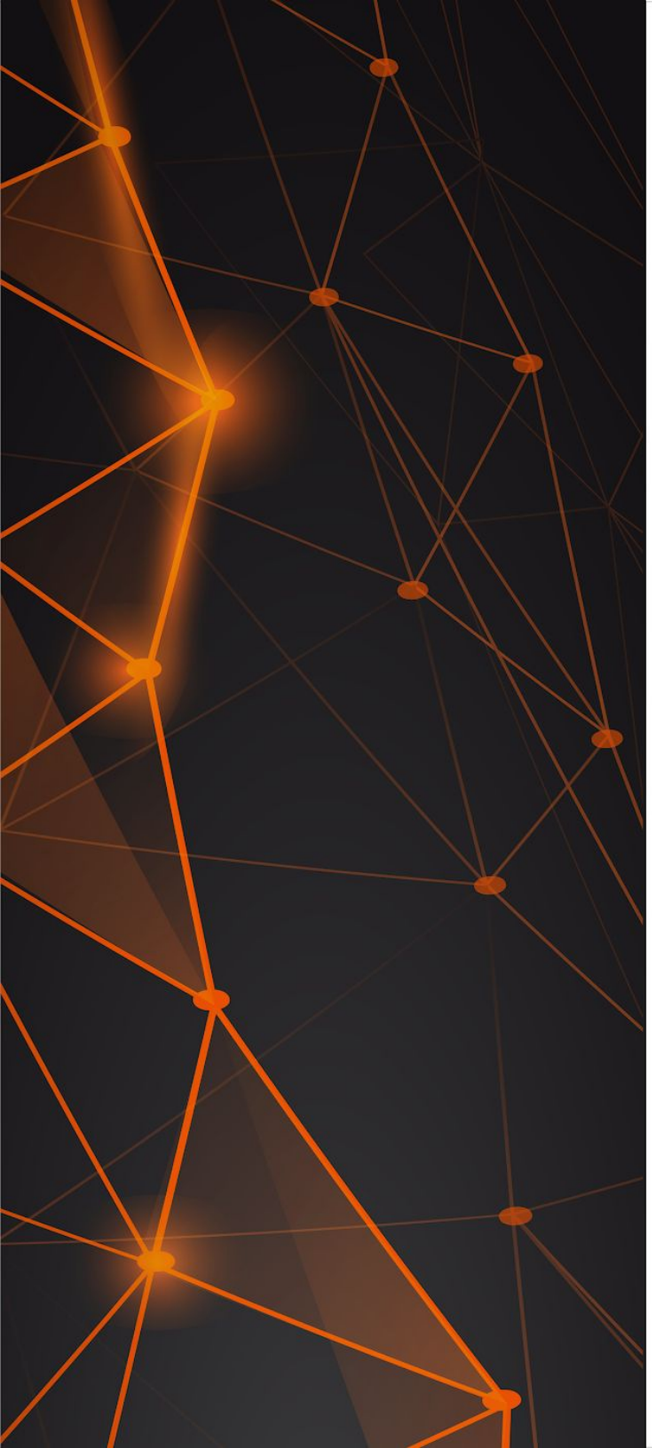
CAMUNDA CON LIVE

Enabling Core Banking Use Cases with Camunda Cloud

From Building the next Generation of a Payments Platform using Camunda Cloud to deliver a Firmwide Generally Available Platform for Microservices Orchestration



CAMUNDA CON LIVE

Agenda

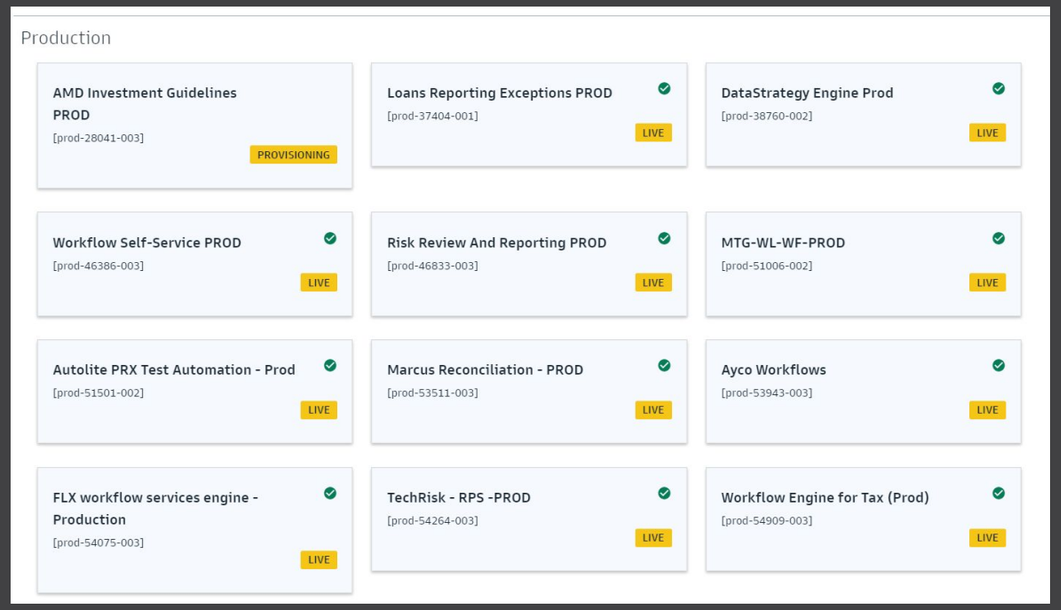
- Goldman Sachs experience with Camunda
- Why a new Platform based on Camunda Cloud
- New Platform Vision
- Camunda Cloud Extensions
- Architecture Overview
- Questions?

