Orchestration de processing image CO3D

COMET: Animation Orchestration de Traitements Distribués

DEFENCE AND SPACE

Christophe ARGUEL (Capgemini) et Vincent GRESSELIN (Airbus Defence & Space)

31 Mai 2022

AIRBUS

Agenda

- Contexte projet CO3D
- Framework d'orchestration Zeebe
- Mise en œuvre de Zeebe pour CO3D
- Conclusion & next steps

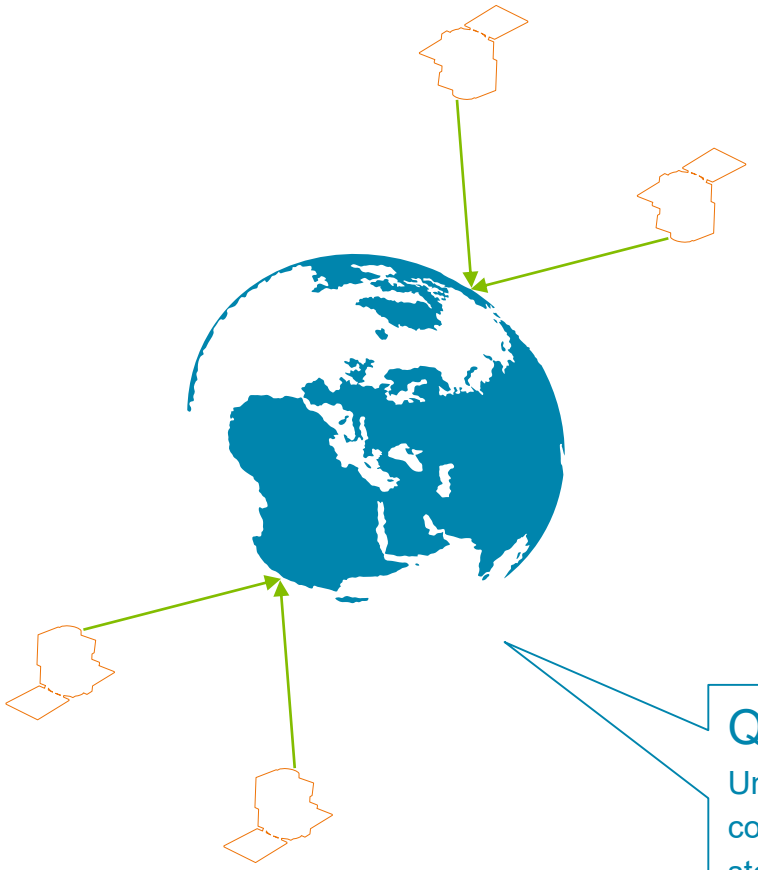
Contexte projet CO3D

Capgemini AIRBUS

Le projet CO3D

MNS

Le but du projet est de fournir un **modèle numérique de surface** (MNS) des terres émergées du monde avec une résolution altimétrique d'un mètre environ.



CLOUD

La production des dalles 3D est effectuée dans le cloud Orange.

50 cm

La résolution optique des satellites.

4

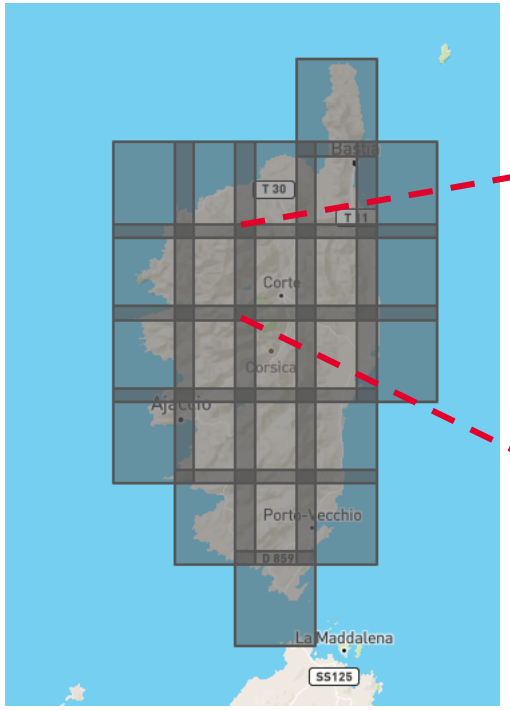
Le nombre de satellites optiques de la constellation CO3D.

Quadri-stereo

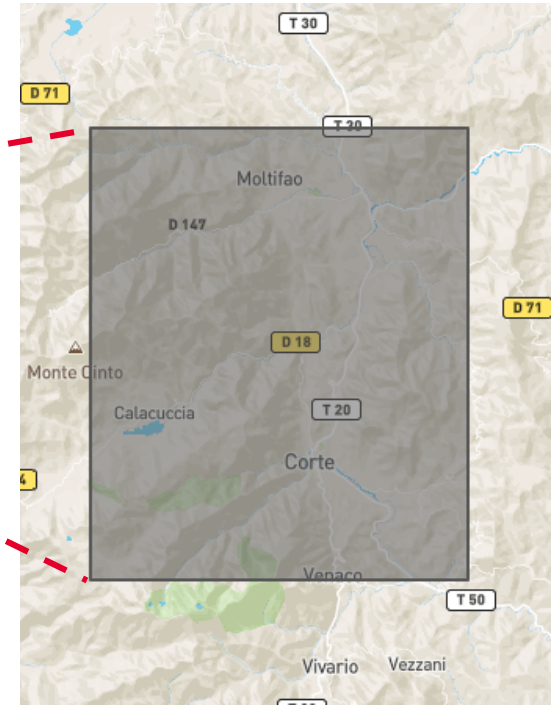
Une partie des dalles MNS sera construite à partir d'acquisitions quadri-stéréo.

