
APACHE SLING & FRIENDS TECH MEETUP
BERLIN, 25-27 SEPTEMBER 2017

# Internet Scale Content Management with Apache Oak on Kubernetes

Fernando Saito, Galo Gimenez, HP Inc

# Content at HP

- HP is in the business of transforming digital objects to physical objects, and physical objects to digital objects
- Digital objects come in multiple forms, 3D models, documents, intermediate rendered artifacts, print jobs, etc.
- Our secure document management platform allows HP devices and applications access and store content
- 65M devices connected, ~50K documents per hour

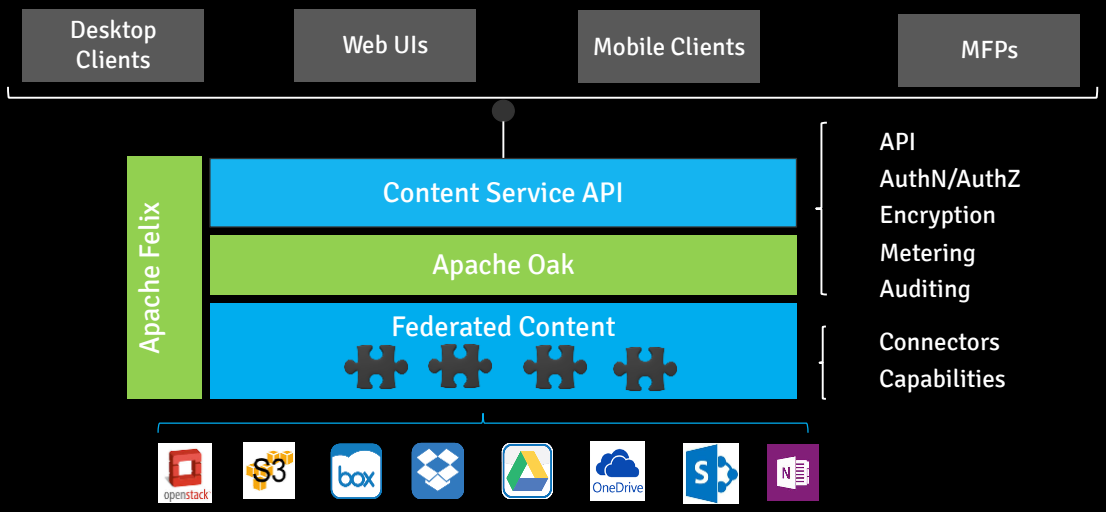


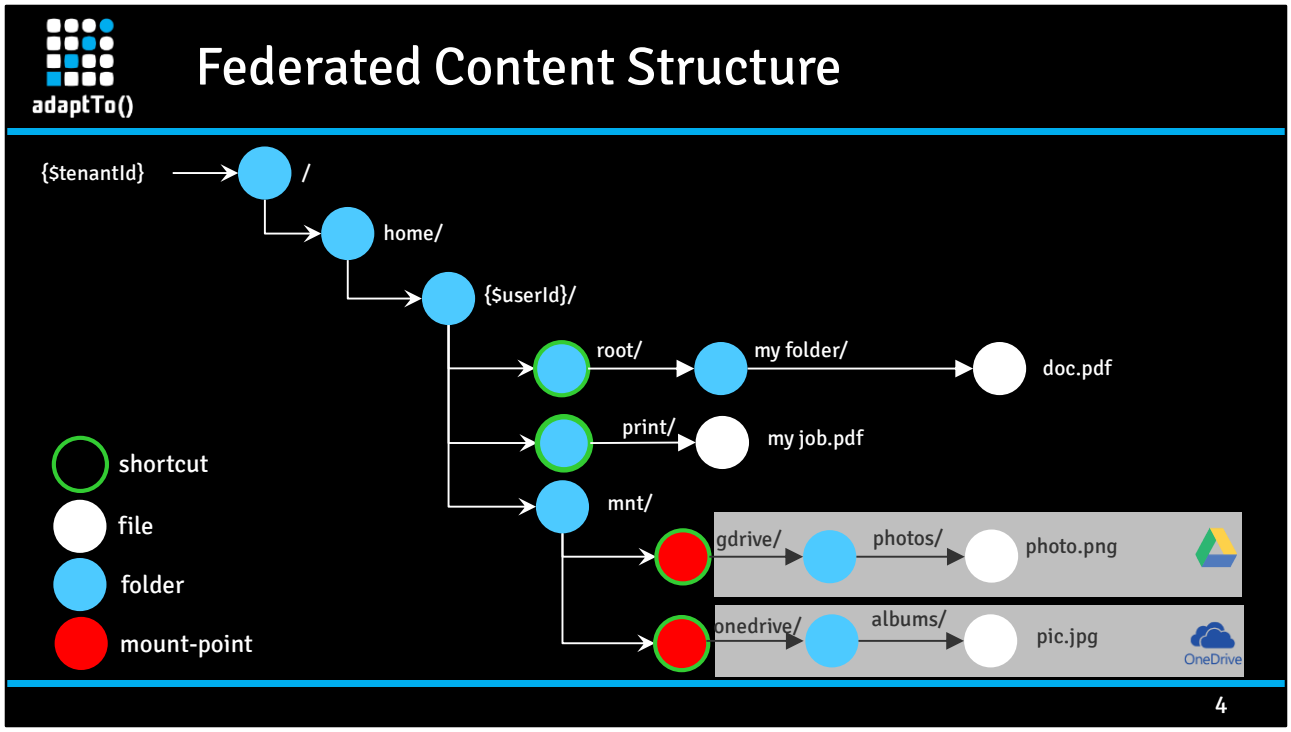


2

Content is a key enabler for many HP devices, be it multi-function printers, high speed commercial printers, 3D printers, or immersive computing platforms for creative solutions like HP Sprout. HP is in the business of transforming content from digital to physical and from physical to digital. Enabling this transformation keeps our devices constantly connected to cloud services.