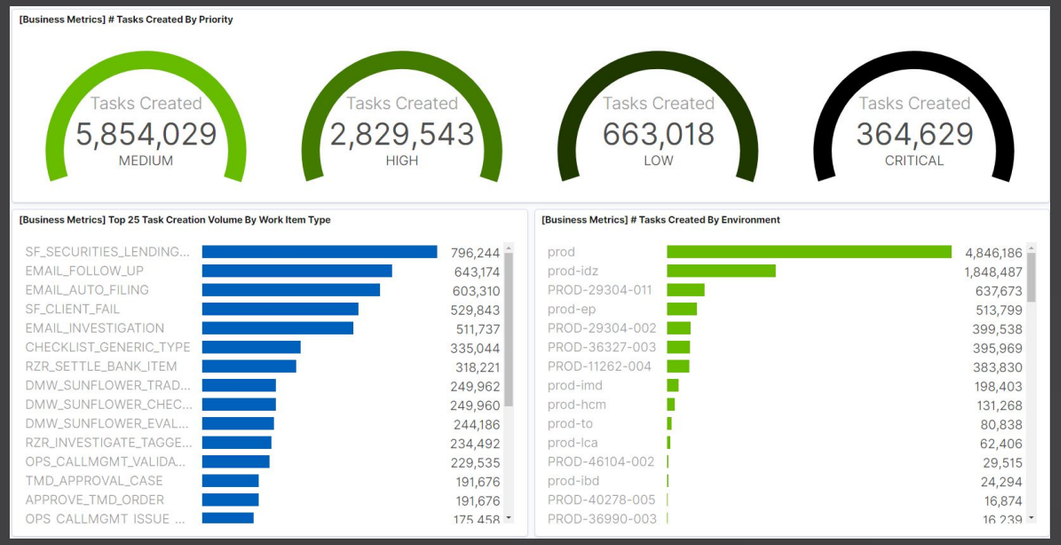
Marcus: BY GOLDMAN SACHS® | work flow | CAMUNDA

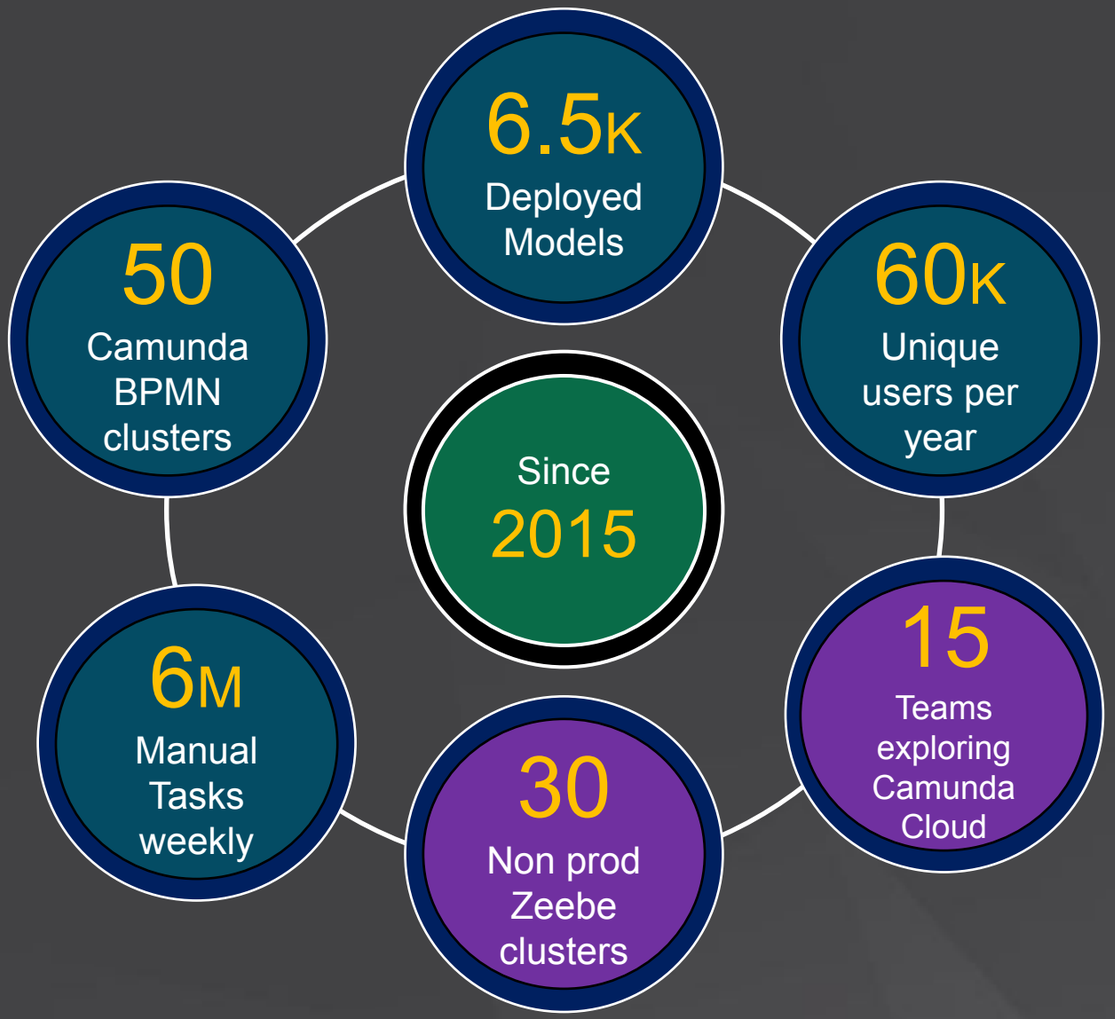
Goldman Sachs | we BUILD

Some statistics

Up and Running at scale since 2015







New Platform Journey

Exploration > Inception > Construction

CAMUNDA CON LIVE

Before 2020

Exploration	Camunda BPMN clients start using our Platform for Microservices Orchestration	Some clients start questioning the capacity of our platform to cope with these new use cases	Workflow Team explores alternatives and Zeebe becomes a candidate

Early 2020 Payments decides to build a new Platform and needs an Orchestrator

Inception	Access to the platform and it s data must be secured using single sign on, authorized and encrypted	Platform needs to process expected volumes satisfying a minimum throughput and maximum latency	Platform operates 24x7 and can recover from a regional failure in 15 minutes without data loss	Platform is observable, offers metrics and monitoring, logs centralization and distributed tracing

Q2 2020 – We decide to build a new GA Platform based on Camunda Cloud.