Les chantiers de production 3D

Chantier L4 de production



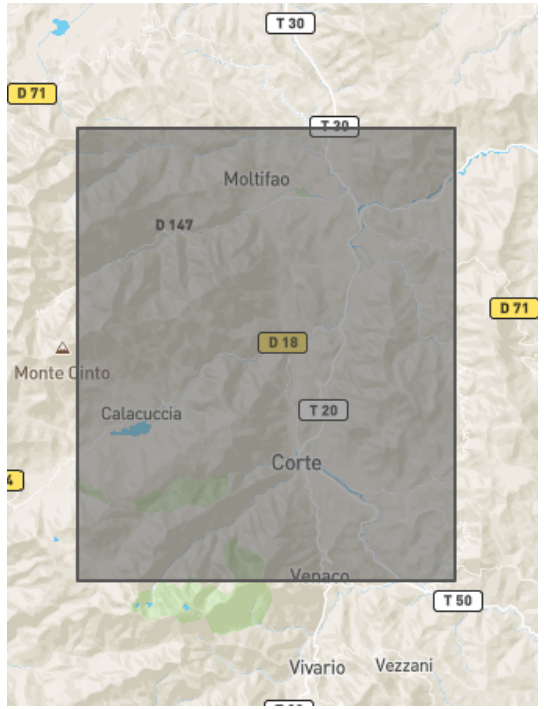
Dalle de production L4



Chantier de production L3

La complétude d’un chantier avant production.

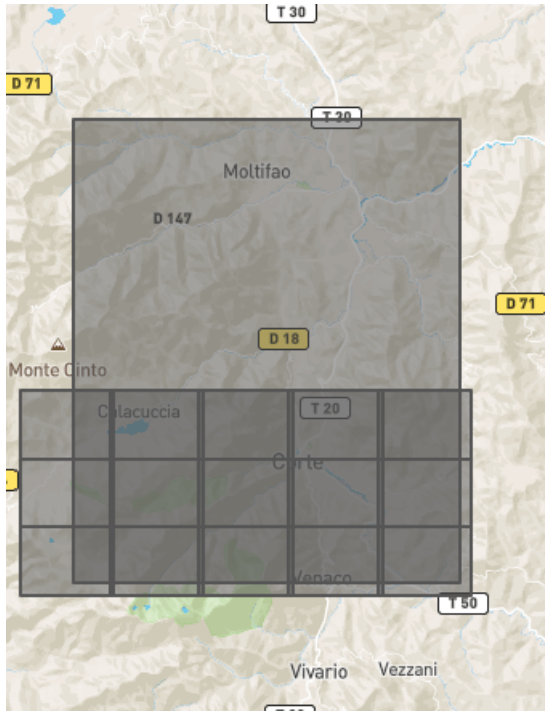
Chantier couvert à 0%, en attente de produits L0.



T0

Aucune paire de produit ne correspond aux critères du chantier

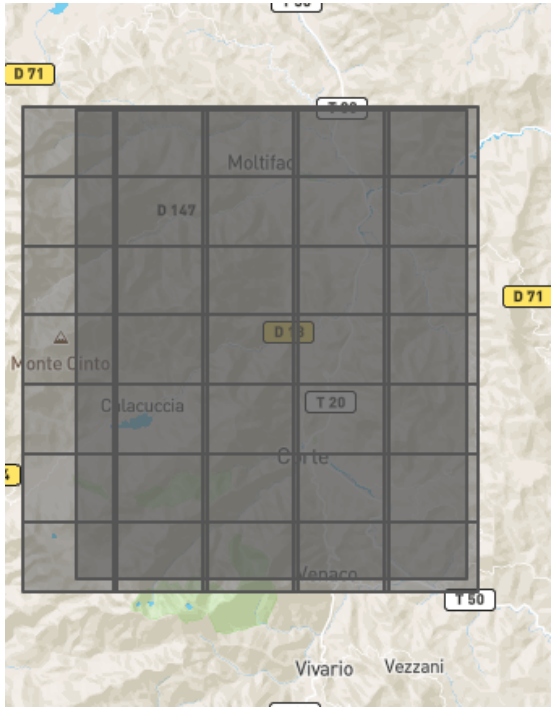
Chantier couvert à ~ 43%, en attente de produits L0.



T0 + 1

15 paires de produits correspondent aux critères du chantier mais ce n’est pas suffisant.

Chantier couvert à 100%, prêt à lancer une production de dalle.



T0 + 2

Il y a assez de paires de produits qui correspondent aux critères du chantier pour le couvrir entièrement.

Environnement d'exécution

Services métiers

Catalogue

Gestion de chantier de production

Workflow Ingestion L0

Workflow vérification des conditions

Workflow de Production 3D

Services communs

zeebe by Camunda

mongoDB

elasticsearch

kafka

amazon S3

fluentbit fluentd Prometheus kibana Grafana

kubernetes

Cloud Orange

Capgemini AIRBUS

Choix de l'orchestrateur de workflow

Critères de sélection du framework

- Bien documenté, avec une documentation claire et à jour.
- Framework d'orchestration
- Open Source
- Langage agnostique
- Disponibilité au moins d'un client Java
- Scalable
- Déployable facilement sur Kubernetes (disponibilité d'un chart HELM)
- Avoir certains opérateurs pour faire des OR, AND, XOR, du parallélisme, etc.
- Dépendance réduite à d'autres services (bases de données, bus de données, etc)
- Facilité de monitoring de l'exécution des workflows.

01
- RETEX projet Sobloo : "Command & control" difficile avec une architecture orientée chorégraphie.

02
- Recherche d'un framework d'orchestration scalable => **CONDUCTOR** de Netflix

03
- Début ingénierie CO3D => On souhaite partir sur de l'orchestration, choix de CONDUCTOR.

04
- RETEX CONDUCTOR => architecture et concepts intéressants mais mauvais suivi du produit et difficultés de déploiement hors environnement AWS

05
- Analyse documentaire d'autres frameworks d'orchestration => Il en existe beaucoup, tous avec leurs spécificités. Zeebe est proche de CONDUCTOR dans les principes mais est beaucoup plus « production ready » que CONDUCTOR.

