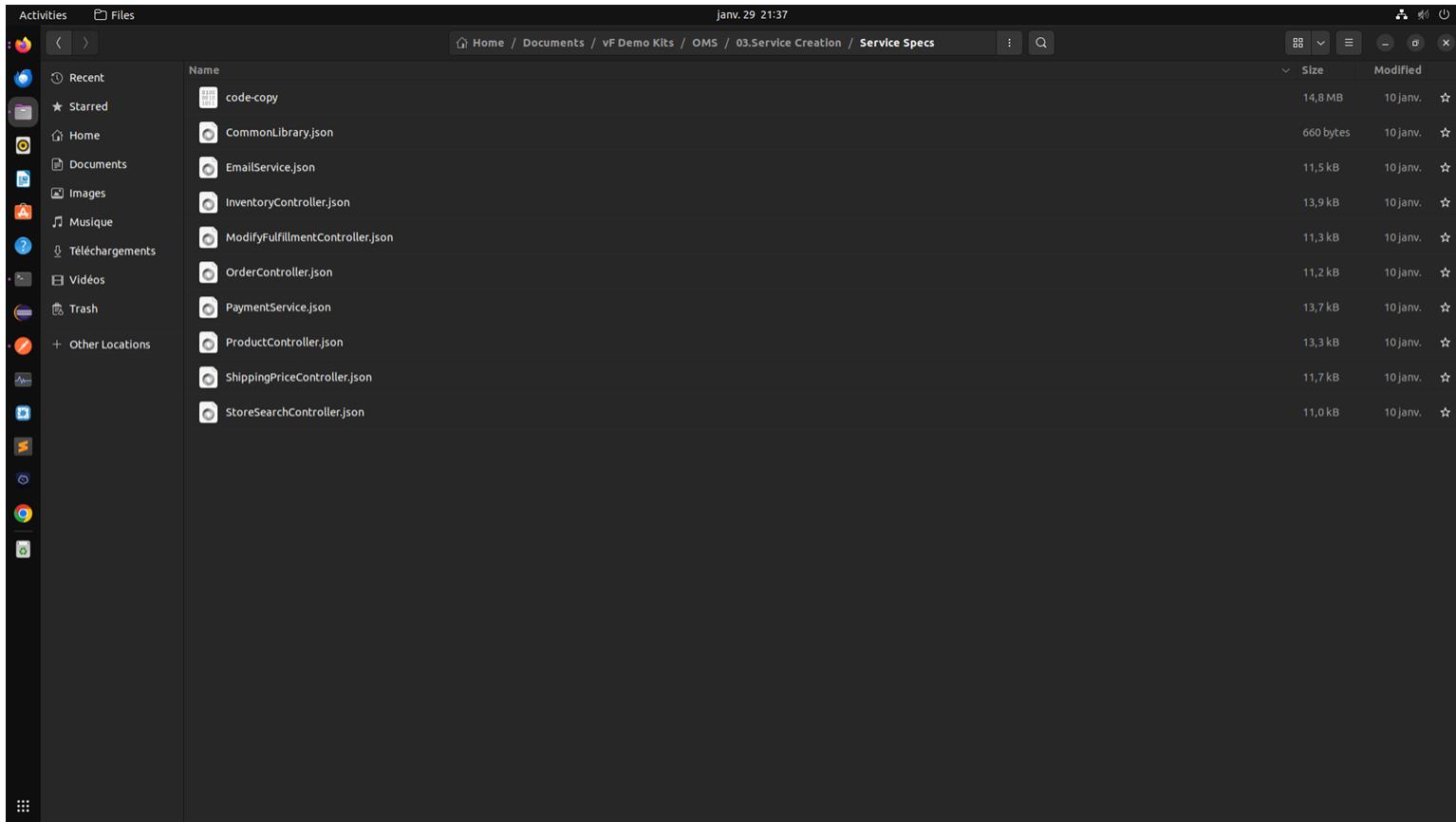
Download common library incrementally

[CANCEL](#) [SUBMIT](#)