Construction	Prometheus metrics and alerting with probers	Alert automation for self healing and self protection	Cluster provisioning integrated with Terraform	Platform runs on premises and any desired cloud
	Automated data retention, data purging, data backups and lake integration to enable business analytics	Disaster recovery and clusters upgrades are automated and exposed to administrators via graphical user interface	Integration with manual tasks and forms using our current proprietary tooling available for Camunda BPMN	User experience (SDLC) remains the same for users targeting Camunda BPMN or Camunda Cloud

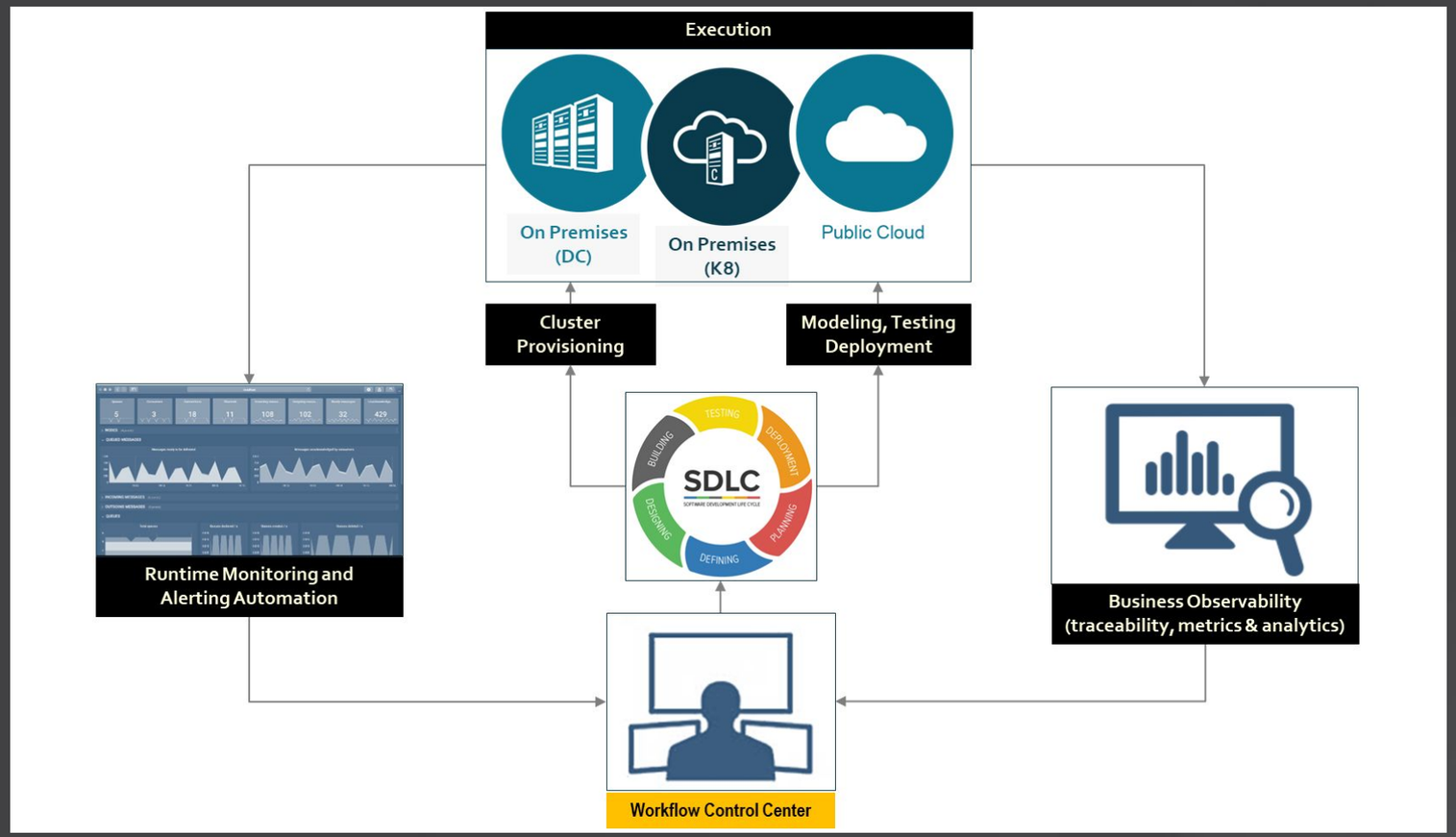
Our Vision

Model Once, Run Anywhere

CAMUNDA CON LIVE

Vision

Workflow Control Center Portal centralizes all required operations to provision, model, deploy, execute and monitor any workflow enabled application. Selected runtime is managed in an homogeneous way decoupling infrastructure and business.



Goldman Sachs Extensions

Custom Camunda Cloud extensions to satisfy non-functional requirements

CAMUNDA CON LIVE

Catalog of extensions and used technologies

Performance Testing

Jmeter
Spring Boot
Kafka
Elastic Search
Moustache

Security

Changes in
Zeebe Client
Gateways
Operate
(data encryption)

