EOM (ECOMP Operations Manager)

Overview for ONAP Community

Habib Torab
Shrikant Acharya
Vijay Venkatesh Kumar

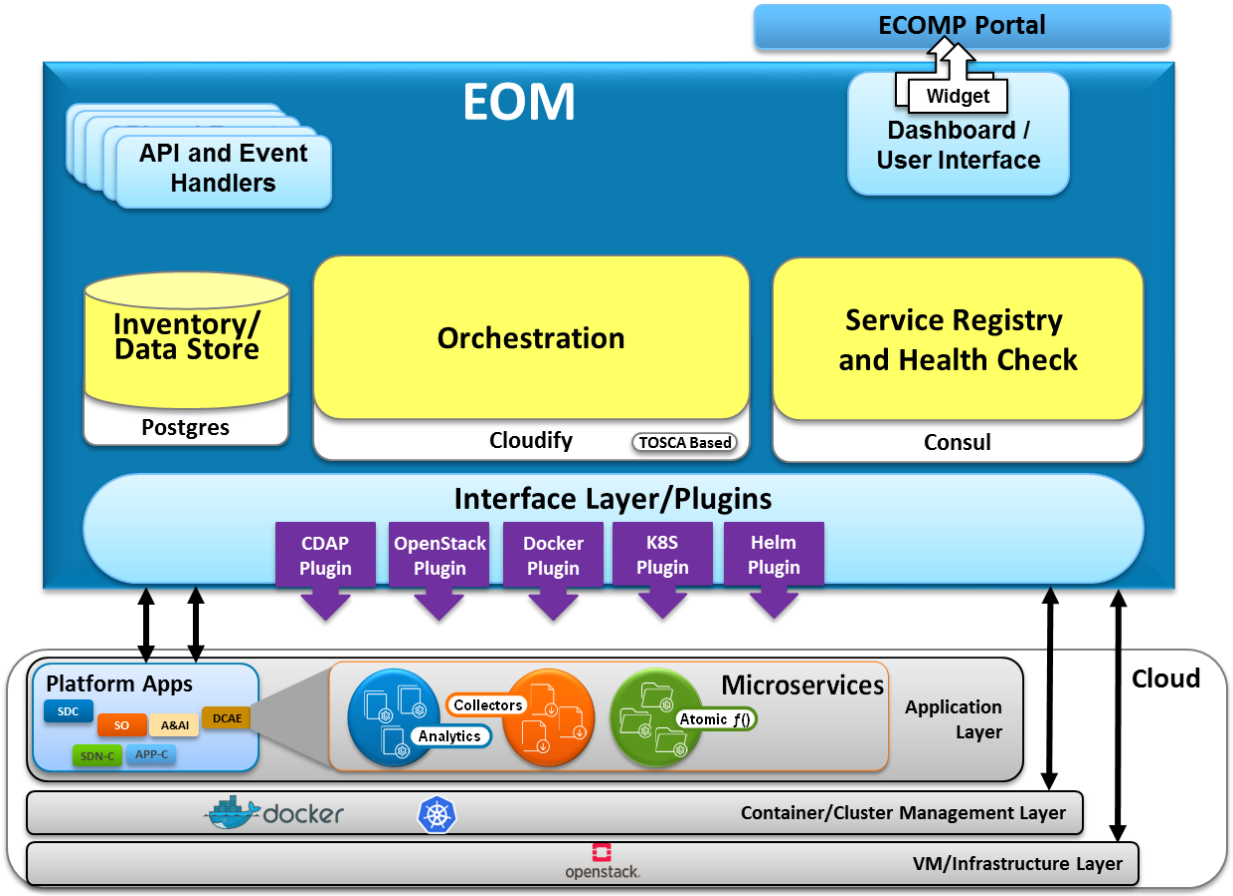
February 20, 2019

Topics

- Background
- What is EOM?
- Architecture
- Features highlights
- Edge Automation requirement coverage
- Delta compared to ONAP DCAE

Background & high level architecture

- The framework was initially designed and built for DCAE.
- It was expanded later and the same framework was used as the main “platform controller”.
- Initially 2 separate instances were used in AT&T: Root Node/ECOMP Controller & DCAE Controller
- Root Node Controller was used to manage all platform applications with the exception of DCAE applications/mS
- The architecture is centered around a top-level TOSCA-based orchestration using Cloudify.
- It leverages Cloudify plugins and supports deployment/management of platform workloads in different cloud environment
- An AT&T contributed Helm plugin is used for helm-based deployments of core platform applications.
- This allows leveraging the helm charts created for platform applications in ONAP.
- Consul is another key subcomponent used for registration/discovery, health checks, and as a key/value store.

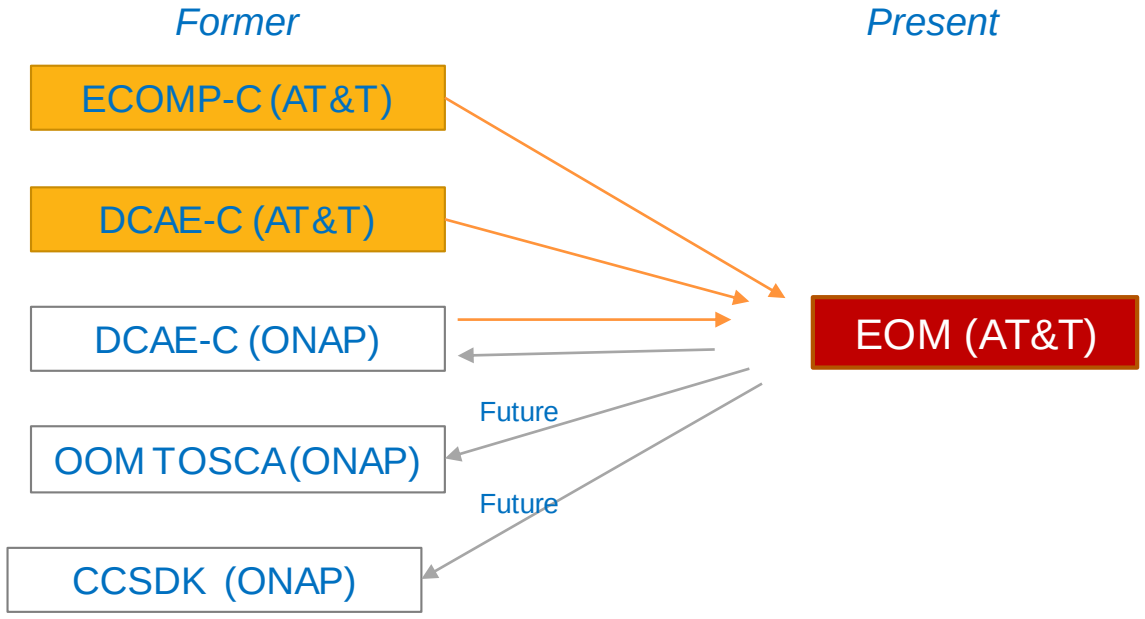


What is EOM?

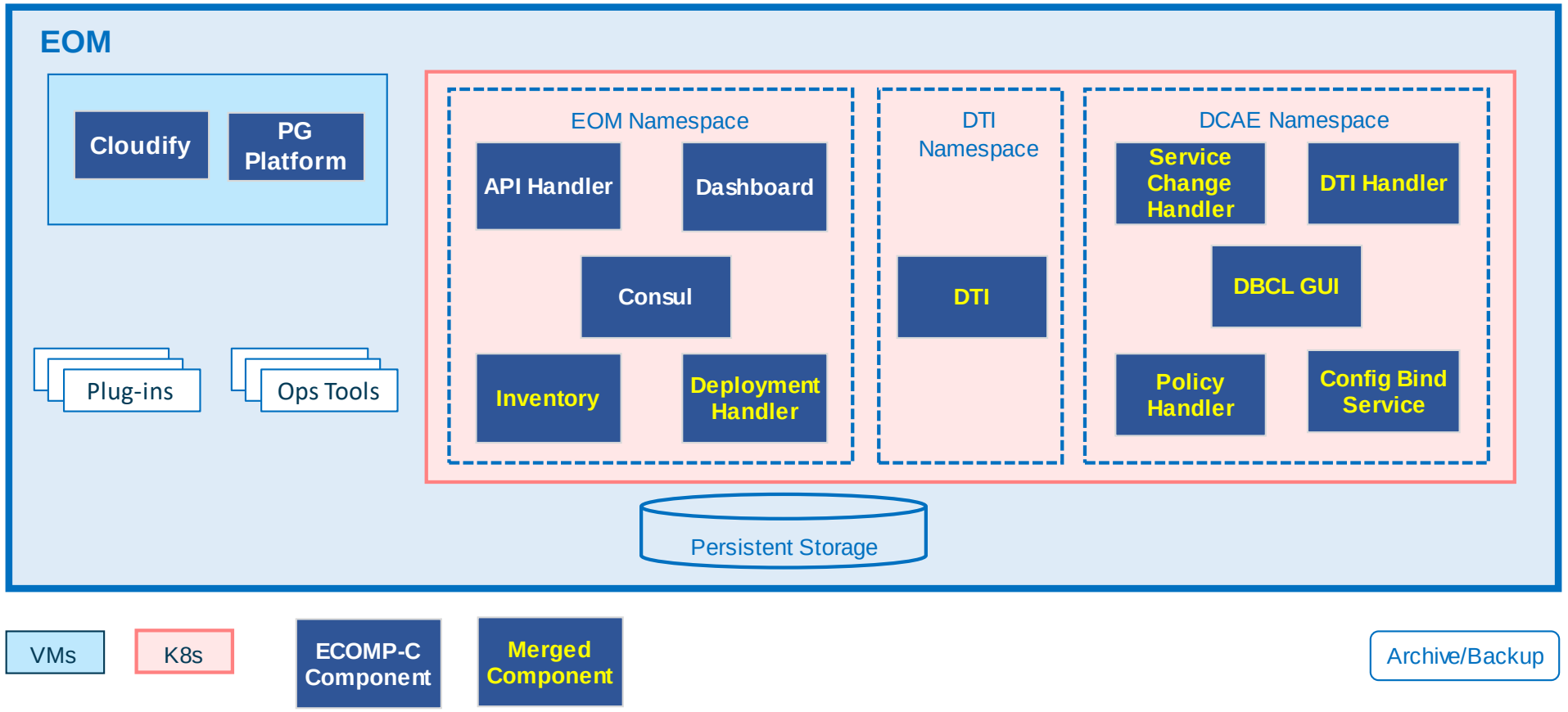
- EOM: ECOMP Subsystem that deploys and manages the lifecycle of ECOMP and all software required to make it operational, including scaling and self-healing
- EOM merges the functionality of:
  - ECOMP-C developed by AT&T
  - DCAE-C developed by AT&T
  - DCAE-C from ONAP
  - ONAP OOM (TOSCA) & HELM Plugin
  - CCSDK (ONAP)
- Single code base
- Continuous process of insourcing/open-sourcing between ONAP and EOM