06
- Tests de performance et d'utilisation de Zeebe concluants. On sélectionne cette solution pour le projet CO3D.

Framework d’orchestration Zeebe

Capgemini AIRBUS

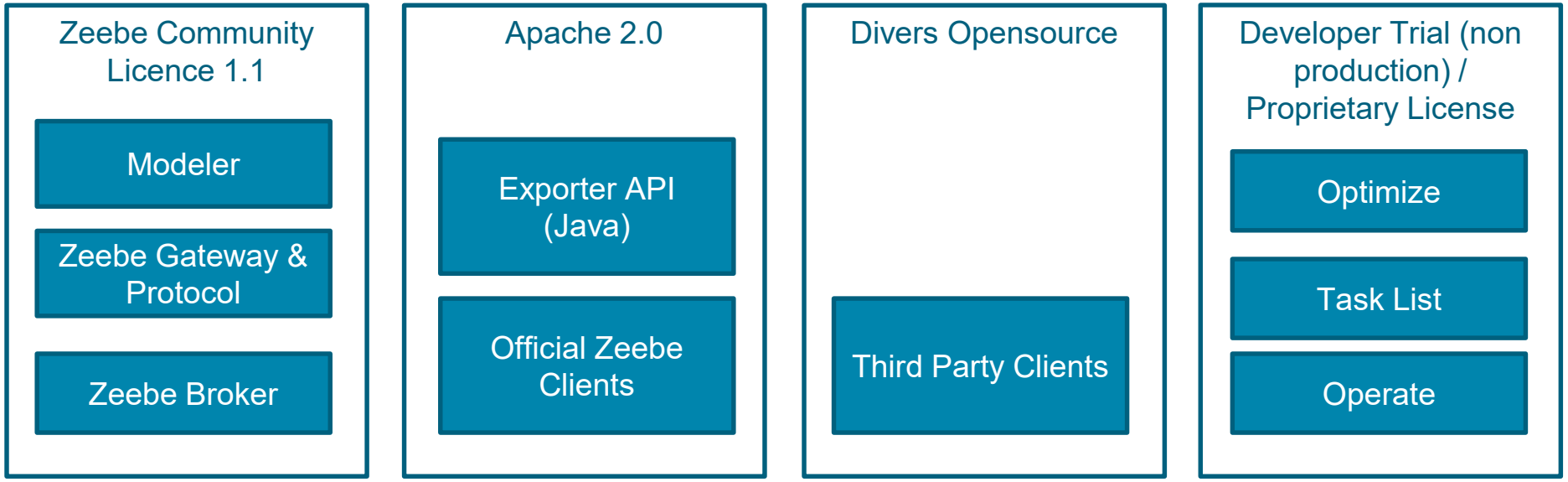
Genèse de Zeebe

- Solution **d'orchestration de workflows** développée par Camunda à partir de 2017
- Version 1.0 en mai 2021

- La **scalabilité** est au cœur de l'architecture de Zeebe
- Workflows décrits en **BPMN** 2.0
- Agnostique du langage de programmation
- Fault Tolerant et Haute Disponibilité

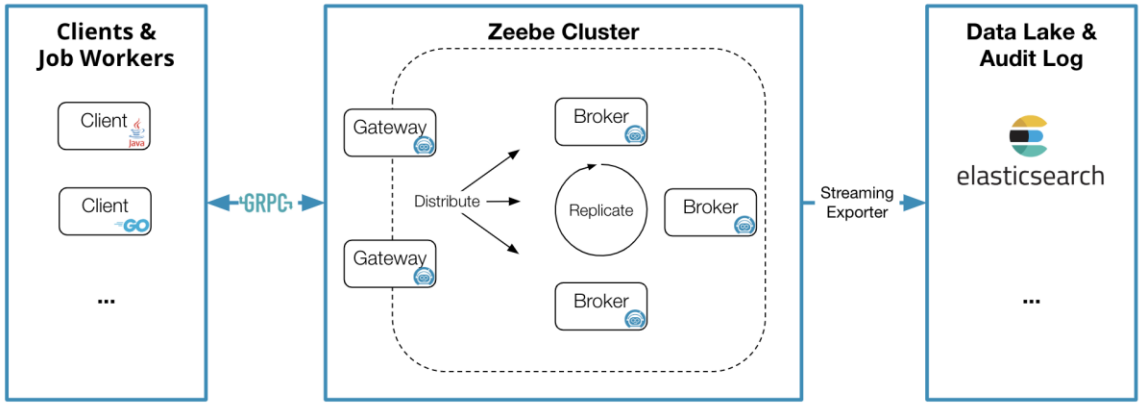
Offre Zeebe

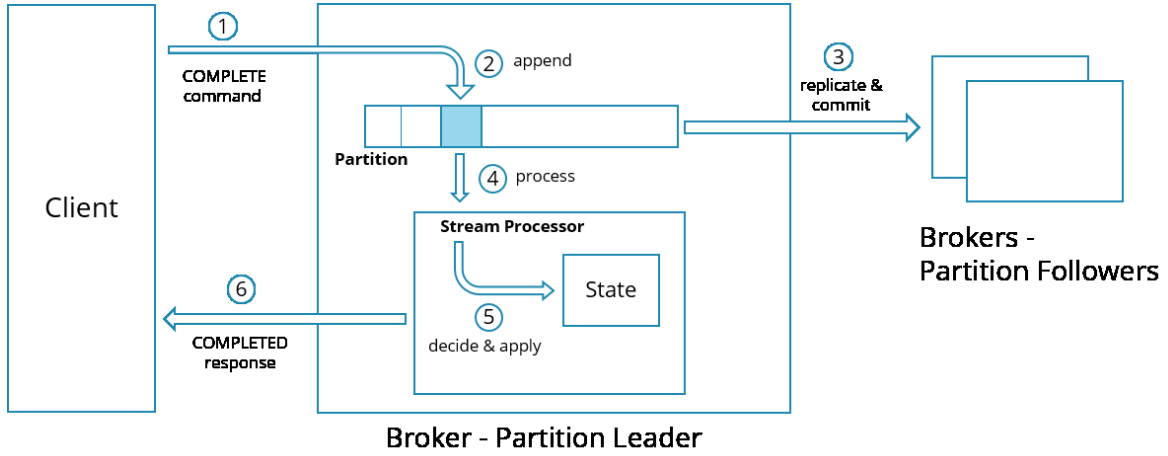
Modèle opensource avec des extensions propriétaires