Disaster Recovery

Kafka Connect
Elastic Search Audit Trail
Enhanced Zeebe Client
Control Plane

Observability

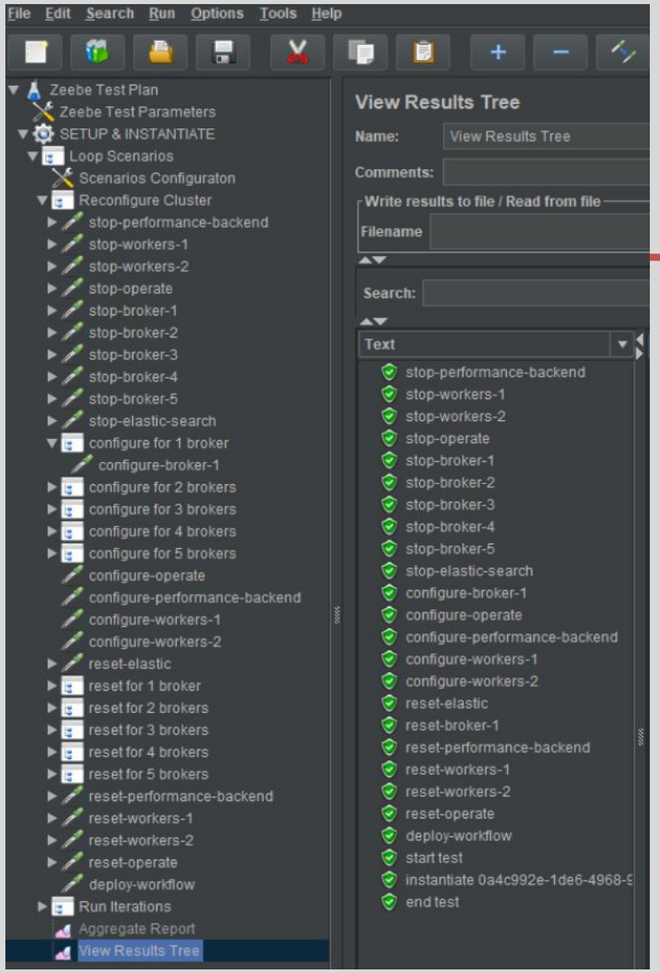
Open Telemetry
Log Centralization Filebeat
Prometheus alerting
Web hooks for Kafka
Probers

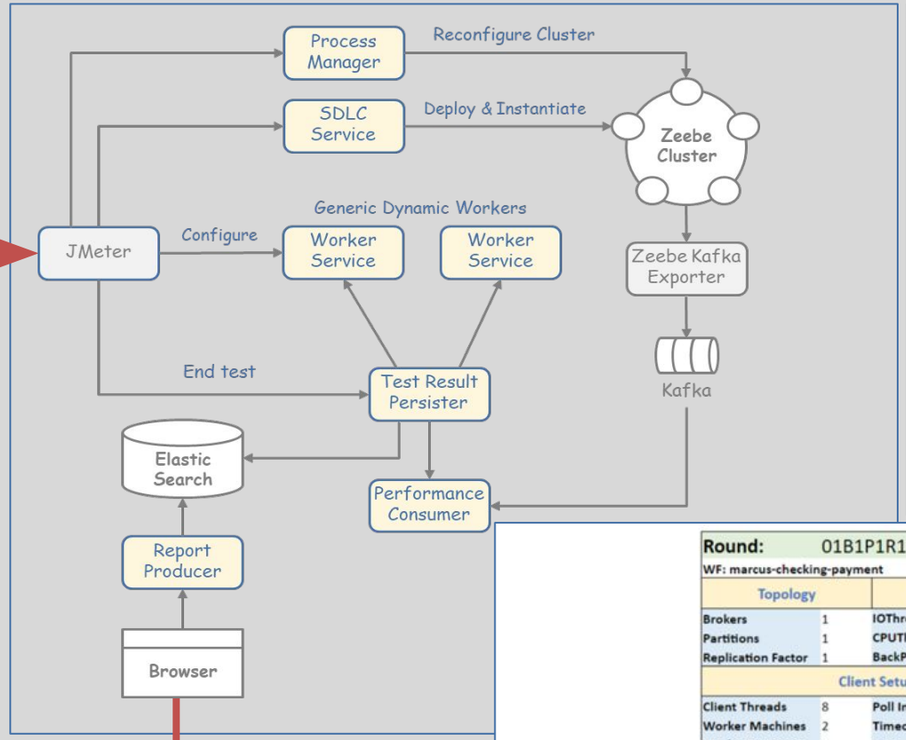
Deployment Automation

Custom Containers
Podman
(Kubernetes)
Terraform

Performance Testing

Mock any workflow model and execute it under load on different cluster topologies






```
workflows:
  - name: payments
    workers:
      - name: validate-payment
        instances: 1
        action: ACCEPTED
      - name: notify-accepted
        instances: 1
      - name: notify-rejection
        instances: 1
      - name: creation-time-controls-request
        instances: 1
        event: creation-time-controls-response
        eventAction: CLEARED
      - name: schedule-request
        instances: 1
        event: release-message
      - name: transmission-time-controls-request
        instances: 1
        event: transmission-time-control-response
        eventAction: CLEARED
```

Round: 01B1P1R1I2C2

WF: marcus-checking-payment

Topology		Brokers Setup	
Brokers	1	IOThreads	2
Partitions	1	CPUThreads	2
Replication Factor	1	BackPressure	VEGAS
Client Setup			
Client Threads	8	Poll Interval	100
Worker Machines	2	Timeout	10
Worker Instances	2	Join	true
Max Jobs Active	100	Instantiation Delay	200

Topology & workers configuration

TEST RESULTS

INSTANCES	DELAY		PARTITIONS 1	1	1	1
1	0	T	2073	1330	1360	1573	1473
		W	364	258	293	481	425
		Z	1709	1072	1067	1092	1048
2	0	T	2097	1638	1726	2210	2092
		W	360	301	336	651	642
		Z	1737	1337	1390	1559	1450
3	0	T	2201	2032	2055	2979	2654
		W	328	359	378	691	637
		Z	1873	1673	1677	2288	2017
4	0	T	2369	2274	2483	3411	3374
		W	326	397	436	710	725
		Z	2043	1877	2047	2701	2649

How many workflows we run in the test

Forced Worker delay

One test
- Total Workflow execution time (avg)
- Workers time (avg)
- Zeebe broker time (avg)