Motivation

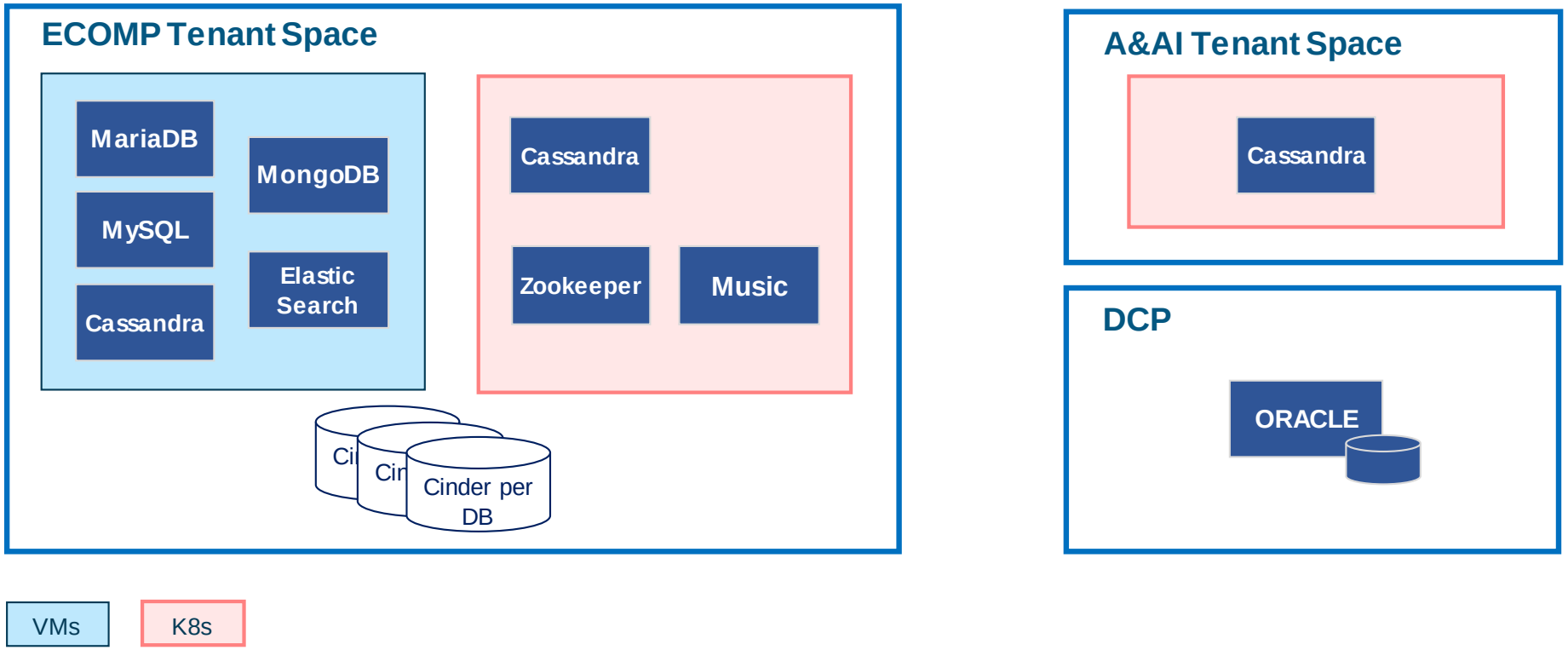
- Three existing controllers, DCAE-C (ONAP), DCAE-C and ECOMP-C, shared several attributes:
  - Same root source code
  - Common platform components across multiple implementations
  - Similar functions: Onboard and manage microservice lifecycles
  - Maintaining two similar systems was inefficient in engineering & development effort, which had to be duplicated for both systems in many cases, and resource staffing
- Decision: Merge controllers



EOM components



DBaaS



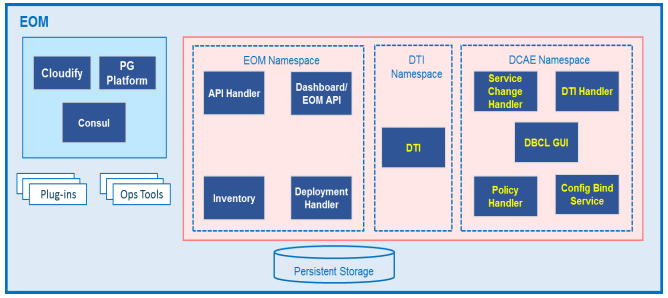
EOM features – highlights

Application Lifecycle Management	Efficient Operations	Resilience	Access Control	Security	DCAE
Management of containerized & non-containerized applications	Efficient installation of EOM Components	EOM Components Geo-Redundancy	AAF authentication	Secure communication via TLS using AAF certs	Onboarding/management of SDC (self-serve) microservices
Cluster Management, Bare metal K8s support	In-place upgrades of EOM components	EOM Components Local Redundancy	Cert management	Support multi-certs (Digicert, etc.)	Onboarding/management of non- SDC microservices
TOSCA & Helm support	Blueprint context awareness	**ECOMP Applications Geo-Redundancy**	User role management	**AAF namespaces for fine-grained role based authentication**	**CLAMP initiated microservices reconfiguration**
Auto-recovery	Script for generation of blueprints for Kubernetes onboarding	DCAE microservices resilience	**AAF role mapping to Kubernetes**	Event Logging and Analytics	DTI initiated microservices reconfiguration
Multi-Site Management	**Runtime configuration**	Affinity/Anti-affinity	Application and microservices user roles via ECOMP Portal		Policy initiated microservice reconfiguration
Metrics management based on Prometheus/Grafana	Multiple plug-ins				DMaaP integration
Auto-Ticketing	DBaaS (MySQL, MariaDB, PostgreSQL, Cassandra, MongoDB)				

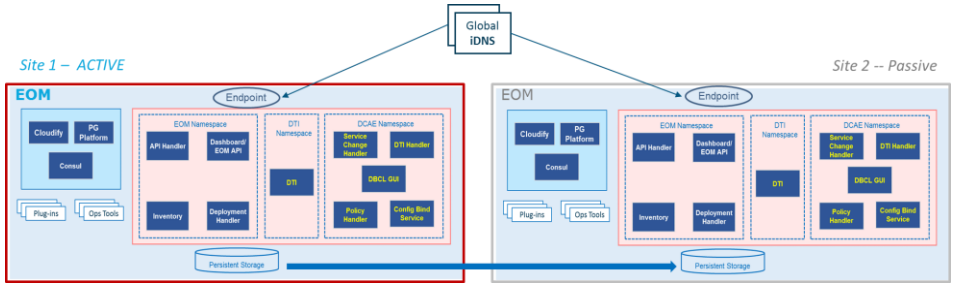
Resilience

EOM supports a rich set of resilience capabilities for itself as well as onboarded applications/microservices:

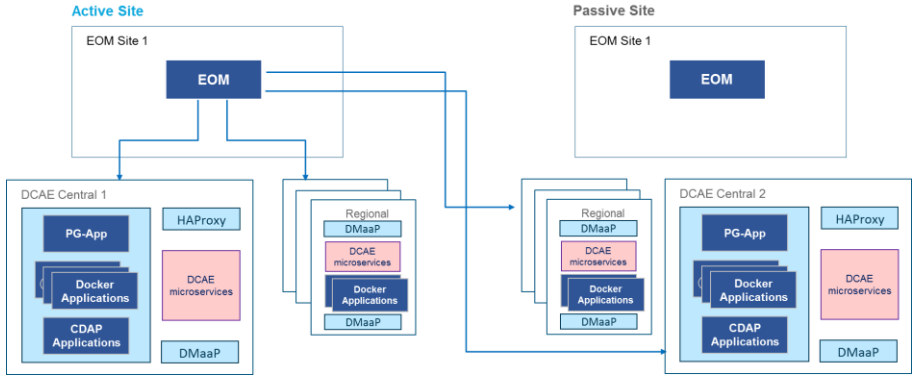
EOM components – Local resilience



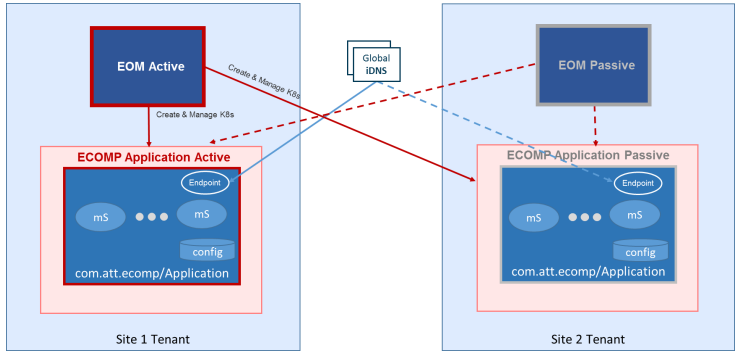
EOM components – geo-resilience



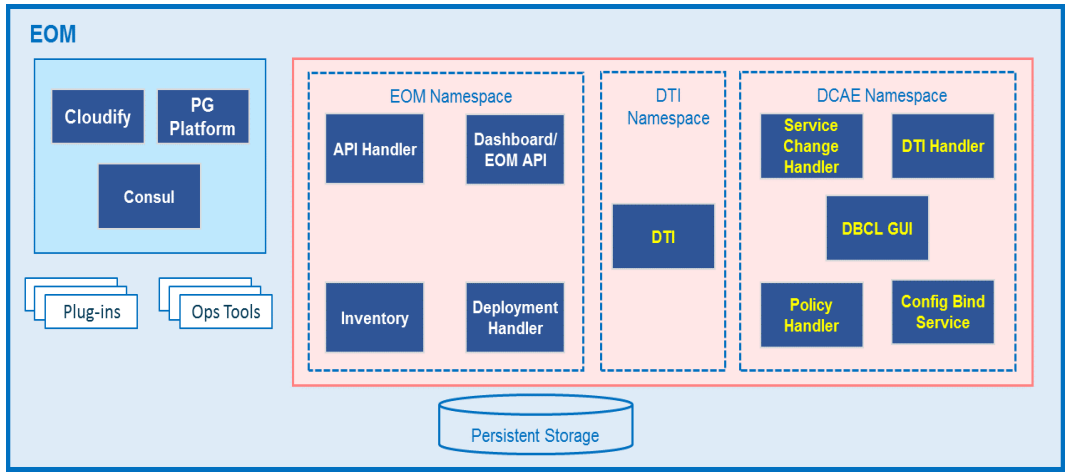
DCAE resilience

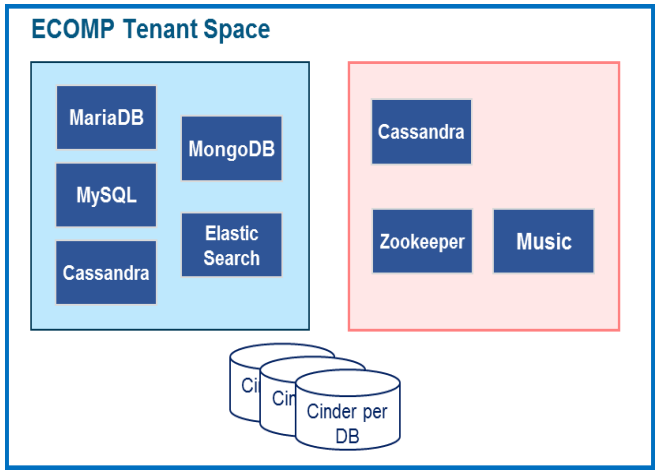


EOM Applications resilience



EOM components – local resilience

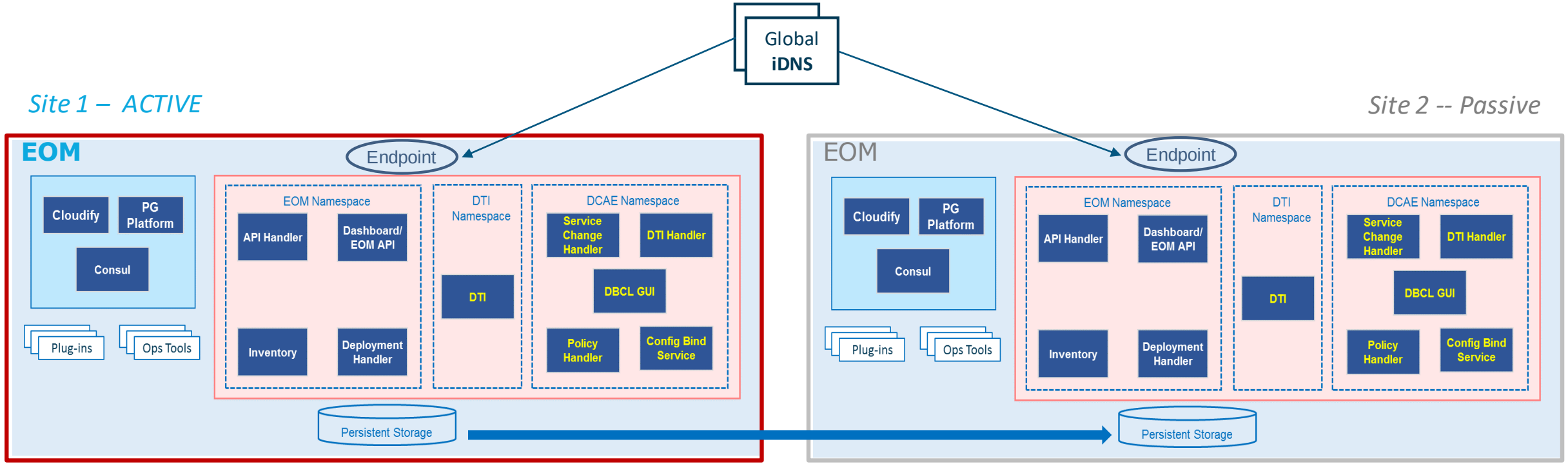




- Resilience for containerized Components is achieved via pod replicas
- Resilience for non-containerized EOM components is achieved by implementing redundant instances
  - Cloudify 4.4, Consul, PostgreSQL, and the DBs (MariaDB, MySQL, Mongo, Elasticsearch and Cassandra) implement their own resilience architectures on their own clusters consisting of 2 or 3 nodes on VMs per site
- Persistent storage is provided via Cinder volumes

VMs K8s

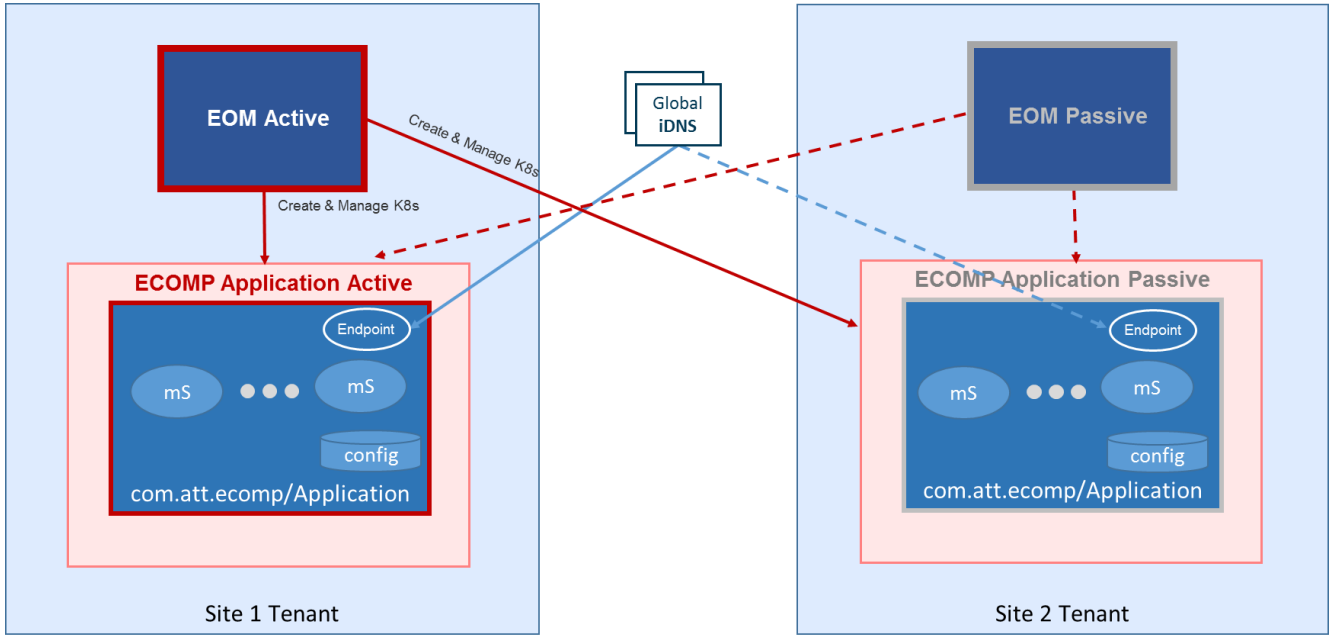
EOM components – geo-resilience



- EOM is installed in two sites: in active mode in one site, passive in the other
- Global iDNS communicates with an end point at each site to determine which site is active
- Data in persistent storage at the passive site is continuously updated with data from the active site
- If the active EOM fails, the passive EOM becomes active and takes over operations until the first EOM recovers, when normal operation is resumed

VMs K8s

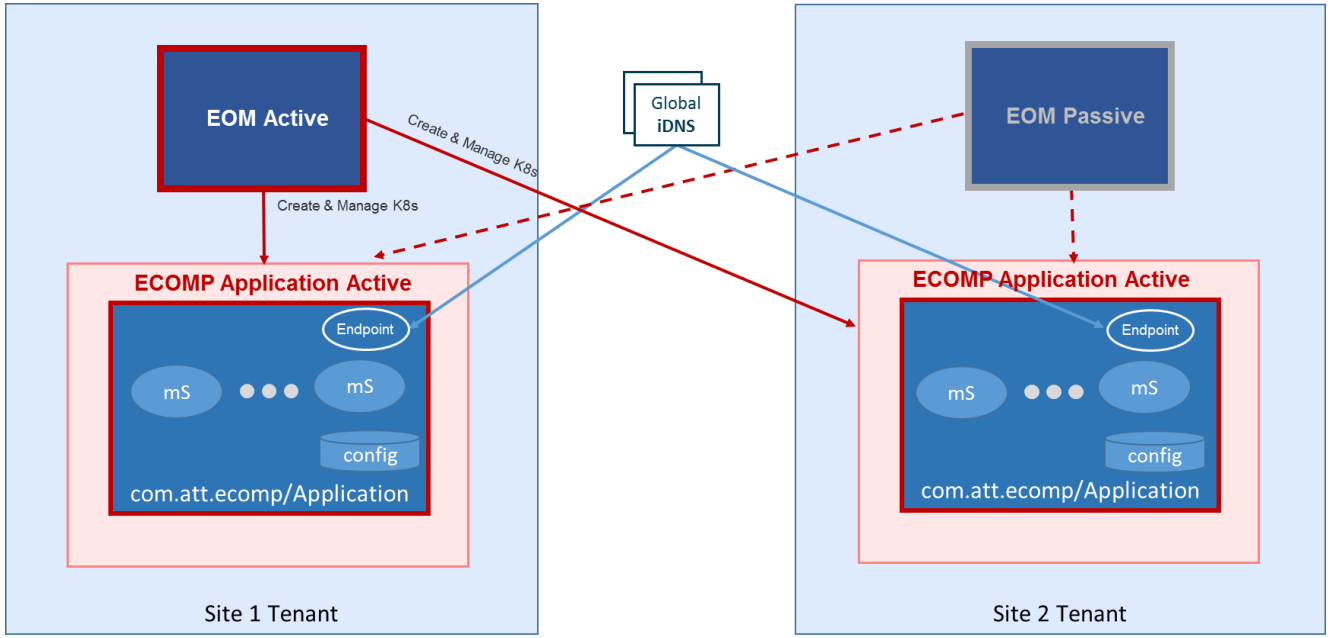
ECOMP applications geo-resilience: active-passive



- ECOMP Applications (e.g., SO) are installed in two sites: In active mode in one site and passive in the other
- An instance of EOM is present in each site: In active mode in one and passive in the other
- The active EOM provides support for monitoring, healing and scaling the ECOMP Applications **at both sites**
- Global iDNS communicates with an end point provided by each ECOMP Application to determine at which site the Application is active
- Each ECOMP Application implements its own resilience architecture
  - Local resilience is based on pod replicas
  - Geo-resilience is achieved via HA Proxy/Global iDNS
  - In the event of the active Application failing, the Application at the other site becomes active.