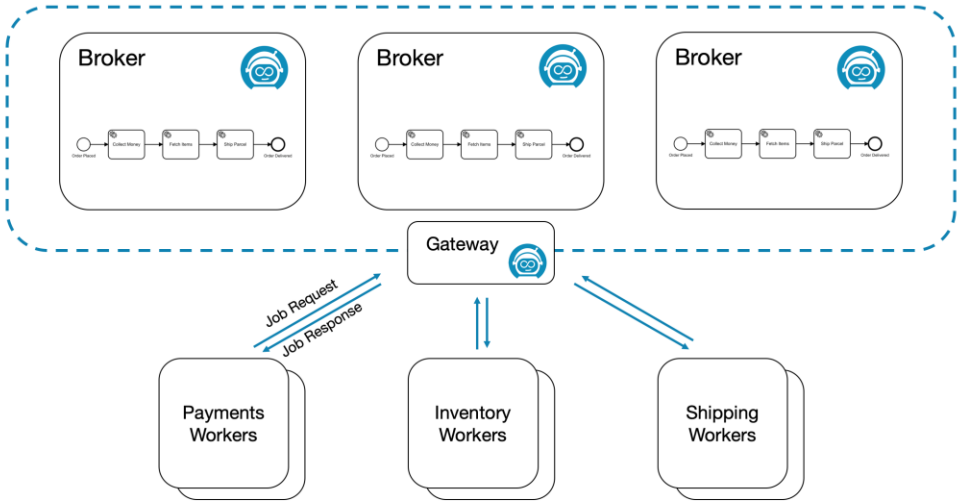


Intégré dans l'offre Camunda Platform 8 – Déploiement selon un modèle SAAS ou On Premise

Architecture Zeebe





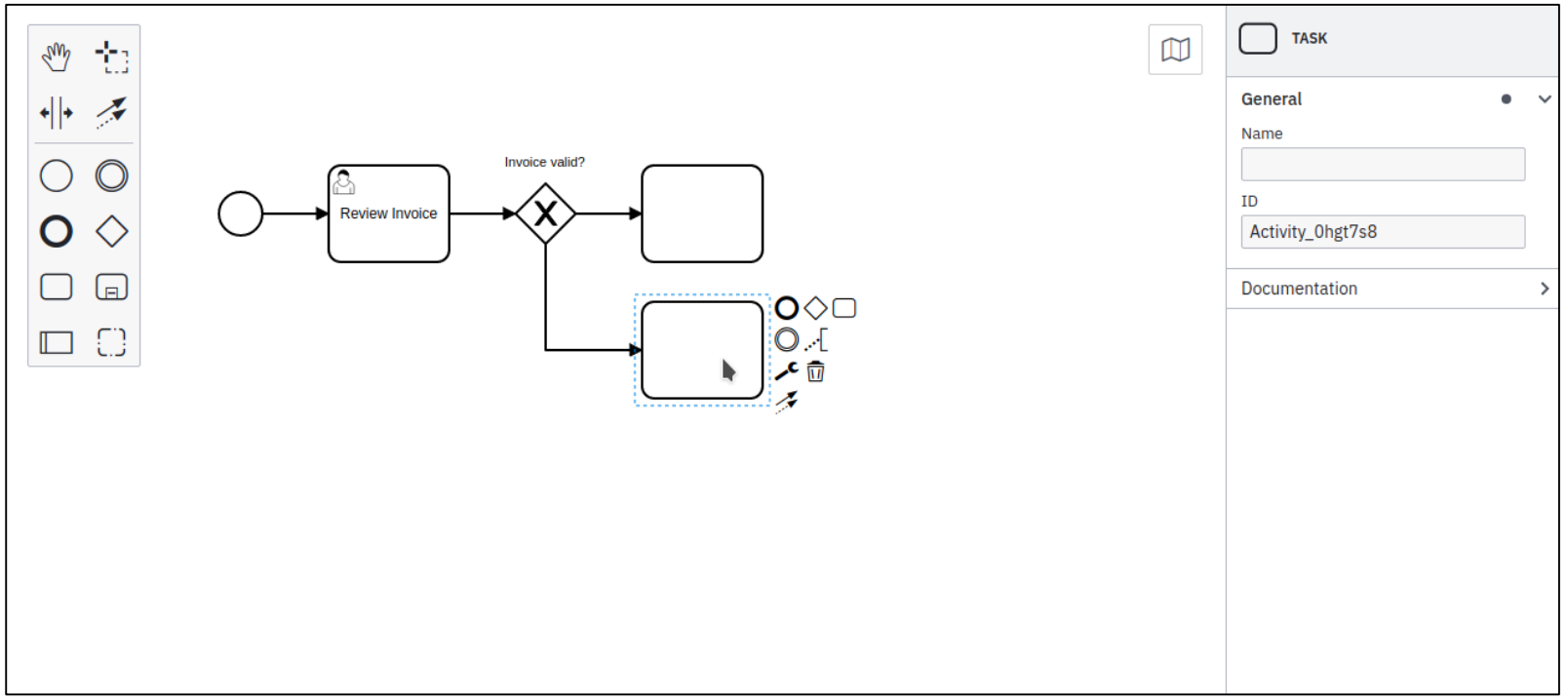


Modèle d'architecture scalable et HA inspiré de Kafka :