Link to test details

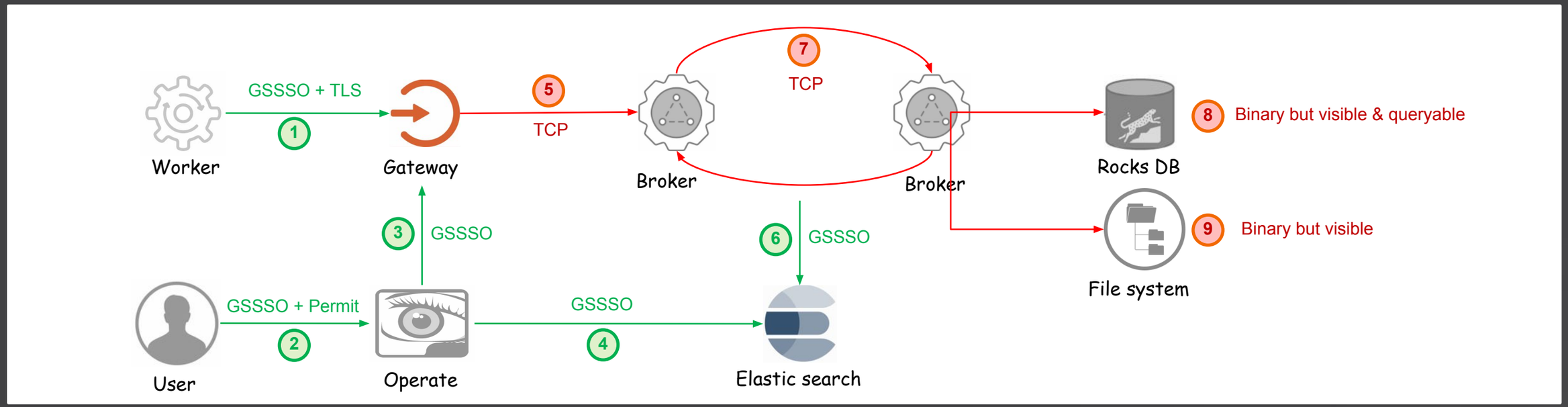
Link to Prometheus details

Security Enhancements

Security Extensions to authenticate, authorize, protect sensitive data and avoid data tampering

CAMUNDA CON LIVE

Areas where security has been improved



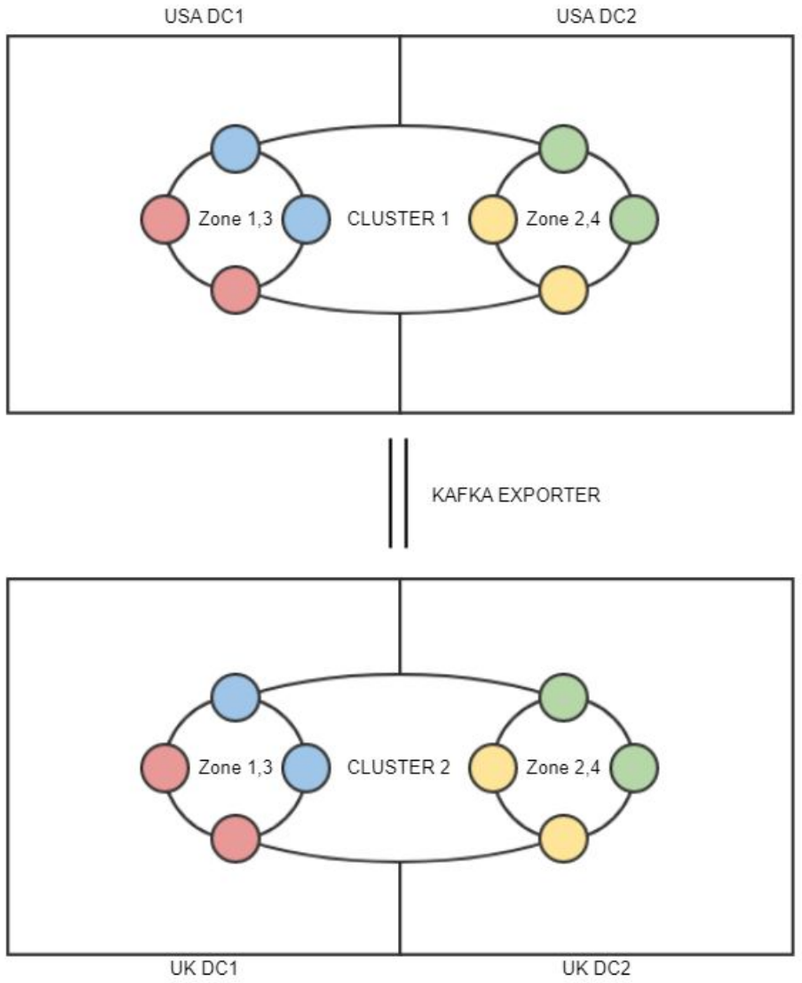
Zeebe Client connects to Single Sign On and obtains a token	Operate and gateways validate the token and authorize the calls	Operate Login page replaced by SSO	Operate passes a token to elastic search that is also secured	Operate passes a token to Kafka that is also secured
Zeebe Client extended with extra operations to encrypt/decrypt	Encryption module is pluggable and controlled by clients	Special variables in payload store a signed hash of the payload	Client validates payload before delivering to workers	Authorization is not cached for instant access revoke