# SDM Content Service Architecture



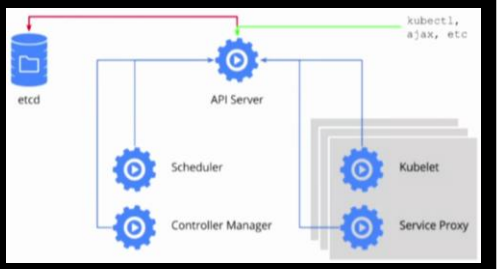



Federated Content

- Register custom node types (connector/mount-point)
- Data access abstraction layer
  - Node interface implementation
  - Federated repository service API client implementation

# Kubernetes

- A container scheduler system inspired in Google experience running containers
- Pods – scheduling containers in the same node
- Discovery – DNS based discovery allows legacy workloads to work
- Stateless and Statefull workloads

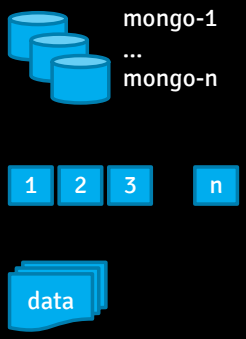



5

Kubernetes is an open-source system for automating deployment, scaling, and management of containerized applications

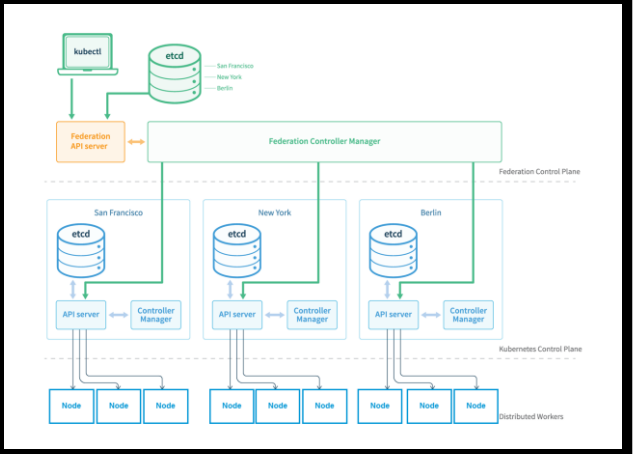
# Kubernetes StatefullSets

- Designed for state full apps – i.e. Oak, MongoDB
- Consistent Naming – (Journaling has the same node names, Mongo replicas)
- Ordered start – (Solves race conditions setting up MongoDB)
- Attached to permanent storage – (Can use local Lucene Indexes and H2 caches)