[CONFIGURE](#) [DOWNLOAD](#)

Activities  Files janv. 29 21:37

Home / Documents / vF Demo Kits / OMS / 03.Service Creation / Service Specs



The screenshot shows a dark-themed file manager window. On the left is a sidebar with icons for various locations like Recent, Starred, Home, Documents, Images, Musique, Téléchargements, Vidéos, and Trash. Below these are sections for Other Locations and a list of recently modified files. The main area displays a list of JSON files under the path 'Service Specs'. The columns are Name, Size, and Modified.

Name	Size	Modified
code-copy	14,8 MB	10 janv.
CommonLibrary.json	660 bytes	10 janv.
EmailService.json	11,5 kB	10 janv.
InventoryController.json	13,9 kB	10 janv.
ModifyFulfillmentController.json	11,3 kB	10 janv.
OrderController.json	11,2 kB	10 janv.
PaymentService.json	13,7 kB	10 janv.
ProductController.json	13,3 kB	10 janv.
ShippingPriceController.json	11,7 kB	10 janv.
StoreSearchController.json	11,0 kB	10 janv.

Cloud Native Innotation Labs

# Step 7

On bascule sur le studio Eclipse +  
explication de comment c'est  
obtenu + montrer rapidement les  
fichiers modifiés pour le  
microservice ProductController

Activities Eclipse IDE for Java Developers - 2023-12 janv. 29 21:38

eclipse-workspace - ProductController/pom.xml - Eclipse IDE

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>2.5.0</version>
        <relativePath/>
        <!-- lookup parent from repository -->
    </parent>
    <groupId>com.oms</groupId>
    <artifactId>ProductController</artifactId>
    <version>1.0-SNAPSHOT</version>
    <build>
        <plugins>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-compiler-plugin</artifactId>
                <configuration>
                    <source>8</source>
                    <target>8</target>
                </configuration>
            </plugin>
            <!--
                The spotless plugin is used to remove unnecessary imports
                If it identifies unnecessary imports it will issue an error
                To fix these error run: mvn spotless:apply
                To disable add -DskipSpotless
                You could also do: mvn clean spotless:apply install
            -->
        </plugins>
    </build>
</project>
```

File Edit Source Navigate Search Project Run Window Help

Package Explorer X

ProductController/pom.xml X spring-config.xml ProductControllerApp.java application.properties ProductService.java

Overview Dependencies Dependency Hierarchy Effective POM pom.xml

Problems Javadoc Declaration Console X

No consoles to display at this time.

Writable Insert 1:1:0

Activities Eclipse IDE for Java Developers - 2023-12 janv. 29 21:39

eclipse-workspace - ProductController/src/main/resources/spring-config.xml - Eclipse IDE

File Edit Source Navigate Search Project Run Window Help

Package Explorer X ProductController/pom.xml spring-config.xml ProductControllerApp.java application.properties ProductService.java