VMs K8s

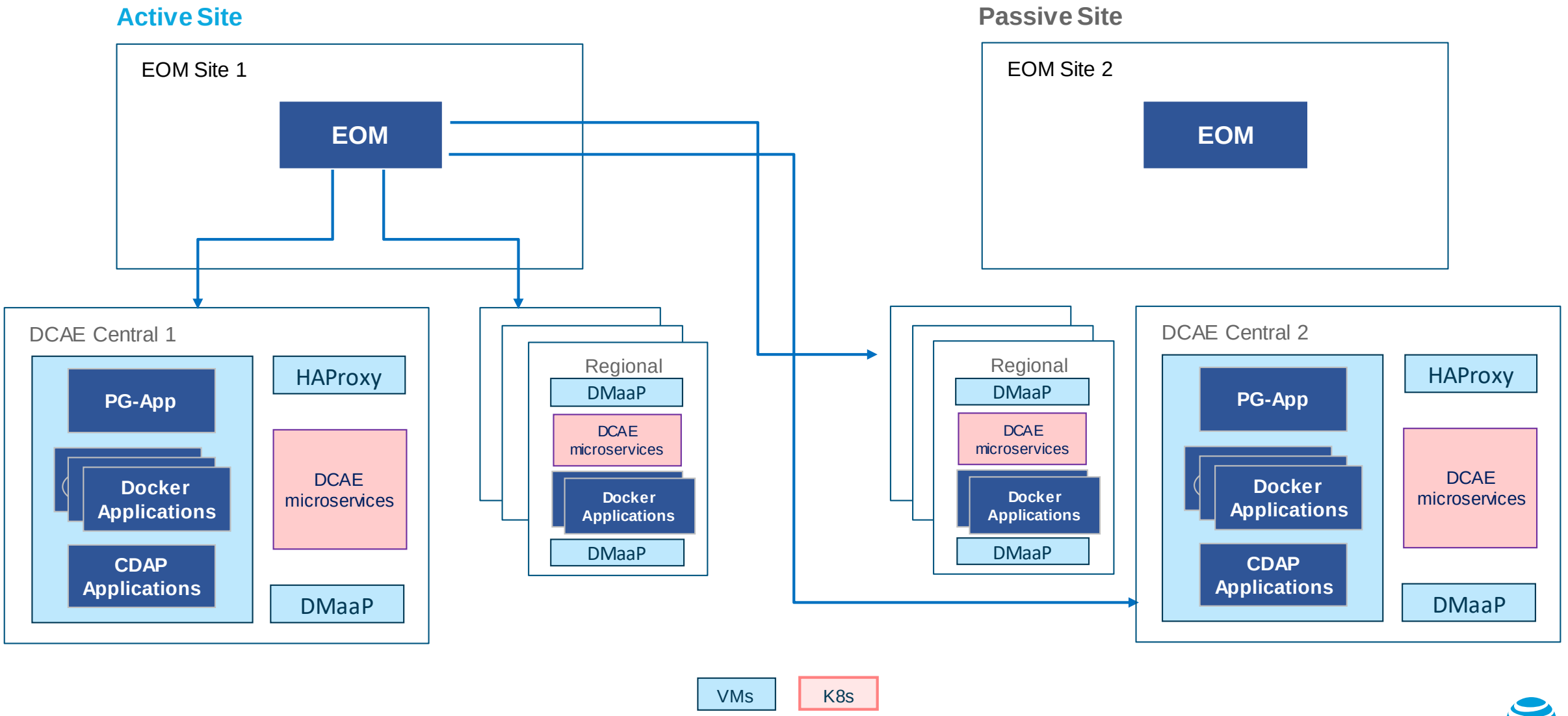
ECOMP applications geo-resilience: active-active



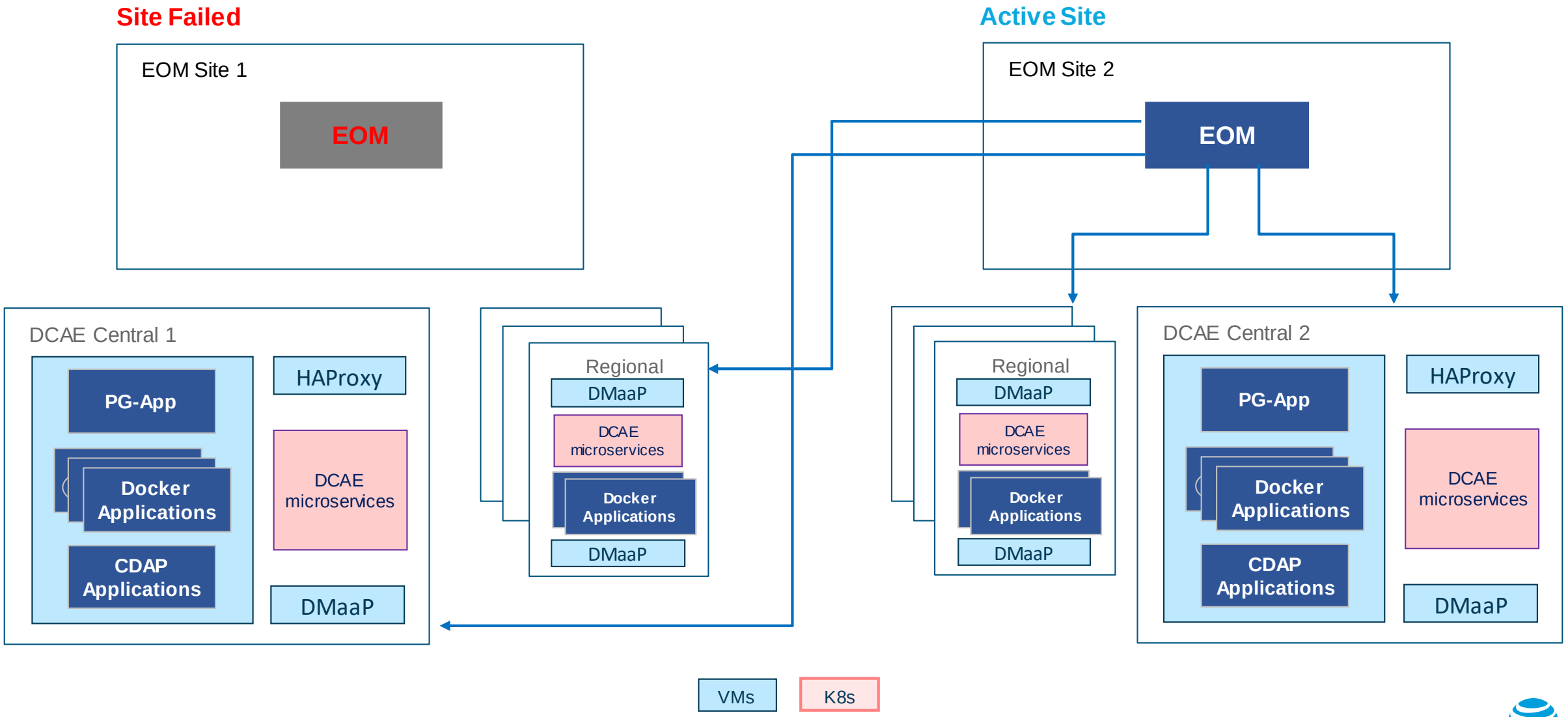
- ECOMP Applications (e.g., SO) are installed in two sites: In active mode in one site and active in the other
- An instance of EOM is present in each site: In active mode in one and passive in the other
- The active EOM provides support for monitoring, healing and scaling the ECOMP Applications **at both sites**
- Global iDNS communicates with an end point provided by each ECOMP Application for balancing the Application load between the active sites
- In the event of the active EOM failing, the passive EOM becomes active and manages the ECOMP Applications at both sites until the first EOM recovers, when normal operation is resumed

VMs K8s

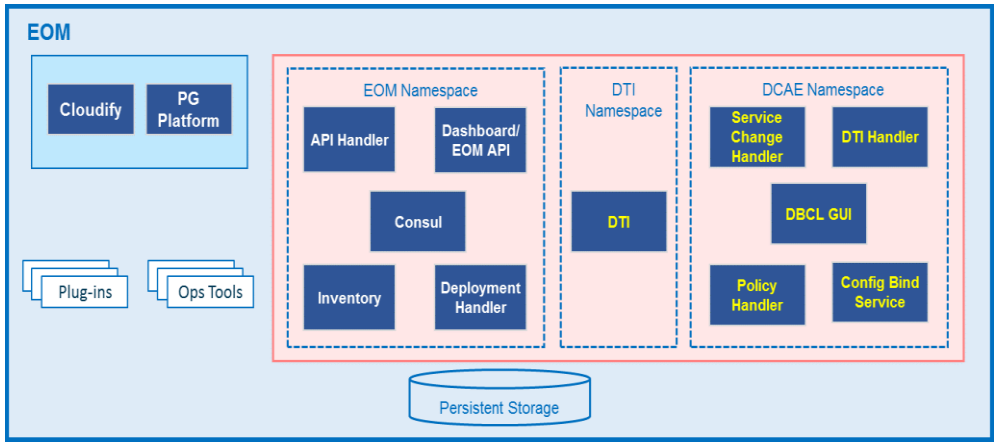
DCAE resilience (1 of 2)