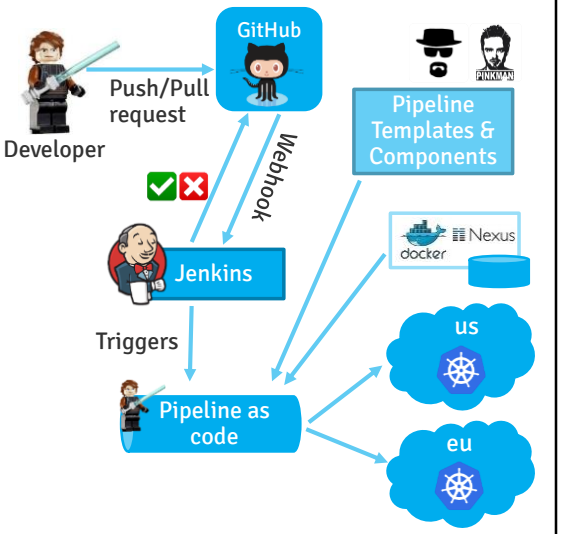
# Kubernetes Federation

- Single API to access multiple Kubernetes Clusters
- Cluster state reconciliation
- Federated Services

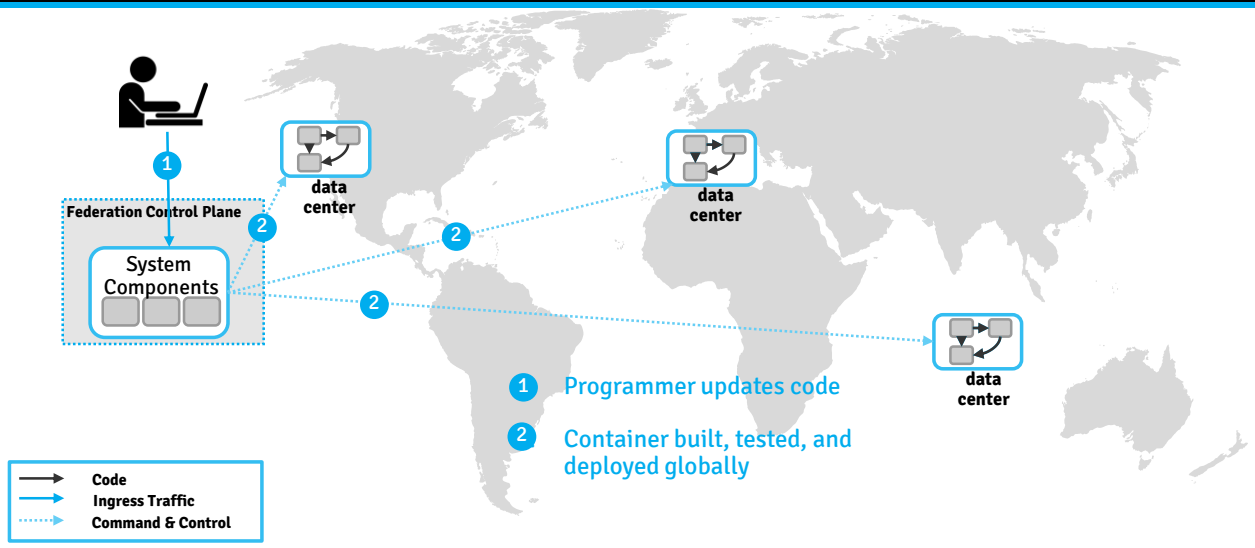


# But Federation still lags some features

- Custom deployment toolset
- Multiple kubernetes clusters contexts
- Annotation modifiers
  - Region and Environment specific configurations
  - Regional and Global DNS
  - Public and Private names
- Resource quota