Disaster Recovery

Leveraging exporters to execute in flight workflows out of region

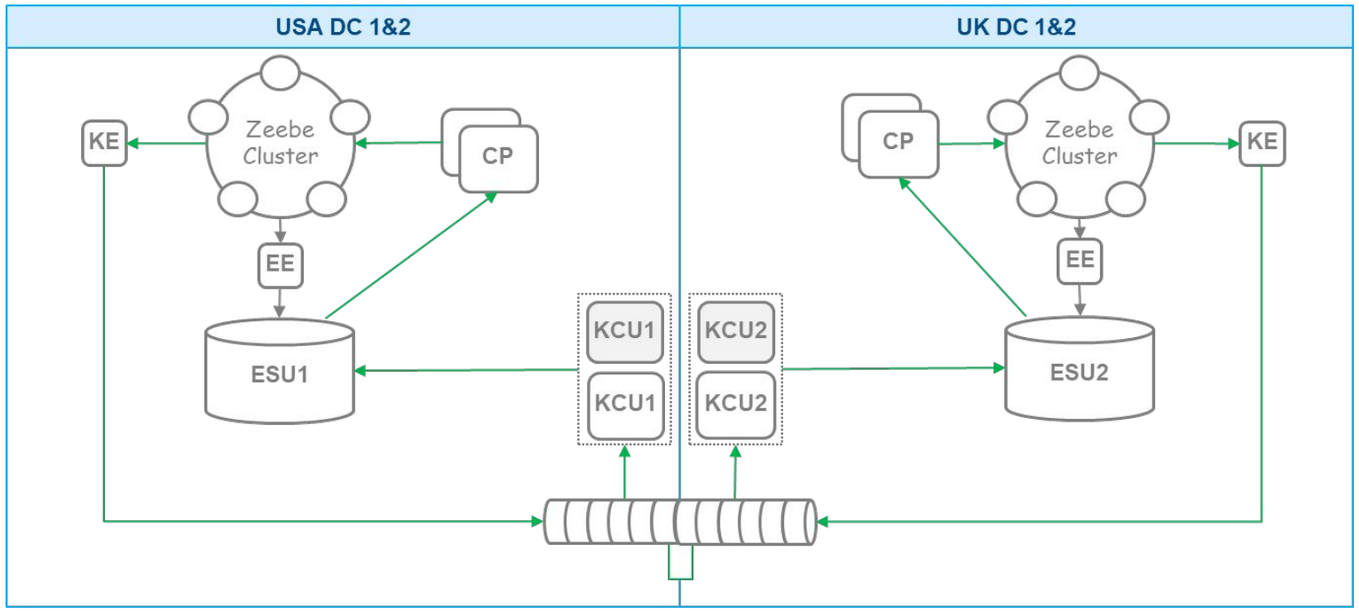
CAMUNDA CON LIVE

Predictable Replication



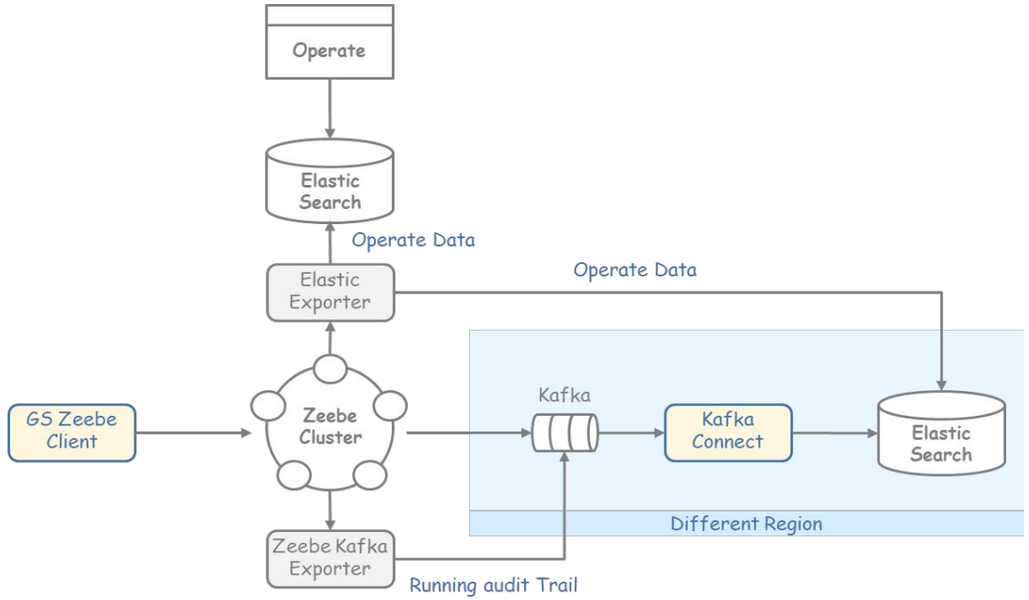
Active – Passive cluster to minimize Latency

- Leverage Zeebe Kafka Exporter to send events produced by Zeebe to Kafka
- Use Kafka Connect to read events and build an audit trail in Elastic Search that captures the execution steps of active workflows
- Use a control plane to re-instantiate workflows that were not completed in the passive cluster
- Use extended Zeebe Client to respond on behalf of workers to move workflows forward as much as possible.



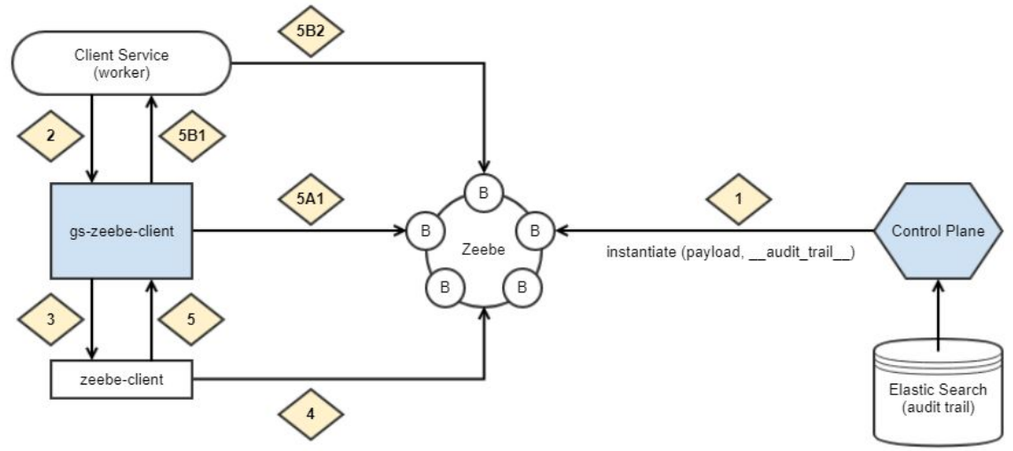
Disaster Recovery

Client enhancements to respond on worker's behalf



Observed Events

Operation	Description
Deploy workflow	Deploys a workflow definition in Zeebe. It will create a new version of the last deployed one
Instantiate Workflow	Creates a new instance. This is the first time we send a payload to the cluster with variables and values
Complete/Reject Job	When Zeebe notifies workers and workers complete or reject the work. We can send more variables here.
Send Events	And external system interacts with Zeebe using an event name and a correlation key in the payload. The workflo
Cancel workflow instance	We hard-stop an instance
Change payload variables	That normally is done via Operate. But payload can be changed programmatically. This is useful when we are res
Resume workflow (after incident)	A workflow instance that becomes an incident will stop but it can be resumed.