```
2
3
4      <!-- CONTEXT -->
5      <context:component-scan base-package="com.oms"/>
6
7      <jpa:repositories base-package="com.oms.repository" entity-manager-factory-ref="entityManagerFactory"/>
8
9      <bean class="com.oms.util.Logger" id="logger">
10         <property name="path" value="oms-log.txt"/>
11     </bean>
12
13     <beans profile="replicated">
14         <gfe:cache/>
15         <gfe:replicated-region id="inventoryRegion" name="Inventory">
16             <gfe:cache-loader>
17                 <bean class="com.oms.gemfire.GemfireCacheLoader">
18                     <property name="repository" ref="inventoryRepository"/>
19                 </bean>
20             </gfe:cache-loader>
21             <gfe:cache-writer>
22                 <bean class="com.oms.gemfire.GemfireCacheWriter">
23                     <property name="repository" ref="inventoryRepository"/>
24                 </bean>
25             </gfe:cache-writer>
26         </gfe:replicated-region>
27     </beans>
28     <beans profile="local">
29         <gfe:cache/>
30         <gfe:local-region id="inventoryRegion" name="Inventory">
31             <gfe:cache-loader>
32                 <bean class="com.oms.gemfire.GemfireCacheLoader">
33                     <property name="repository" ref="inventoryRepository"/>
34                 </bean>
35             </gfe:cache-loader>
36             <gfe:cache-writer>
37                 <bean class="com.oms.gemfire.GemfireCacheWriter">
38                     <property name="repository" ref="inventoryRepository"/>
39                 </bean>
40             </gfe:cache-writer>
41         </gfe:local-region>
42     </beans>
43     <beans profile="client">
44         <bean class="org.springframework.data.gemfire.mapping.MappingPdxSerializer" id="pdxSerializer"/>
45         <gfe:client-cache pdx-serializer-ref="pdxSerializer"/>
46     </beans>
47     <gfe:pool>
```

Problems Javadoc Declaration Console X

No consoles to display at this time.

Writable Insert 13:23:1611

Activities Eclipse IDE for Java Developers - 2023-12 janv. 29 21:39

eclipse-workspace - ProductController/src/main/java/com/oms/productcontroller/ProductControllerApp.java - Eclipse IDE

File Edit Source Refactor Source Navigate Search Project Run Window Help

Package Explorer X ProductController/pom.xml spring-config.xml ProductControllerApp.java application.properties ProductService.java

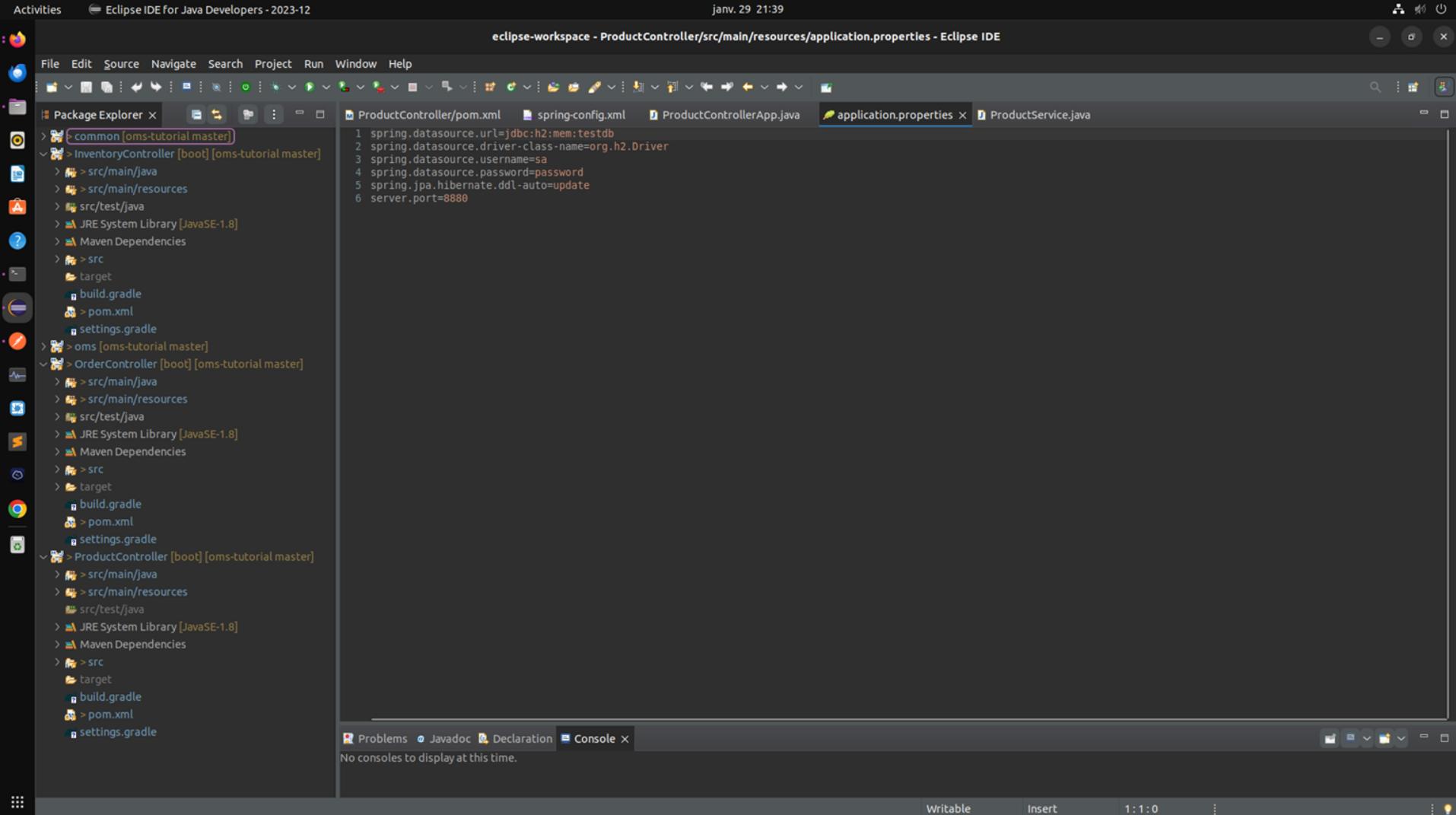
```
1 package com.oms.productcontroller;
2
3 import org.springframework.boot.SpringApplication;
4
5 @SpringBootApplication
6 @EntityScan(basePackages = {"com.oms.entity"})
7 @ImportResource({"classpath:spring-config.xml"})
8 public class ProductControllerApp {
9     public static void main(String[] args) {
10         SpringApplication.run(ProductControllerApp.class, args);
11     }
12 }
```

Problems Javadoc Declaration Console X

No consoles to display at this time.

Writable Smart Insert 1:1:0

This screenshot shows the Eclipse IDE interface for Java development. The title bar indicates the workspace is 'eclipse-workspace' and the current file is 'ProductControllerApp.java'. The status bar at the bottom shows the date and time as 'janv. 29 21:39' and the code editor as '1:1:0'. The left sidebar contains the 'Package Explorer' showing project structures for 'common', 'InventoryController', 'OrderController', and 'ProductController'. The 'ProductController' project is currently selected. The right side of the screen displays the Java code for 'ProductControllerApp.java' which includes imports for SpringBootApplication, EntityScan, and ImportResource annotations, and a main method that runs the application. Below the code editor is a 'Console' tab which is currently empty. The bottom navigation bar includes tabs for 'Problems', 'Javadoc', 'Declaration', and 'Console'. The status bar also includes 'Writable' and 'Smart Insert' buttons.



Activities Eclipse IDE for Java Developers - 2023-12 janv. 29 21:40

eclipse-workspace - ProductController/src/main/java/com/oms/service/ProductService.java - Eclipse IDE

File Edit Source Refactor Source Navigate Search Project Run Window Help

Package Explorer X ProductController/pom.xml spring-config.xml ProductControllerApp.java application.properties ProductService.java X