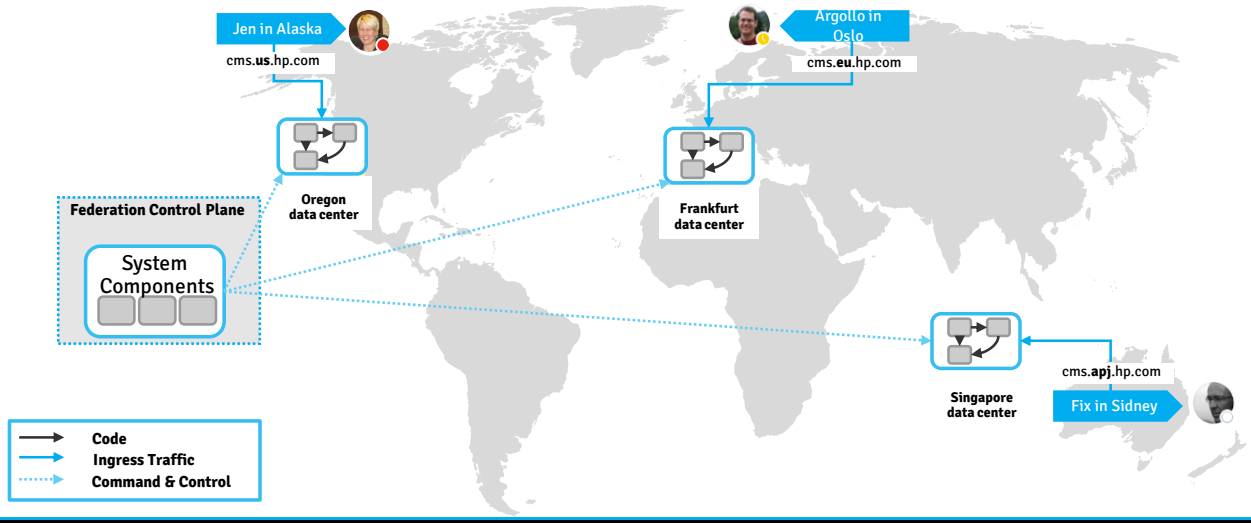


# Global Deployment



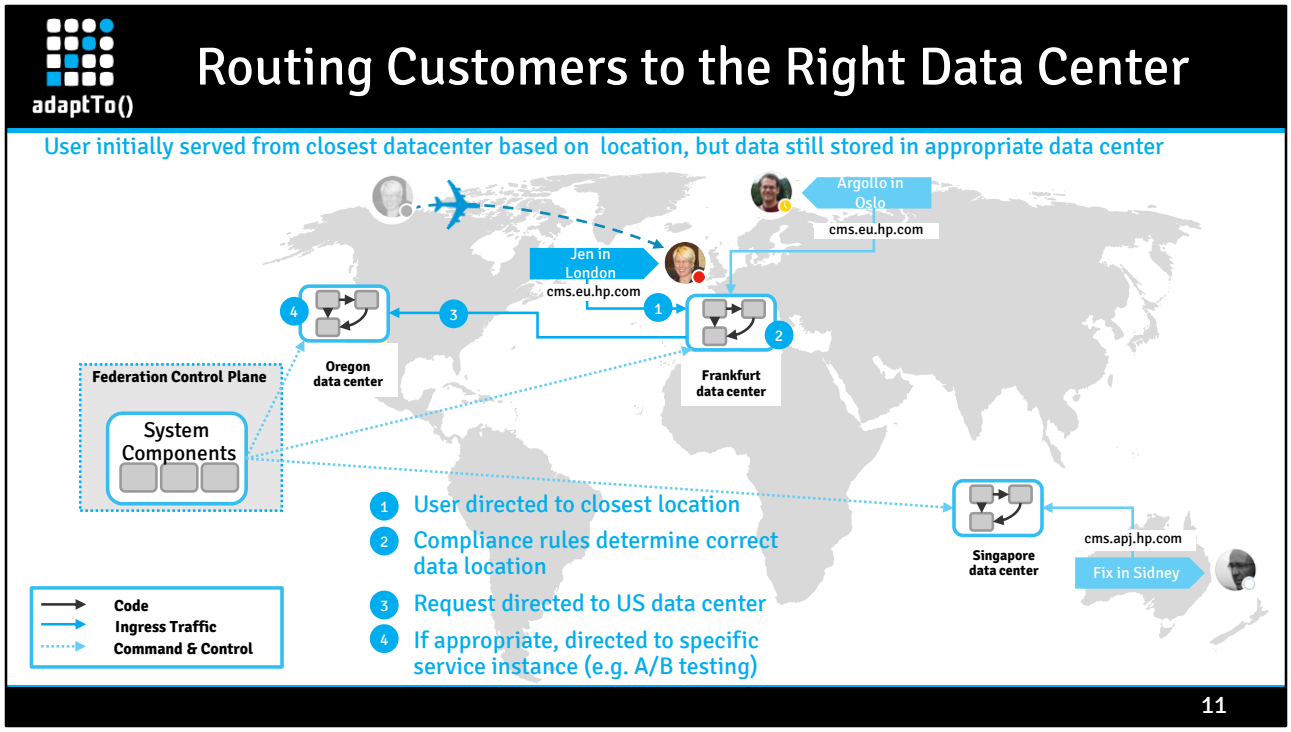
# Optimize Customer Experience

User data is stored in the data center appropriate for that user based on global data compliance policies