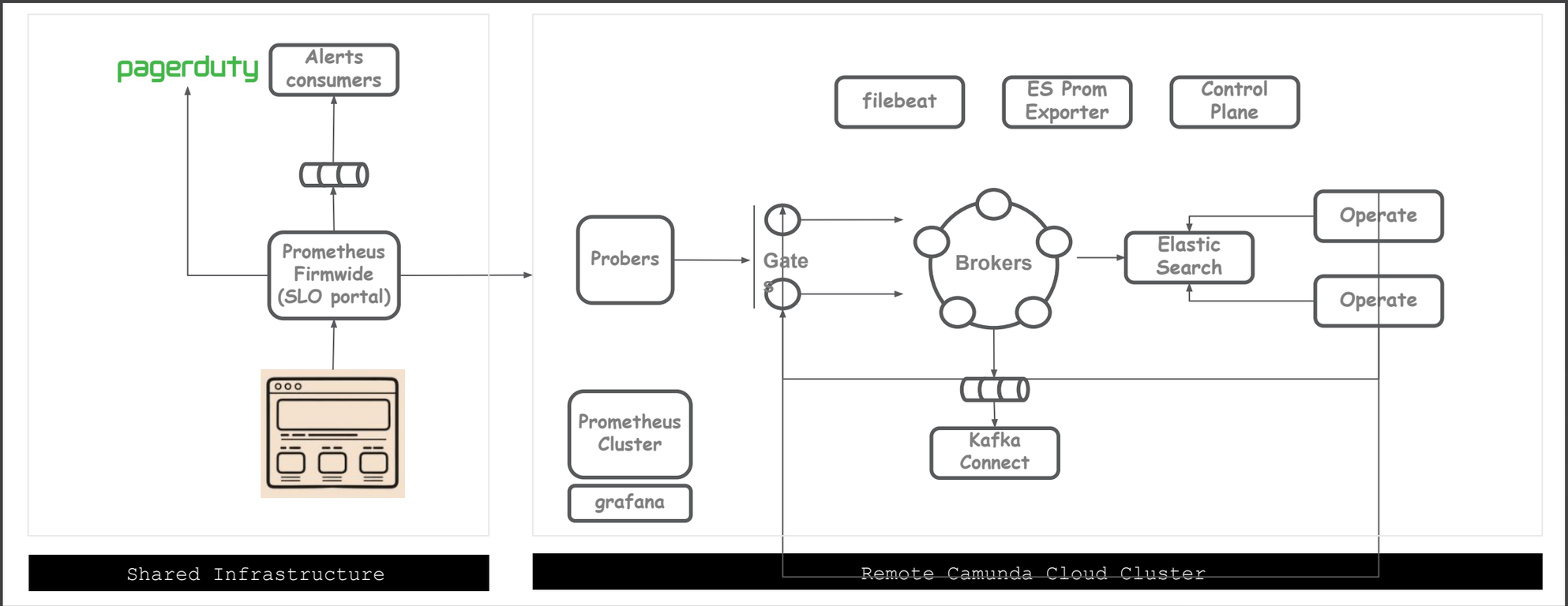


Client Mechanics

Step	Description
1	• Control plane instantiates a workflow in the recovery cluster • We pass the original payload and a special variable called __audit_trail__ with the steps and variables that represent the execution of the original workflow
2	• Any worker (provided by the client) polls for work using the gs-zeebe-client (written by workflow team)
3	• gs-zeebe-client delegates on standard zeebe-client to poll for work
4	• zeebe client returns work to do that contains the original payload and the audit trail
5	• gs-zeebe-client checks if we have in the audit trail an step that matches with the worker name
5A1	• If there is a match the gs-zeebe-client responds to Zeebe (and client service is not notified for more work to do)
5B1	• If there is no match, gs-zeebe-client passes the work to do (the job) to the client service (the worker)
5B2	• Client service (worker) responds to Zeebe

Observability

All Platform components are observable. Data is centralized



All components expose metrics and centralized their logs

SLI/SLO/Alerting is publicly available.