```
1 package com.oms.service;
2
3 import java.util.LinkedList;
4
5 @Service
6 @Transactional
7 public class ProductService {
8
9     @Autowired
10    ProductRepository productRepository;
11
12    @Autowired
13    OrderLineRepository orderLineRepository;
14
15    @Autowired
16    Logger logger;
17
18    // WebClient inventoryServiceClient = WebClient.create("http://localhost:8081");
19    WebClient inventoryServiceClient = WebClient.create("http://vf-inventory:8080");
20
21    public Product registerProduct(Product product) {
22        logger.log(this.getClass().getName());
23        Product response = productRepository.save(product);
24        return response;
25    }
26
27    public Product getProductById(String pid) {
28        logger.log(this.getClass().getName());
29        Optional<Product> findById = productRepository.findById(pid);
30        return findById.orElse(null);
31    }
32
33    public Product getName(String name) {
34        logger.log(this.getClass().getName());
35        Optional<Product> findByName = productRepository.findByName(name);
36        return findByName.orElse(null);
37    }
38
39    public List<Product> getAllProducts() {
40        logger.log(this.getClass().getName());
41        return productRepository.findAll();
42    }
43
44    public List<Product> getProductsDescribedWith(String text) {
45        logger.log(this.getClass().getName());
46    }
47}
```

Problems Javadoc Declaration Console X

No consoles to display at this time.

Writable Smart Insert 1:1:0

Cloud Native Innotation Labs

# Step 8

On termine en présentant la création  
des containers et en montrant le  
test des APIs REST

Activities Terminal janv. 29 21:42 franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-services

franck@fbe-linux-vm: ~/Applications/apache/tomcat-9.0.84/bin

franck@fbe-linux-vm: ~/Sandbox/oms-tutorial/oms-services