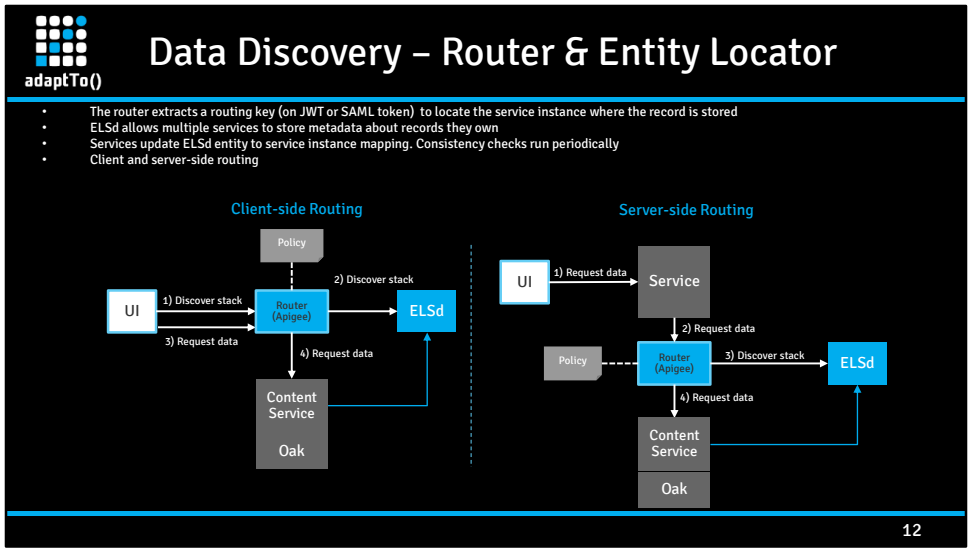

User signs up for an HP application or service, their data gets stored in the data center that is appropriate based on the global data compliance policies that are implemented as part of the service mesh or policy enforcement.