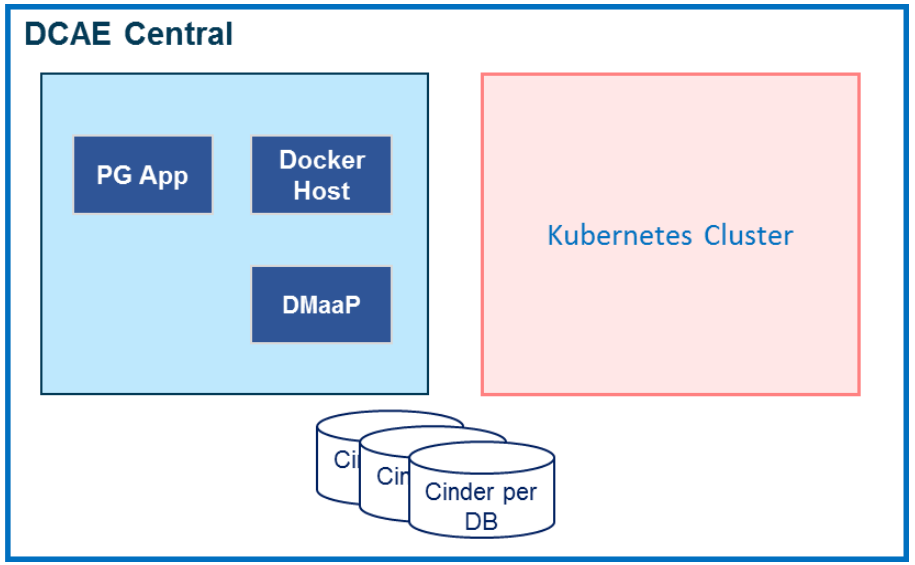


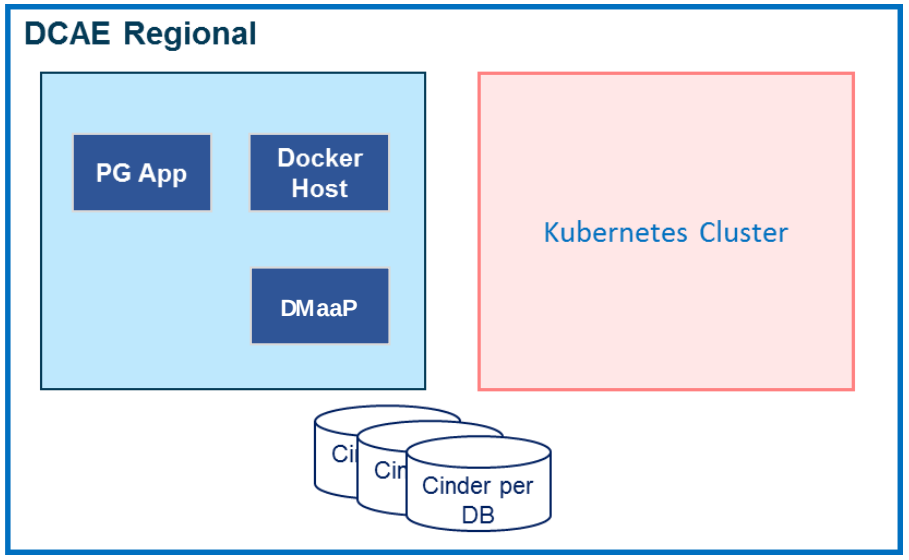
DCAE resilience (2 of 2)



Onboarding DCAE components via EOM







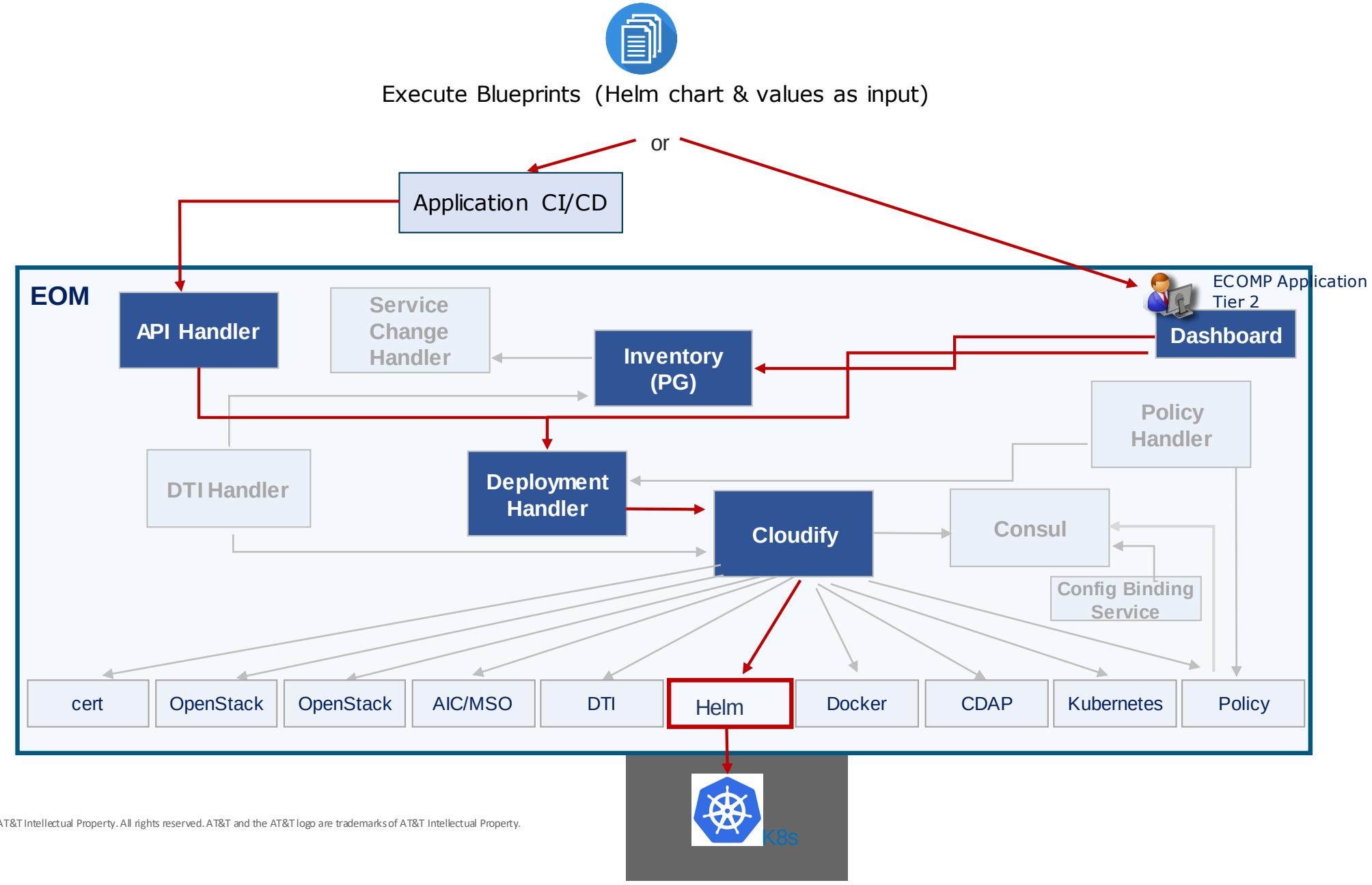
DCAE Platform components installed:

- Docker Cluster on DCAE Central and Regional sites
- PostgreSQL (Application)
- DMaaP on Central and Regional sites
- Kubernetes cluster on Central and Regional sites
- Ops Tools

EOM and Edge Automation requirements

Category	Requirement Item	EOM Mapping
Platform (Day 0)	Day 0 Bootstrap process : Ability to create ONAP platform using a single script	EOM multi-site platform creation
Onboarding	Ability to onboard management applications, deployed in cloud-regions, in ONAP-Central. Shall not have any expectations that all management applications are onboarded as a single bundle.	Allow new MS/applications/components to be onboarded independently
Onboarding	Ability to compose multiple management applications to be part of one management bundle and defining the dependency graph of applications belonging to a bundle	Supported through DCAE, SDC*, CLAMP
Instantiation	Ability to deploy management applications in selected cloud regions owned by ONAP operator	EOM WIP
Instantiation	Ability to deploy management applications in selected cloud regions not owned by ONAP operator, but has business relationship (Examples: Public Clouds or Edge Clouds owned by some other organization)	EOM PoC to support third party clouds
Run time	Ability to upgrade management application components without loss of functionality	Multi-site footprint & in place upgrade
Run time	High availability of management applications in the cloud regions	Multi-site resilience (active-active, active-passive management applications)
Platform	Enable tiered approach to support the volume, scaling and geo-distribution of platform and management applications	EOM can be instantiated to support geographic distributed management
Platform	Multi-site support	Single instance of EOM can orchestrate multiple site and manage life cycle of onboarded management applications
Security	Namespace separation and resource management	EOM supported

Onboarding ECOMP applications



Delta with ONAP DCAE

- EOM architecture is closely aligned to ONAP DCAE
  - Base platform components are already in ONAP
    - PolicyHandler, Deployment Handler, ConfigBindingService, InventoryAPI (available in R3)
    - Dashboard (Targeted R4)
- DTI/handler – For dynamic reconfiguration of deployment component
- Plugins/tools – To simplify monitoring of deployment components
- Helm Chart support (Dublin)
- Capability to Deploy DMaaP at edge

Monitoring EOM components & ECOMP applications

Monitoring of EOM Components and ECOMP Applications will be via a combination of several capabilities

Consul:

1. Performs health check on EOM Components hosted in the K8s cluster as well as on VMs (e.g. Cloudify, PG) and sends alerts to DMaaP
2. Performs health check on ECOMP Applications hosted in their K8s cluster as well as on VMs (e.g. databases) and sends alerts to DMaaP

Prometheus:

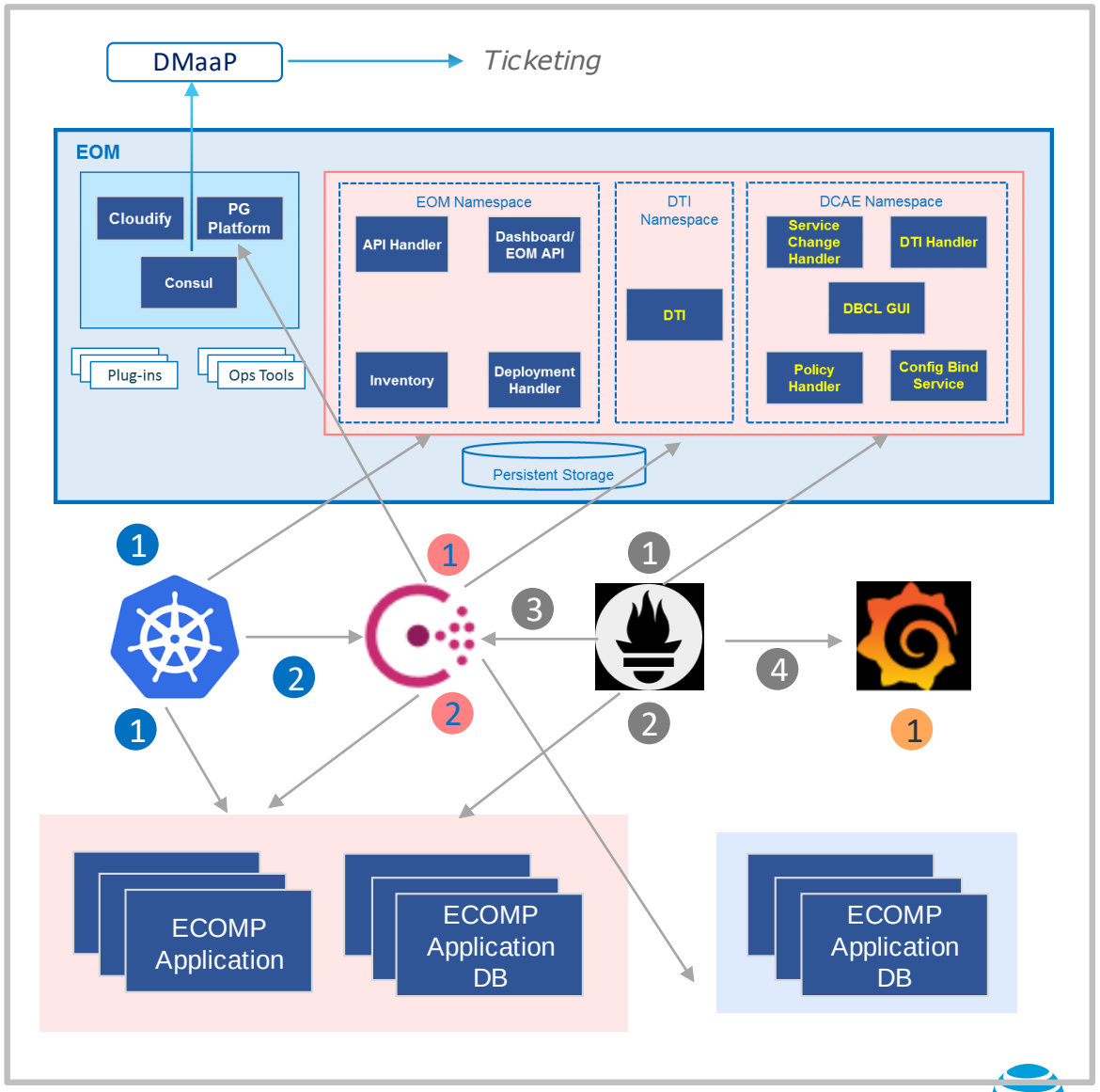
1. Collects performance metrics from EOM Components hosted on their K8s cluster
2. Collects performance metrics from the K8s cluster hosting the ECOMP Applications
3. Collects health check information from Consul
4. Transmits the metrics collected by itself and the Consul health check data to Grafana

Grafana:

1. Produces visualizations (dashboards etc.) and alerts based on the data obtained from Prometheus and Consul

Kubernetes

1. Healthchecks the containers
2. Kube2PyConsul utility sends the health check status to Consul



Summary

- EOM meets critical Tier 1 service provider platform requirements:
  - Highly Resilient
  - High Capacity/Transaction volumes
  - Highly Scalable/geographically distributed (microservices, centers. …)
  - Highly Flexible (wide range of databases, microservices, protocols, …)
  - Fast/agile onboarding of microservices
- Addresses Edge Automation group objectives:
  - Controller alignment
  - Management application requirements
- ONAP adoption of EOM involves minimal efforts (DCAE+)

Questions?

Backup slides

EOM and Edge Automation Requirements

Category	Requirement Item	EOM Mapping
Platform (Day 0)	Day 0 Bootstrap process : Ability to create ONAP platform using a single script	
Onboarding	Ability to onboard management applications, that are to be deployed in cloud-regions, in ONAP-Central. Shall not have any expectations that all management applications are onboarded as a single bundle.	Allow new MS/applications/components to be onboarded independently
Onboarding	Ability to compose multiple management applications to be part of one management bundle and defining the dependency graph of applications belonging to a bundle	Allow Service assurance flow composition and deployment of individual or group of component
Instantiation	Ability to deploy management applications in selected cloud regions that are owned by ONAP operator	Allow Service assurance flow composition and deployment of individual or group of component
Instantiation	Ability to deploy management applications that are ephemeral (example: Analytics applications)	Allow Service assurance flow composition and deployment of individual or group of component
Instantiation	Ability to deploy management applications in selected cloud regions that are not owned by ONAP operator, but has business relationship (Examples: Public Clouds or Edge Clouds owned by some other organization)	
Instantiation	Support for deploying management applications independent of each other when there are no dependencies (no expectation that all management applications are brought up together).	Allow Service assurance flow composition and deployment of individual or group of component
Instantiation	Ability to deploy few management applications based on VNF instantiations and bring down when VNF is terminated	Dynamic deployment of MS based on xNF instantiation
Instantiation	Ability to apply configuration (Day0 configuration) of management applications at the time of deployment	

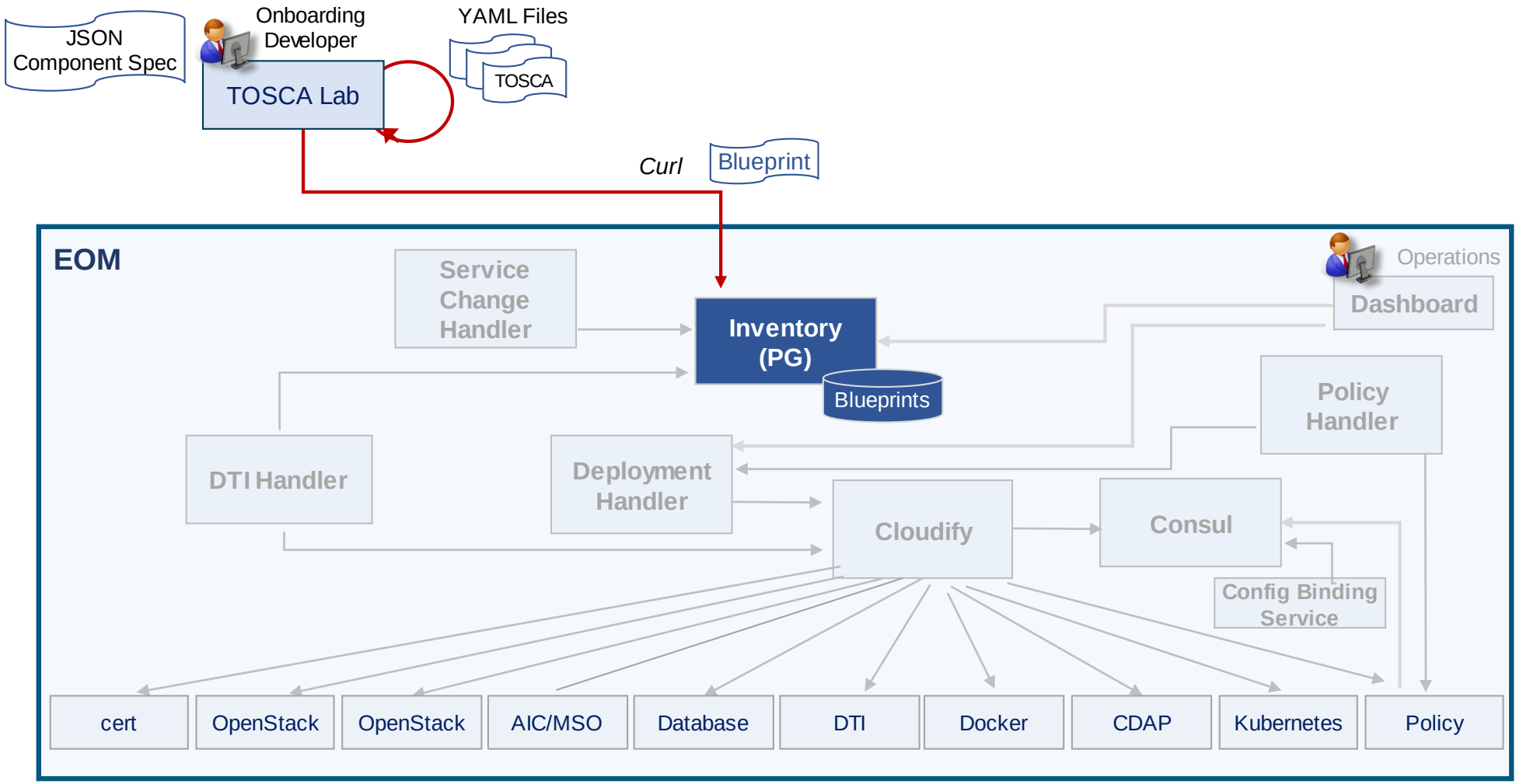
EOM and Edge Automation Requirements

Category	Requirement Item	EOM Mapping
Instantiation	Support for various Day0 configuration profiles (e.g. different profiles for different cloud regions w/ differing capabilities)	
Instantiation	Support for placement of management applications based on platform features (example: GPU, FPGA etc…)	
Instantiation	Support for consistent Day0 configuration mechanisms - should be the same path as Day 2.	
Run time	Support for Day 2 configuration of single or multiple instances of management applications in various cloud regions	
Run time	Support for management applications depending on other management applications - Support for configuration (Day2 configuration) of provider services when the consuming service is being instantiated and removal of the configuration on provider services when consuming service is terminated (Example: When analytics applications are brought up, analytics/collection framework need to be updated with additional configuration such as DB table, Kafka topic etc..)	Dynamic topics(MR) and feeds(DR) provisioning and role assignment for MS
Run time	Support for Day 2 configuration (add/delete) of appropriate management applications upon VNF instantiation/termination (Example: configuration of analytics & collection services when VNFs are brought up and removing the added configuration upon VNF termination)	Dynamic reconfiguration of MS based on xNF instantiations
Run time	Support for consistent Day 2 configurations across management components - should be same path as Day 0.	
Networking	Secure connectivity between central ONAP and management applications in cloud regions	
Networking	Support for various connectivity protocols (Kafka, HTTP 1.1, 2.0, GRPC, Netconf etc…) between ONAP -Central and management components in cloud regions	
Run time	Monitoring and visualization of management applications of cloud-regions along with ONAP components at the ONAP-Central	Complete view of MS and relation maintained at single/multisite K8S scenarios Health check of all deployment component to be available for CLAMP/external system
Run time	Scale-out of management application components at the cloud-regions & traffic (transaction) distribution	