- Configuration du nombre de partitions et de réplicas
- 1 partition est un flux ordonné d'événements de workflows
- Chaque Workflow est associé à une partition
- Distribution des partitions sur les brokers
- Distribution des jobs aux différentes instances de workers

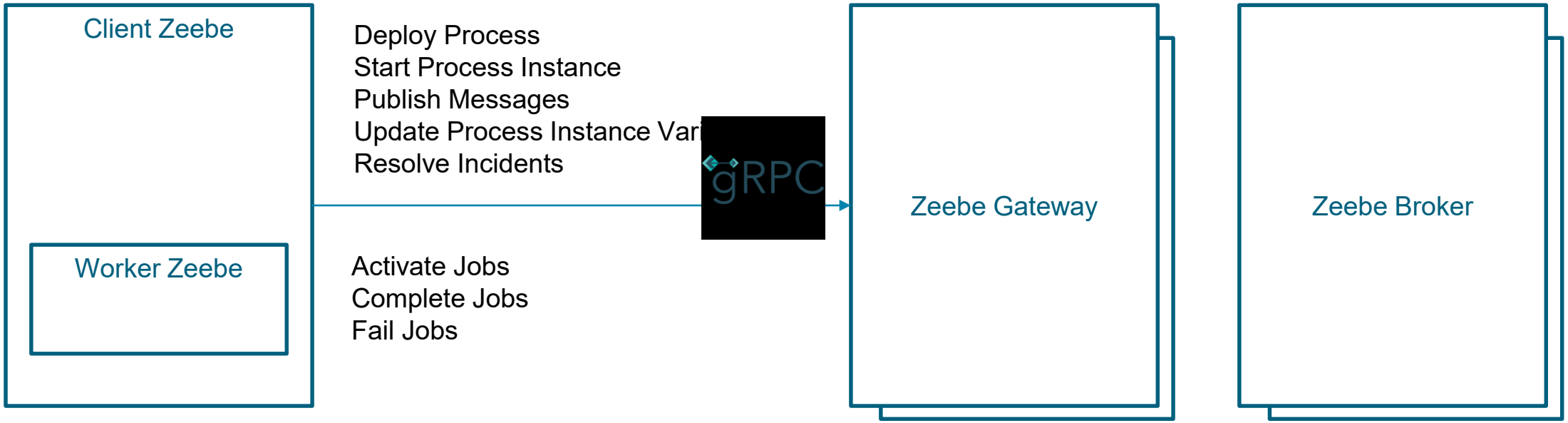
Workflow BPMN 2.0

- Support partiel de BPMN 2.0 (voir https://docs.camunda.io/docs/components/modeler/bpmn/bpmn-coverage/)
- 2 modelers disponibles
  - WEB (Camunda Platform seulement)
  - Desktop



Clients et Workers Zeebe

- **Client Zeebe** : Application qui communique avec Zeebe Broker via GRPC
- **Worker** : Client Zeebe qui implémente une ou plusieurs activités de workflows



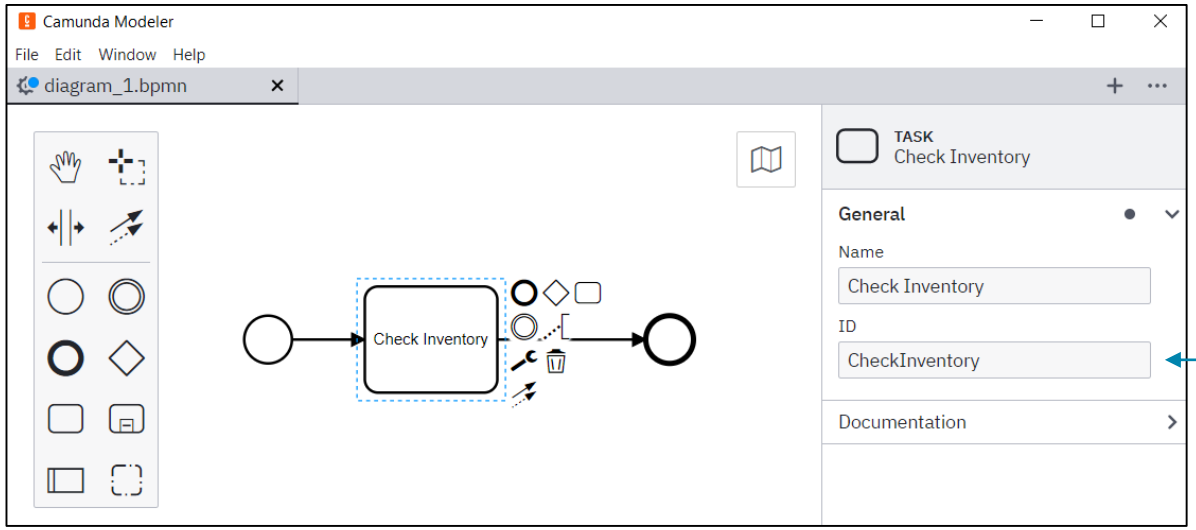
Official Clients
- Java
- Go
- CLI

Third Party Clients
- C#
- Javascript/NodeJS
- Micronaut
- Python
- Ruby
- RUST
- Spring

La scalabilité est portée par :
- **Le nombre d'instances de Brokers**
- **Le nombre d'instances de Workers**