```
franck@fbe-linux-vm:~/Sandbox/oms-tutorial/oms-services$ docker images | grep oms
OMS-inventorycontroller          latest      fe41406501d0  44 years ago  307MB
microk8s:32000/oms-inventorycontroller  latest      fe41406501d0  44 years ago  307MB
microk8s:32000/oms-ordercontroller    latest      3c13913c753a  44 years ago  253MB
OMS-ordercontroller                latest      3c13913c753a  44 years ago  253MB
OMS-productcontroller              latest      2ee9d394f431  44 years ago  261MB
microk8s:32000/oms-productcontroller latest      2ee9d394f431  44 years ago  261MB
microk8s:32000/oms-productcontroller <none>     441f5557fb70  44 years ago  261MB
microk8s:32000/oms-productcontroller <none>     cf7ec29beb6   44 years ago  261MB
franck@fbe-linux-vm:~/Sandbox/oms-tutorial/oms-services$
```

Activities Sublime Text janv. 29 21:43

~/Sandbox/oms-tutorial/vfunction-services.yaml (Sandbox) - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

FOLDERS

- Sandbox
  - docker-for-oracle-webclient
  - docker-images
  - eclipse-workspace
  - medrec
  - medrec-extracted
  - oms-tutorial
    - oms-services
    - oms-webmvn
  - service-specs
    - .gitignore
    - LICENSE
    - README.md
- vfunction-services.yaml
- services

delete from BILL\_TO\_ADDRESS; vfunction-services.yaml