Runtime Monitoring is cluster specific and low level

Distributed tracing only visible at client level

All data points aggregated in workflow control center

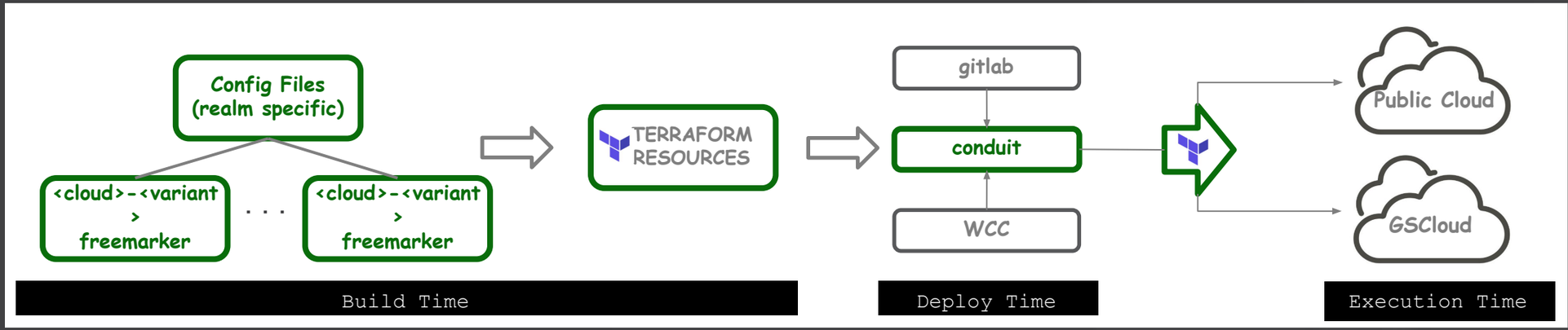
Deployment Automation

Moving from a centralized model to a managed service

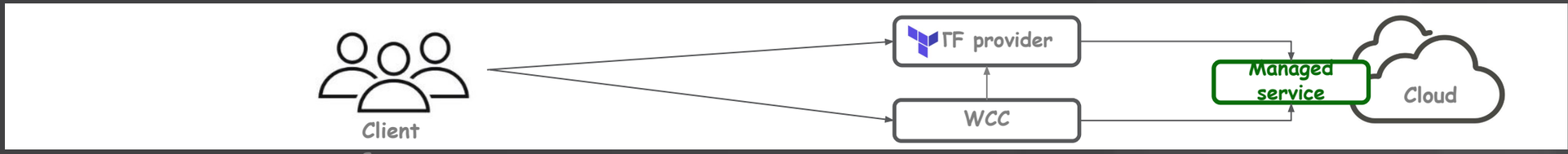
CAMUNDA CON LIVE

Requirement	Meaning
Multi Client	Ability to provision clusters for multiple clients
Multi Environment	Clients need more than one cluster
Multi Cloud	Clients want to run the platform where their services are
Different Variants per Cloud	Not every client wants to run in the same way
Different Cluster Topologies	Not all components are always deployed

Current



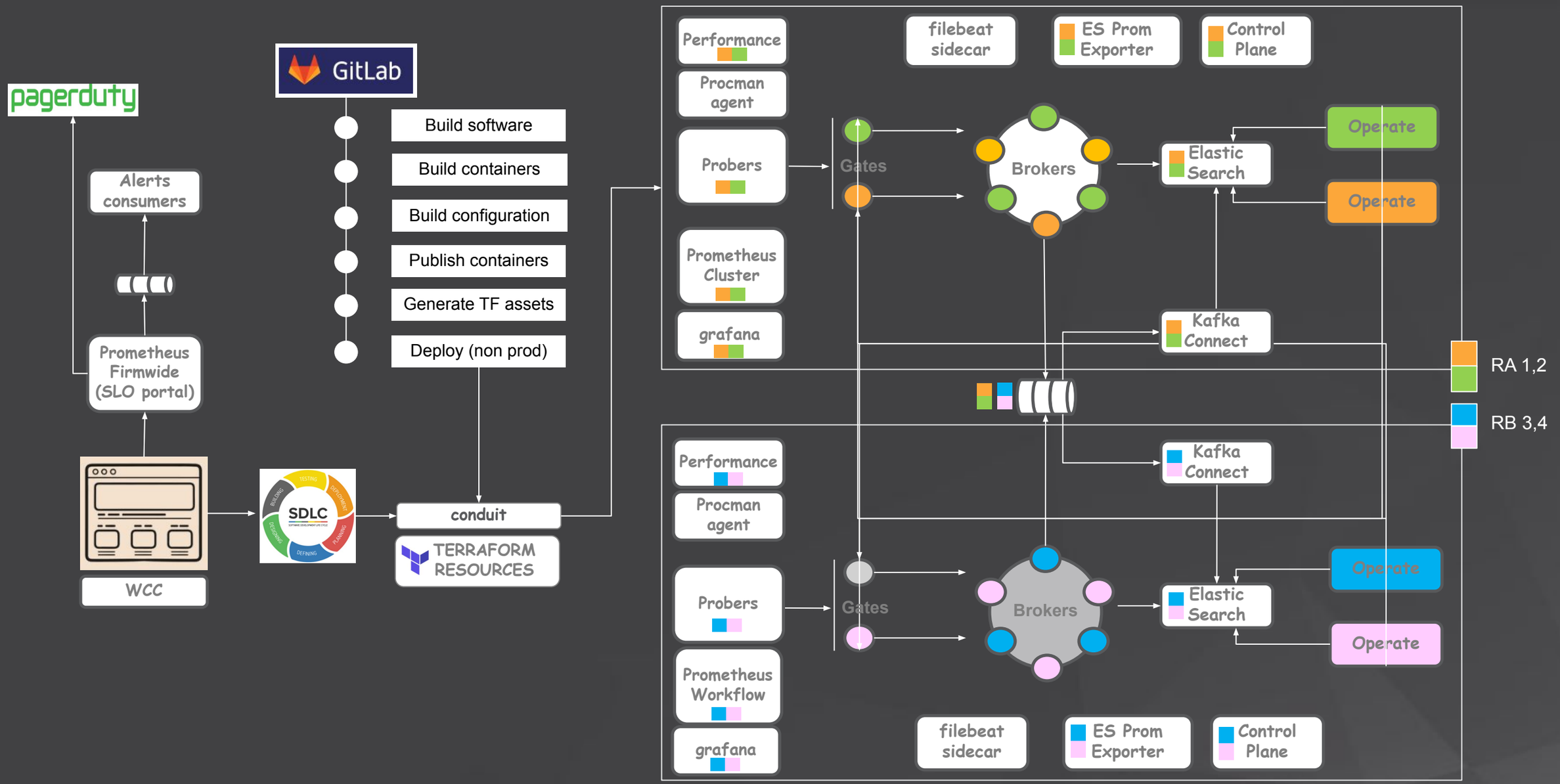
Next



Terraform + Custom Containers + Camunda Cloud Console

Full Architecture

Platform Current Topology



CAMUNDA CON LIVE

Questions?

Thanks - Javier Sabino