Principe d'implémentation d'un worker

- Pattern Publish/Subscribe basé sur une approche long polling

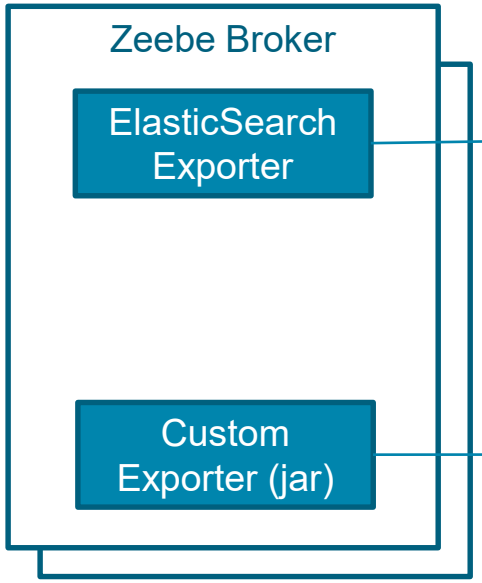


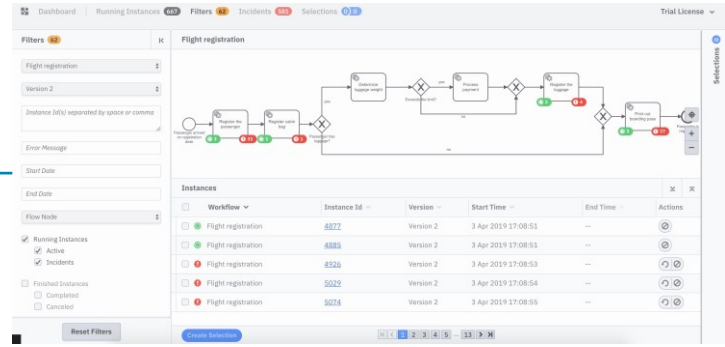
Exemple d'implémentation avec le SDK Zeebe Spring

```
@ZeebeWorker(type = "CheckInventory")
public void handleJobFoo(final JobClient client, final ActivatedJob job) {
  // do whatever you need to do
  client.newCompleteCommand(job.getKey())
      .variables("{\"inventorySize\": 100}")
      .send();
 }
```

Exporter API

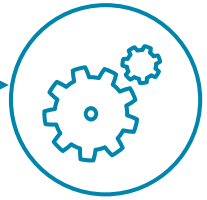
- Mise en œuvre du pattern CQRS (Command Query Responsibility Segregation)
  - GRPC : Commande des workflows
  - Exporter API : Requête sur les états de workflows





Zeebe Operate

```
public class CustomExporter implements Exporter {
  public void configure(Context context) { }
  public void open(Controller controller) { }
  public void close() { }
  public void export(Record record) { }
}
```



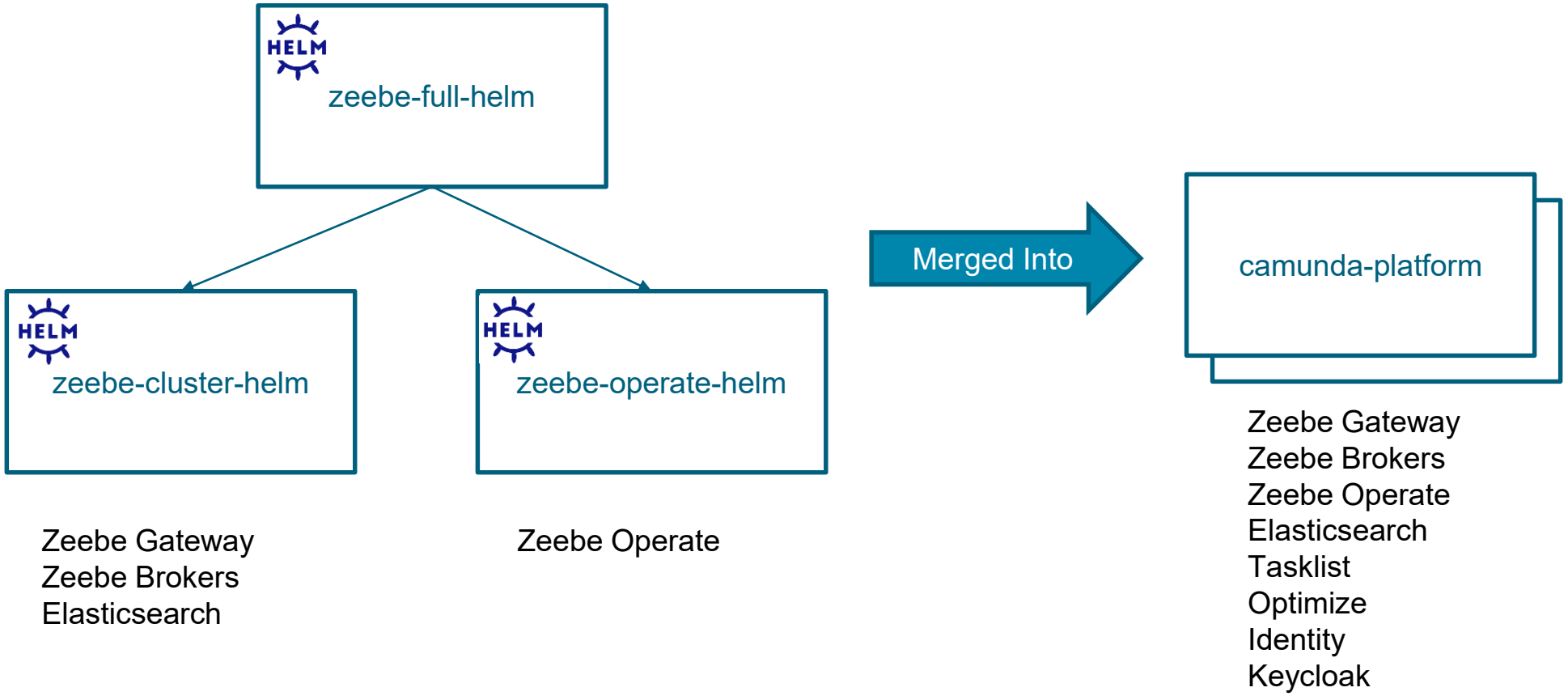
Système Tiers

Méthode appelée par le Broker pour chaque événement Zeebe :