```
1 apiVersion: apps/v1
2 kind: Deployment
3 metadata:
4   labels:
5     app: h2-db
6   name: h2-db
7   namespace: vfunction
8 spec:
9   replicas: 1
10  selector:
11    matchLabels:
12      app: h2-db
13  template:
14    metadata:
15      labels:
16        app: h2-db
17    spec:
18      containers:
19        - image: oscarfonts/h2
20          name: h2
21          env:
22            - name: H2_OPTIONS
23              value: "-ifNotExists"
24          ports:
25            - containerPort: 81
26            - containerPort: 1521
27 ...
28 apiVersion: v1
29 kind: Service
30 metadata:
31   labels:
```

Line 1, Column 1 master 39 Spaces: 2 YAML

The sidebar contains the following sections:

- Cluster
- Applications
- Nodes
- Workloads** (selected)
  - Overview
  - Pods
  - Deployments** (selected)
  - Daemon Sets
  - Stateful Sets
  - Replica Sets
  - Replication Controllers
  - Jobs
  - Cron Jobs
- Config
- Network
  - Services
  - Endpoints
  - Ingresses
  - Ingress Classes
  - Network Policies
  - Port Forwarding
- Storage
- Namespaces
- Events
- Helm
- Access Control
- Custom Resources

Overview Pods Deployments Daemon Sets Stateful Sets Replica Sets Replication Controllers Jobs Cron Jobs

### Deployments

4 items

Namespace: vfunction

Name	Namespace	Pods	Replicas	Age	Conditions	⋮
h2-db	vfunction	1/1	1	24d	Available Progressing	⋮
vf-product	vfunction	1/1	1	24d	Available Progressing	⋮
vf-inventory	vfunction	1/1	1	24d	Available Progressing	⋮
vf-order	vfunction	1/1	1	24d	Available Progressing	⋮

Activities Lens Local fbe-linux-vm FR

janv. 29 21:46

UPGRADE

microk8s

Overview Pods Deployments Daemon Sets Stateful Sets Replica Sets Replication Controllers Jobs Cron Jobs

Pods 4 items Namespace: vfunction

Name	Namespace	Contai...	CPU	Memory	Restarts	Controlled By	Node	QoS	Age	Status	⋮
h2-db-659bc7c6c8-tgx4v	vfunction	■	N/A	N/A	2	ReplicaSet	fbe-linux-*	BestEffort	24d	Running	⋮
vf-inventory-5f55768554-44x82	vfunction	■	N/A	N/A	59	ReplicaSet	fbe-linux-*	BestEffort	24d	Running	⋮
vf-order-6cf8cfb89-mmmtcq	vfunction	■	N/A	N/A	59	ReplicaSet	fbe-linux-*	BestEffort	24d	Running	⋮
vf-product-6d54bf8bf-85zld	vfunction	■	N/A	N/A	60	ReplicaSet	fbe-linux-*	BestEffort	24d	Running	⋮

Workloads Overview Pods Deployments Daemon Sets Stateful Sets Replica Sets Replication Controllers Jobs Cron Jobs

Config Network Services Endpoints Ingresses Ingress Classes Network Policies Port Forwarding Storage Namespaces Events Helm Access Control Custom Resources

- Cluster
- Applications
- Nodes
- Workloads
  - Overview
  - Pods
  - Deployments
  - Daemon Sets
  - Stateful Sets
  - Replica Sets
  - Replication Controllers
  - Jobs
  - Cron Jobs
- Config
- Network
  - Services
  - Endpoints
  - Ingresses
  - Ingress Classes
  - Network Policies
  - Port Forwarding
- Storage
- Namespaces
- Events
- Helm
- Access Control
- Custom Resources

janv. 29 21:47

UPGRADE Local fbe-linux-vm FR

### Services

Name	Namespace	Type	Cluster IP	Ports	External IP	Selector	Age	Status	⋮
h2-admin	vfunction	NodePort	10.152.183.14	81:30395/TCP	-	app=h2-db	24d	Active	⋮
h2-data	vfunction	NodePort	10.152.183.39	1521:32151/TCP	-	app=h2-db	24d	Active	⋮
vf-inventory	vfunction	NodePort	10.152.183.19	8080:30081/TCP	-	app=vf-inve	24d	Active	⋮
vf-order	vfunction	NodePort	10.152.183.47	8080:30083/TCP	-	app=vf-orde	24d	Active	⋮
vf-product	vfunction	NodePort	10.152.183.24	8080:30082/TCP	-	app=vf-proc	24d	Active	⋮

Activities Postman janv. 29 21:48

Create Multiple Inventory Items - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Bell Upgrade

My Workspace New Import POST Create Multiple Inventory Items

Collections Environments History

OMS Services / Inventory / Create Multiple Inventory Items

POST http://microk8s:30081/inventory/multi-create

Params Authorization Headers (9) Body Pre-request Script Tests Settings

Body (JSON)