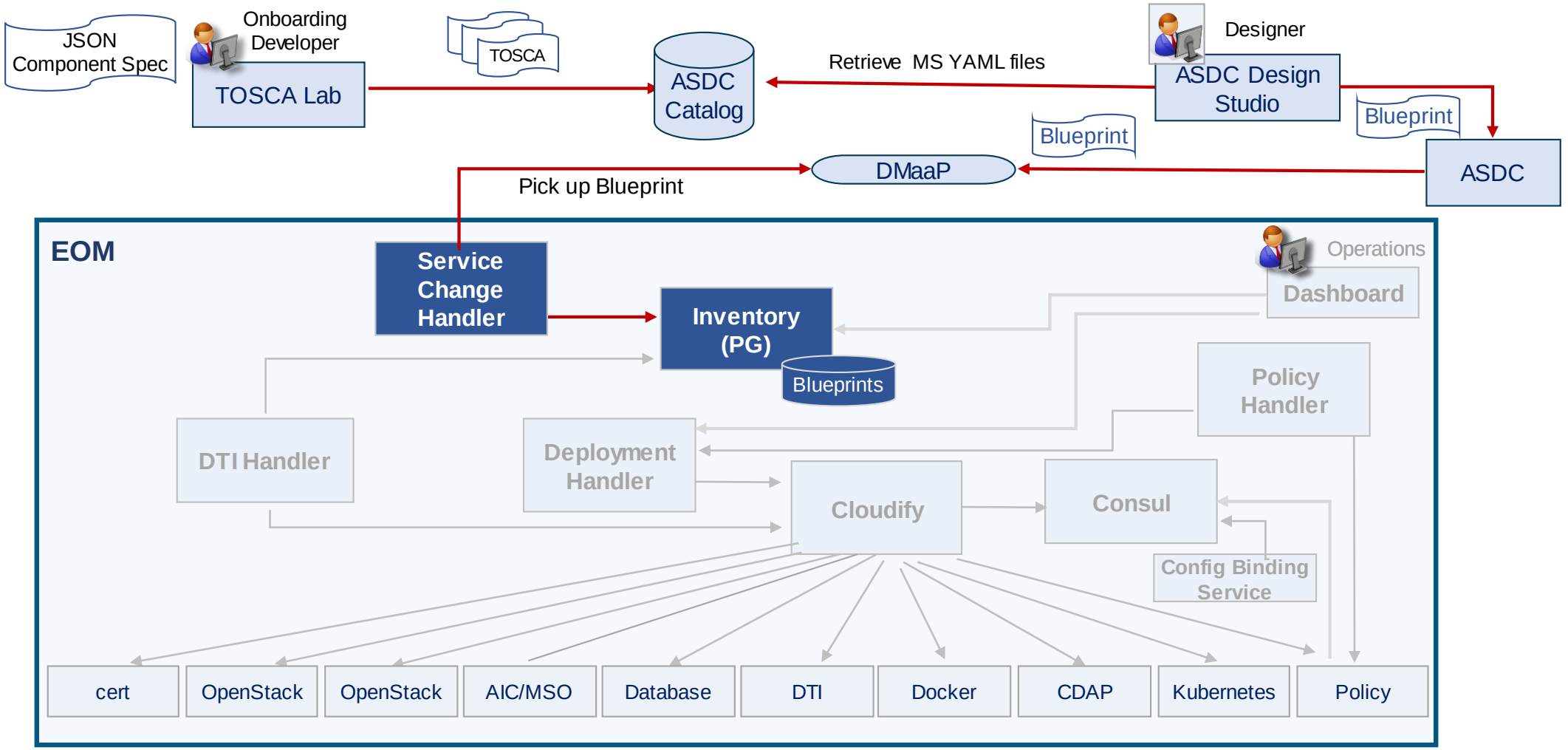
EOM and Edge Automation Requirements

Category	Requirement Item	EOM Mapping
Run time	Ability to upgrade management application components without loss of functionality	Multi-site footprint + in place upgrade capability available
Run time	High availability of management applications in the cloud regions	Multi-site resiliency (active-active, active-passive management applications)
Miscellaneous	Support for ONAP-compliant third party management applications that provide similar functionality as ONAP management applications. Some of the key aspects of ONAP-compliance include but are not limited to the following - API compatibility, Cloud Native Packaging in ONAP Helm chart format etc.	
Miscellaneous	Support management applications as containers	
Miscellaneous	Support management applications as VMs	
Security	Security and privacy aspects of management applications (To be expanded)	
Security	Access control management	
Security	Namespace separation and resource management	
Instantiation	Support for MS deployment not binded to any VNF/service; these are application which are service agnostic can be managed by dynamic configuration rule to support different use cases	
Miscellaneous	Backward compatibility with existing application based on TOSCA	
Miscellaneous	Non-containerized management application support	
Run time	Service registration, Alert monitoring and Ticket creation	
Platform	Enable tiered approach to support the volume, scaling and geo-distribution of platform and management applications	
Platform	CI/CD integration	
Platform	Backup/Restore capabilities	