- Activation/Terminaison d'un Workflow, d'une activité
- Création d'une variable
- Création/Suppression d'un incident
- …

Déploiement Kubernetes

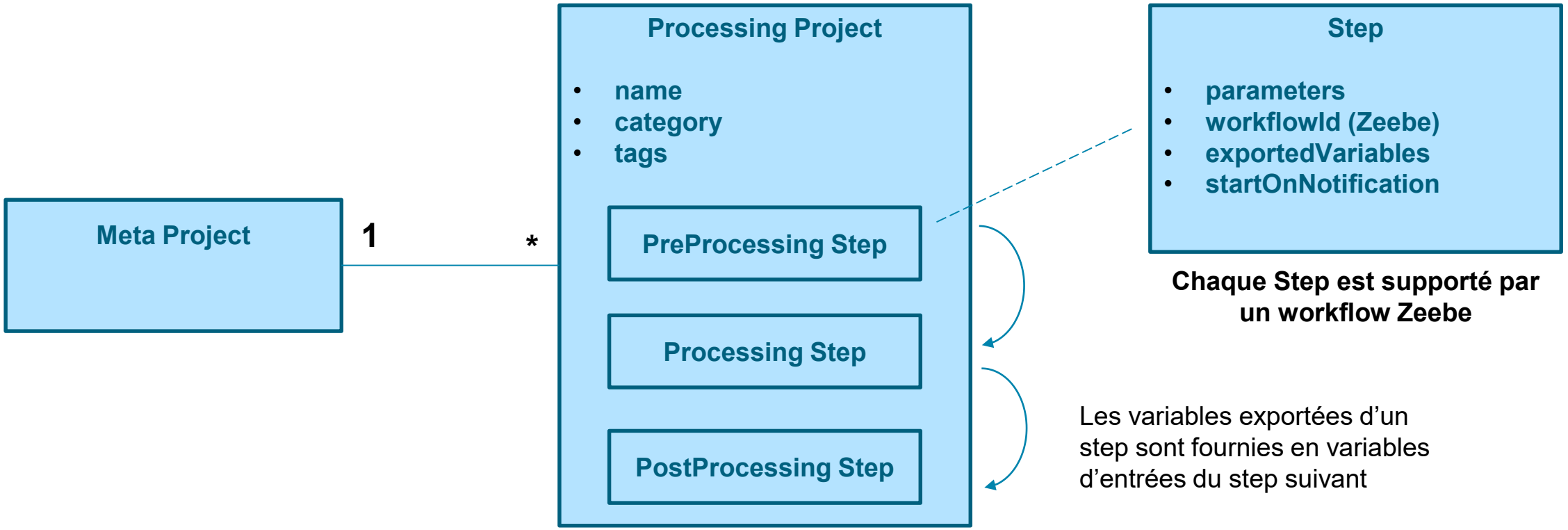
- Disponibilité de Charts HELM permettant de déployer rapidement un cluster Zeebe dans un environnement de production



Mise en oeuvre de Zeebe pour CO3D

Capgemini AIRBUS

Modèle d'abstraction des traitements CO3D



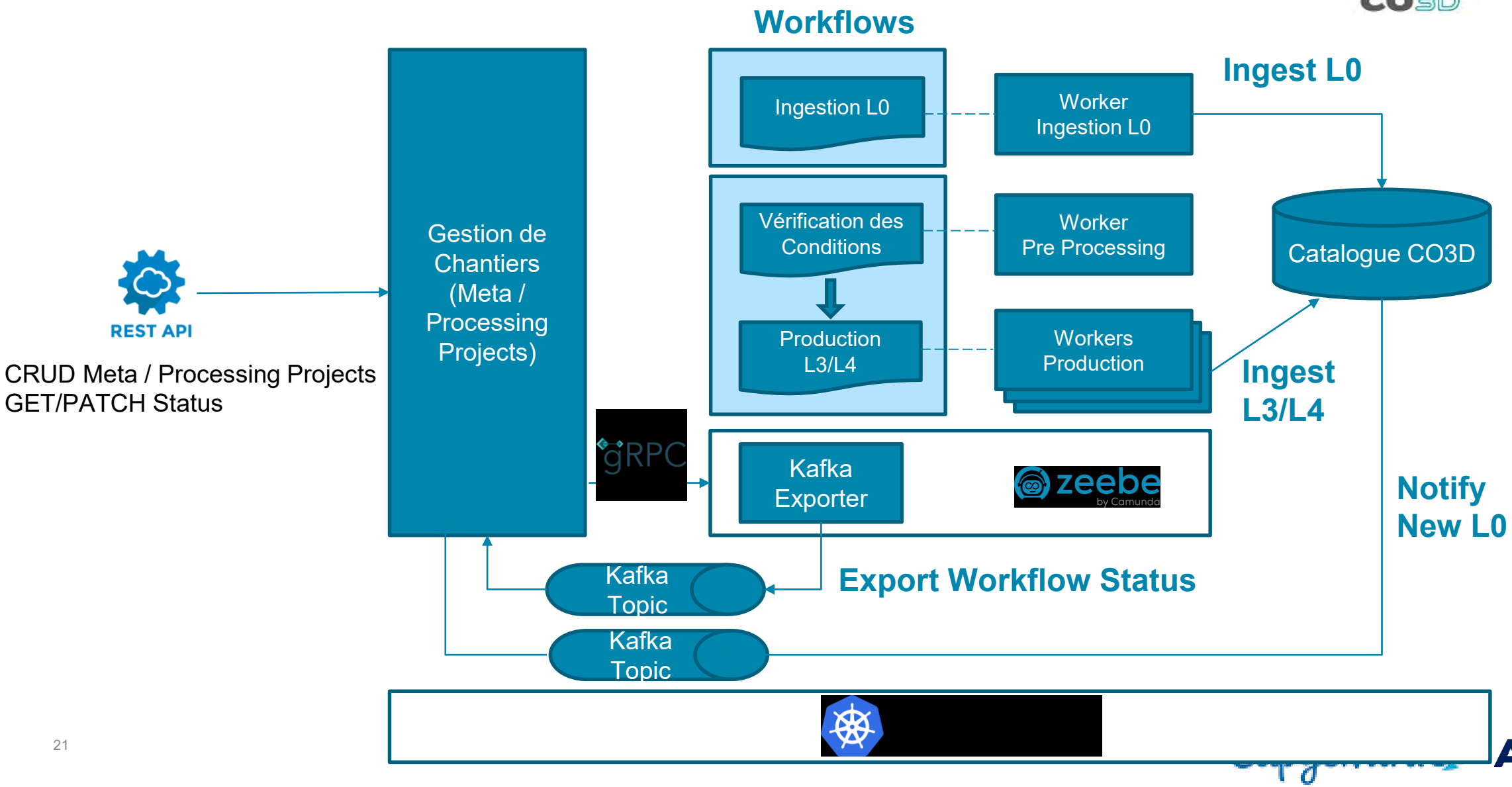
Chaque Step est supporté par un workflow Zeebe