```
1 [  
2   {  
3     "skuId": "SM-S20-BLK",  
4     "storeId": "11",  
5     "quantity": 20  
6   },  
7   {  
8     "skuId": "SM-S20-WHT",  
9     "storeId": "11",  
10    "quantity": 25  
11 }]
```

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 1289 ms Size: 452 B Save as example

Pretty Raw Preview Visualize JSON

```
1 [  
2   {  
3     "skuId": "SM-S20-BLK",  
4     "storeId": "11",  
5     "quantity": 20  
6   },  
7   {  
8     "skuId": "SM-S20-WHT",  
9     "storeId": "11",  
10    "quantity": 25  
11 },  
12   {  
13     "skuId": "IPHNL-12-64-MINI-BLK",  
14     "storeId": "11",  
15     "quantity": 13  
16   },  
17   {  
18     "skuId": "IPHNL-12-64-MINI-WHT",  
19     "storeId": "11",  
20     "quantity": 23  
21   },  
22 ]
```

Online Find and replace Console

Postbot Runner Capture requests Cookies Trash

Activities Postman janv. 29 21:49 Get inventory Item - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import POST Create Multiple Inventory | GET Get Inventory Item +

OMS Services / Inventory / Get Inventory Item

GET http://microk8s:30081/inventory/SM-S20-BLK

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	Key	Value	Description	... Bulk Edit
	Key	Value	Description	

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 38 ms Size: 215 B Save as example

Pretty Raw Preview Visualize JSON

```
1 {  
2   "skuId": "SM-S20-BLK",  
3   "storeId": "11",  
4   "quantity": 20  
5 }
```

Postbot Runner Capture requests Cookies Trash

Activities Postman janv. 29 21:49 Create Products - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import POST Create Products +

OMS Services / Products / Create Products

POST http://microk8s:30082/product/register-list

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send Cookies Beautify

POST Create Products

GET Find Product By ID  
GET Find Product Inventory By ID  
GET Find Inventory By Product D...  
GET Find All Products  
GET Find Order Lines By Product ...  
GET Get Charges For Product  
Orders  
OMS WebMVC

1 [  
2 {  
3 "productId": "SM-S20-BLK",  
4 "name": "S20 Black",  
5 "description": "Samsung S20 Black",  
6 "manuf": "Samsung"  
7 },  
8 {  
9 "productId": "SM-S20-WHT",  
10 "name": "S20 White",  
11 "description": "Samsung S20 White",  
12 "manuf": "Samsung"  
13 },  
14 {  
15 "productId": "IPHN-12-64-MINI-BLK",  
16 "name": "iPhone 12 Mini 1",  
17 "description": "iPhone 12 Mini 64GB Black",  
18 "manuf": "Apple"  
19 },  
20 {  
21 "productId": "IPHN-12-64-MINI-WHT",  
22 "name": "iPhone 12 Mini 2",  
23 "description": "iPhone 12 Mini 64GB White",  
24 "manuf": "Apple"  
25 }]

Status: 200 OK Time: 701 ms Size: 721 B Save as example

Pretty Raw Preview Visualize JSON

Body Cookies Headers (5) Test Results

Postbot Runner Capture requests Cookies Trash

Activities Postman janv. 29 21:50 Find All Products - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import POST Create Products GET Find All Products +

No Environment

Collections Environments History

OMS Services / Products / Find All Products

GET http://microk8s:30082/product/all

Params Authorization Headers (6) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send Cookies Beautify

POST Create Products

GET Find Product By ID

GET Find Product Inventory By ID

GET Find Inventory By Product D...

GET Find All Products

GET Find Order Lines By Product ...

GET Get Charges For Product

Orders

GET OMS WebMVC

Body Cookies Headers (5) Test Results Status: 200 OK Time: 303 ms Size: 721 B Save as example

Pretty Raw Preview Visualize JSON