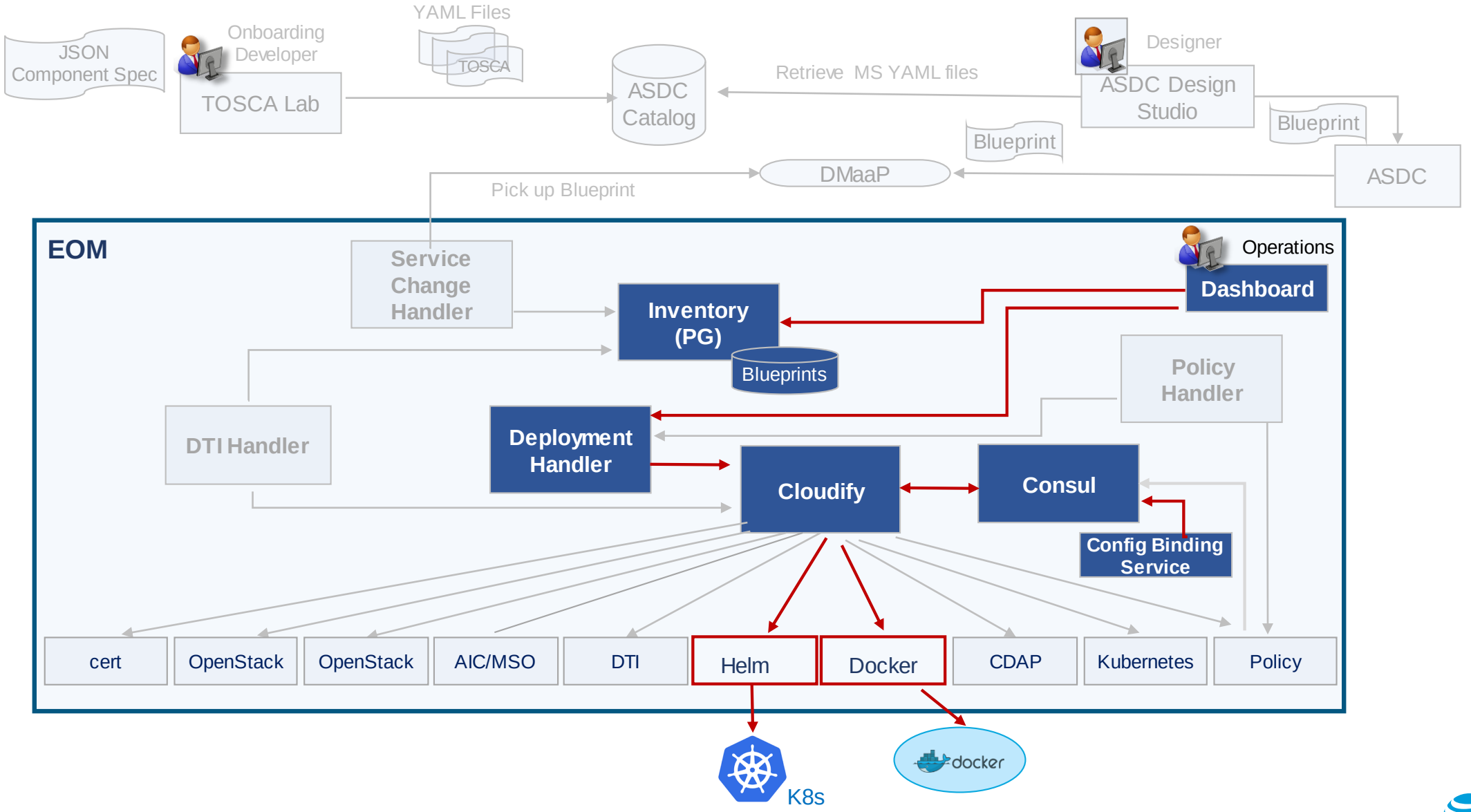
Microservice Onboarding without ASDC



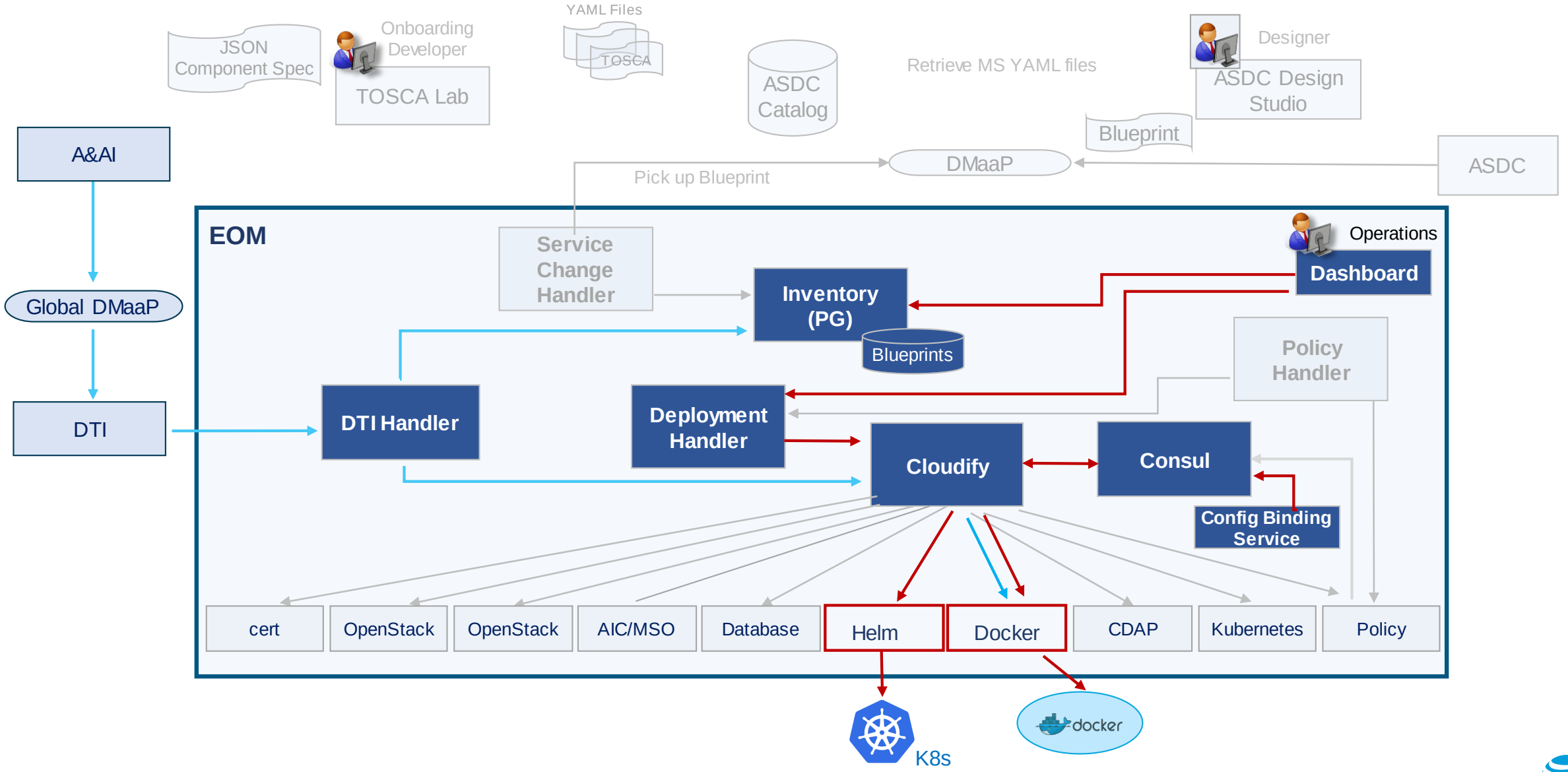
Microservice Onboarding with ASDC (self-service)



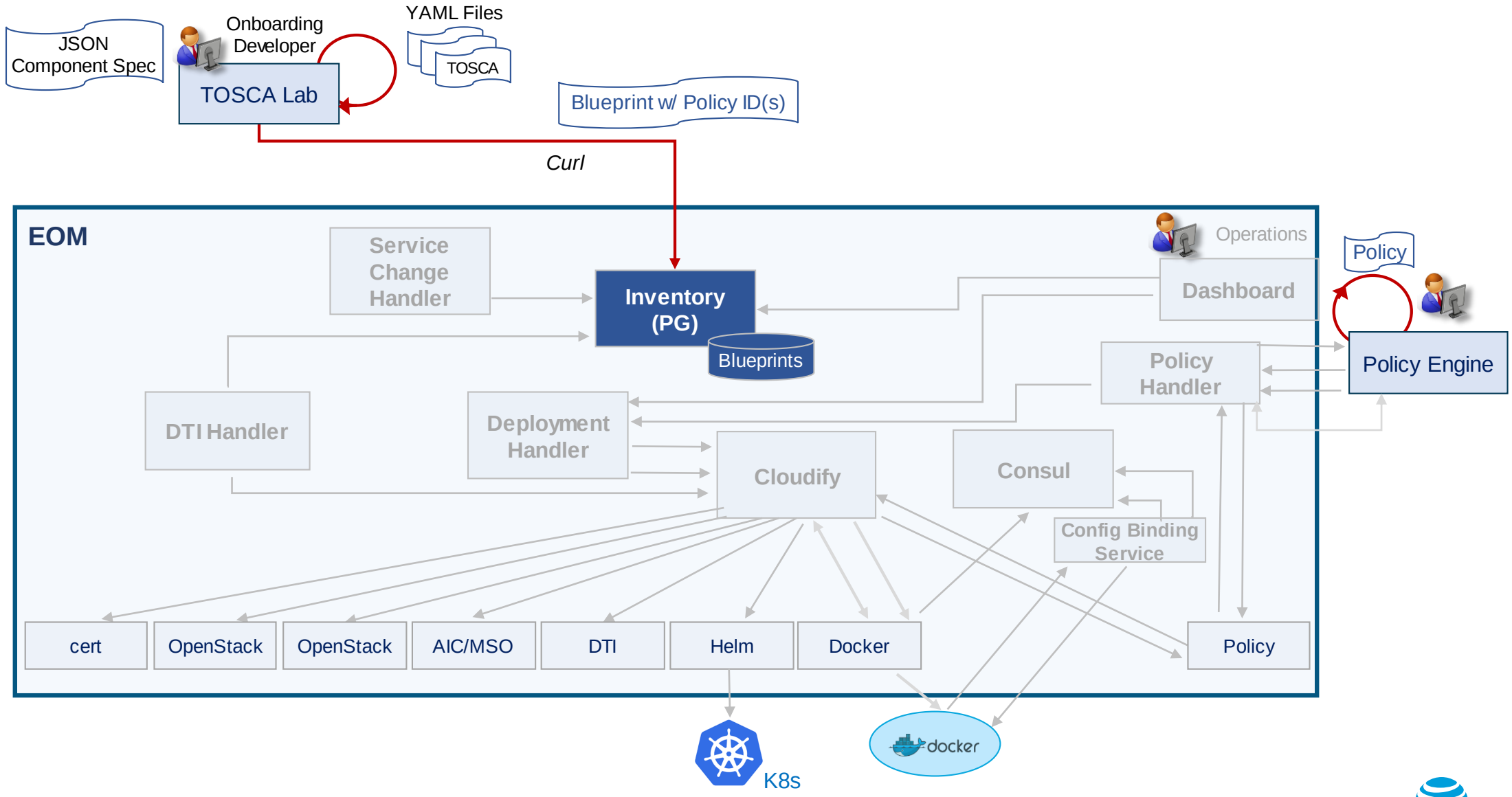
Microservice Deployment (via Dashboard)



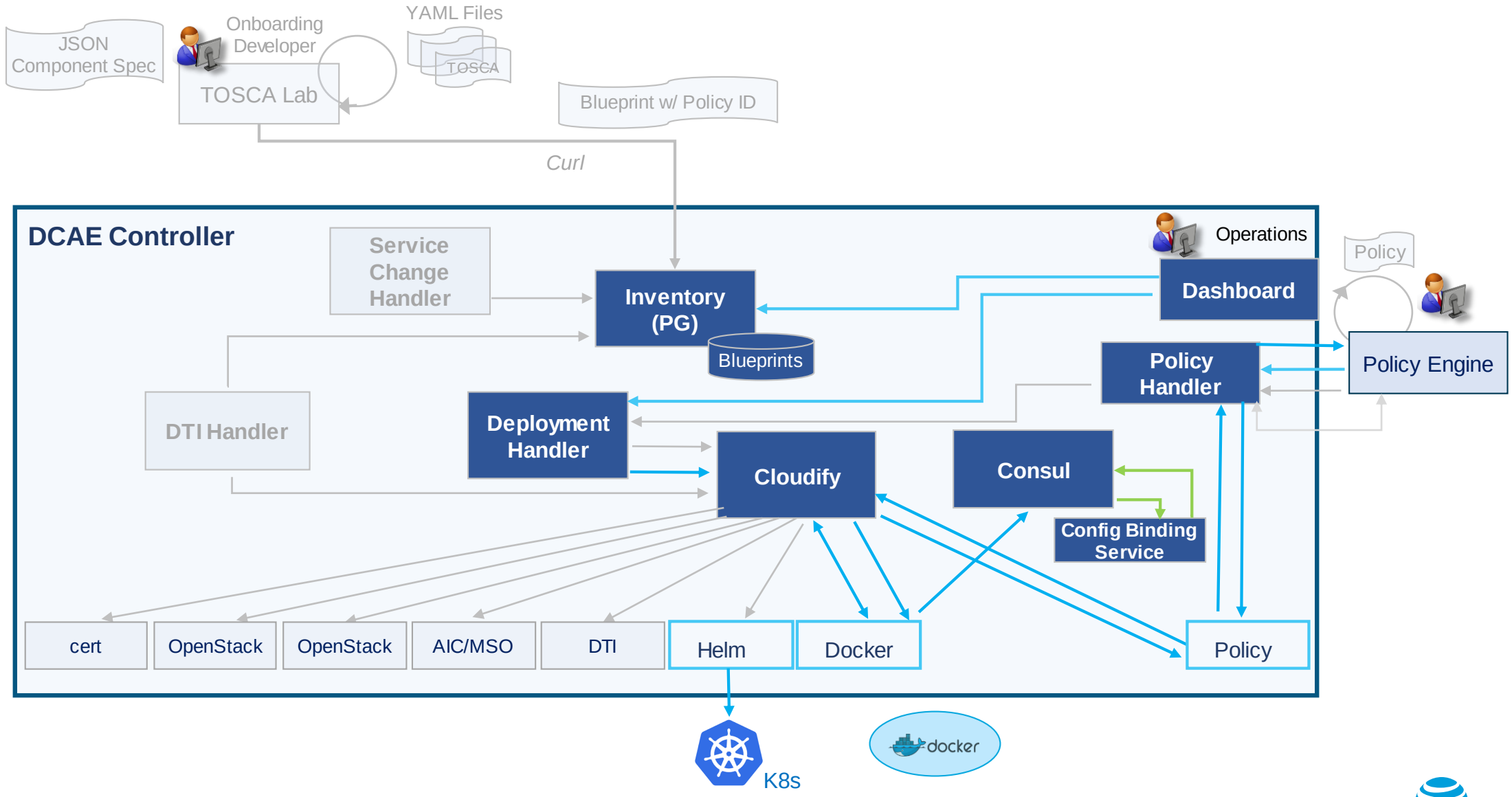
Reconfiguration via DTI



Policy Flow part 1 – Blueprint (with Policy ID) to Inventory



Policy Flow part 2 – Blueprint Deployment



Policy Flow part 3 – Policy Update