Les variables exportées d'un step sont fournies en variables d'entrées du step suivant

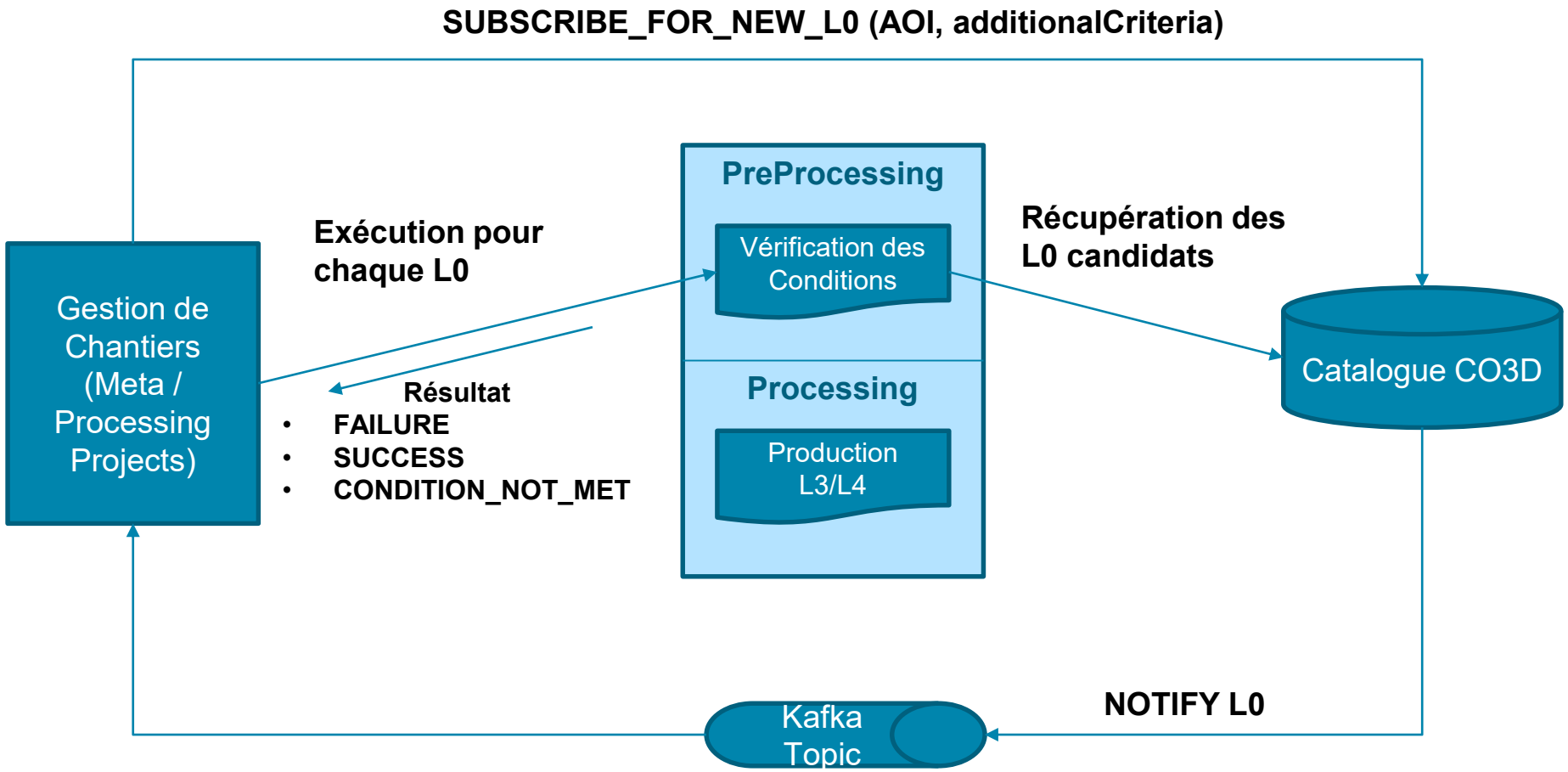
Chantier L4

Production L3
Production Dalle L4
Ingestion L0

Architecture orchestration de traitements CO3D



Intégration des Productions L3/L4 avec le catalogue CO3D



Conclusions & Next Steps

Capgemini AIRBUS

Retour d'expérience Zeebe pour CO3D

Bénéfices	Points de vigilance
• Maturité / « production ready » • Concepts rapidement assimilables • Déploiement facilité et développement rapide • Intégration et adaptabilité dans un système plus large • Pérennité via le support/maintenance Camunda	• Modèle de licensing peu souple • Pas de prise en charge du déploiement des workers • Pas de gestion des ressources ni des priorités • Pas d'API de Query native en OSS • Perte de lisibilité de la roadmap suite à l'intégration dans Camunda Platform