```
1 [  
2 {  
3   "productId": "SM-S20-BLK",  
4   "name": "S20 Black",  
5   "description": "Samsung S20 Black",  
6   "manuf": "Samsung"  
7 },  
8 {  
9   "productId": "SM-S20-WHT",  
10  "name": "S20 White",  
11  "description": "Samsung S20 White",  
12  "manuf": "Samsung"  
13 },  
14 {  
15   "productId": "IPHN-12-64-MINI-BLK",  
16   "name": "iPhone 12 Mini 1",  
17   "description": "iPhone 12 Mini 64GB Black",  
18   "manuf": "Apple"  
19 },  
20 {  
21   "productId": "IPHN-12-64-MINI-WHT",  
22   "name": "iPhone 12 Mini 2",  
23   "description": "iPhone 12 Mini 64GB White",  
24   "manuf": "Apple"  
25 }]
```

Online Find and replace Console Postbot Runner Capture requests Cookies Trash

Activities Postman janv. 29 21:50 Create Multiple Orders - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import POST Create Multiple Orders +

OMS Services / Orders / Create Multiple Orders

POST http://microk8s:30083/order/multi

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

POST Create Order POST Create Multiple Orders GET Get Order By ID

OMS WebMVC

Save Send Cookies Beautify

POST Create Order POST Create Multiple Orders GET Get Order By ID

OMS WebMVC

Body Cookies Headers (5) Test Results

Status: 200 OK Time: 964 ms Size: 3.87 KB Save as example

Pretty Raw Preview Visualize JSON

```
[{"customerOrderId": "1", "primaryPhone": "+9525944885", "customerEmailId": "as@yahoo.com", "orderStatus": "SUBMITTED", "firstName": "Anand", "orderDate": null, "profileId": null, "lastName": null, "entryType": null}, {"customerOrderId": "1", "primaryPhone": "9525944885", "customerEmailId": "as@yahoo.com", "orderStatus": "SUBMITTED", "firstName": "Anand", "orderDate": null, "profileId": null, "lastName": null, "entryType": null, "billToAddress": {"billToAddressId": "101", "firstName": "Anand", "lastName": "sudhakar", "address1": null, "address2": null, "city": "BLOOMINGTON", "state": "MN", "zipCode": "55344"}, "shipToAddress": {}}]
```

Postbot Runner Capture requests Cookies Trash

Activities Postman janv. 29 21:51 Get Order By ID - My Workspace

File Edit View Help

Home Workspaces API Network

Search Postman

Invite Settings Notifications Upgrade

My Workspace New Import POST Create Multiple Orders GET Get Order By ID +

No Environment

Collections Environments History OMS Services / Orders / Get Order By ID

GET http://microk8s:30083/order/2

Params Authorization Headers (6) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Send Cookies Beautify

POST Create Order POST Create Multiple Orders GET Get Order By ID

OMS WebMVC

Body Cookies Headers (5) Test Results Status: 200 OK Time: 91 ms Size: 1.39 KB Save as example

Pretty Raw Preview Visualize JSON

```
1 {
2     "customerOrderId": "2",
3     "primaryPhone": "9525944805",
4     "customerEmailId": "as@yahoo.com",
5     "orderStatus": "SUBMITTED",
6     "firstName": "Anand",
7     "orderDate": null,
8     "profileId": null,
9     "lastName": null,
10    "entryType": null,
11    "billToAddress": {
12        "billToAddressId": "101",
13        "firstName": "Anand",
14        "lastName": "sudhakar",
15        "address1": null,
16        "address2": null,
17        "city": "BLOOMINGTON",
18        "state": "MN",
19        "zipCode": "55344"
20    },
21    "shipToAddress": {
22        "shipToAddressId": "102",
23        "-----"
24    }
25}
```

Postbot Runner Capture requests Cookies Trash

Cloud Native Innotation Labs

# Step 9

OpenMind, expliquer comment cela aide à simplifier la modernisation, les nouvelles features, le cycle de release, les opérations ...

# Cette modernisation impacte de multiples domaines