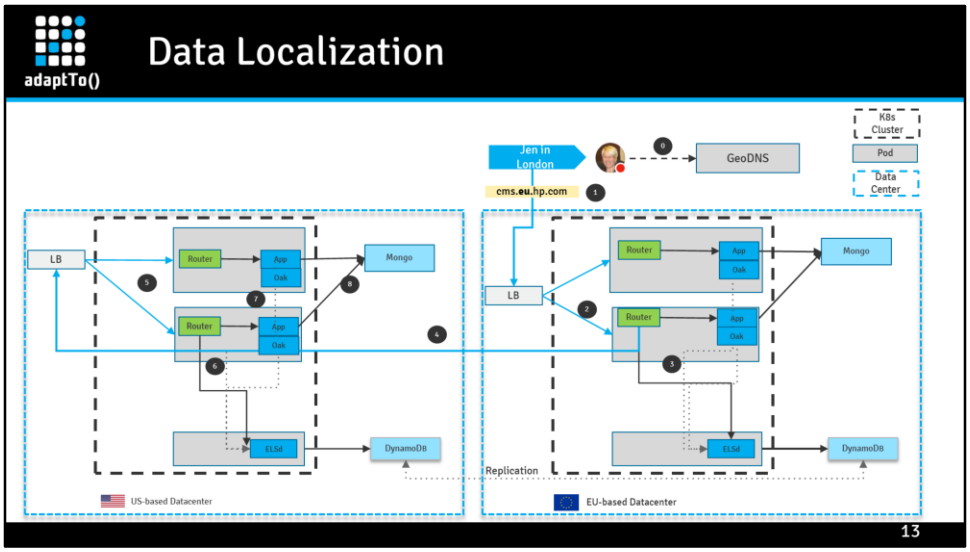

Business trip to London, I start to log in to my HP application…

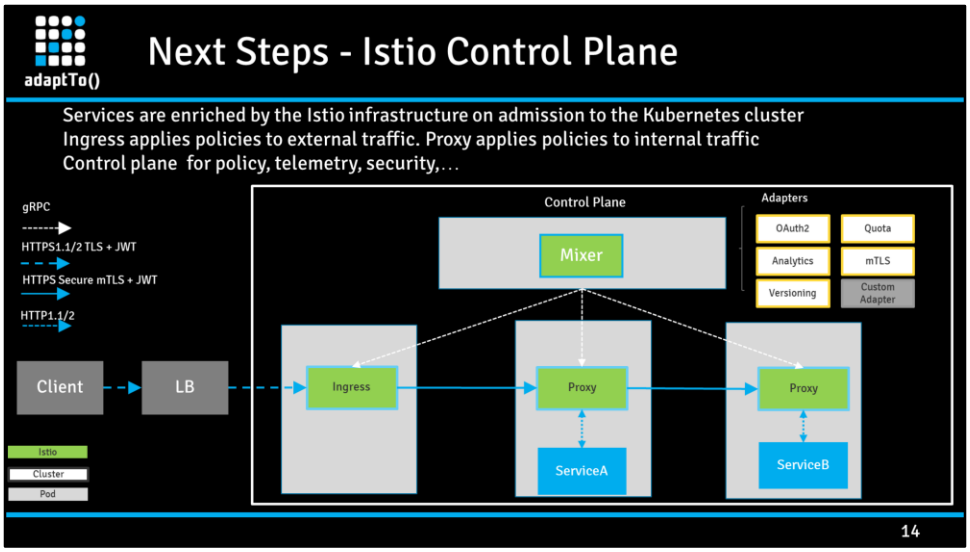
1. My request is directed to closest location based my geolocation
2. At the service mesh my request detects that my data actually resides in the US data center, so my
3. request is directed to US data center
4. And optionally , if we want, we can direct the request to a specific service instance or version, for example if we wanted to do A/B testing or canary testing.

So briefly, and at a high level, that is what GDRS does from an application team or developer perspective, as well as an end customer perspective.



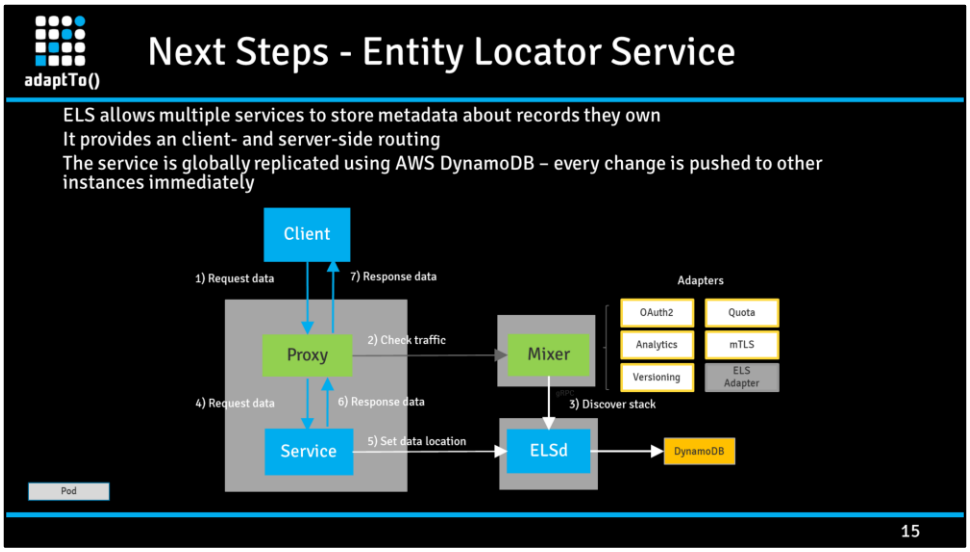

http://docs.apigee.com/microgateway/latest/overview-edge-microgateway
https://github.com/hpcwp/elsd


adaptTo()
Data Localization
K8s Cluster
Pod
Data Center
Jen in London
cms.eu.hp.com
GeoDNS
LB
Router
App
Oak
Mongo
ELSd
DynamoDB
Replication
US-based Datacenter
EU-based Datacenter
13




https://istio.io/docs/concepts/what-is-istio/overview.html
Istio is an open source project. It is a network traffic fabric, to support high level networking functions like quotas, authentication, canary test, monitoring.

Istio advantages over Apigee
- mTLS / Mutual TLS: guarantees the identity of the server to the client as well as identify of the client to the server



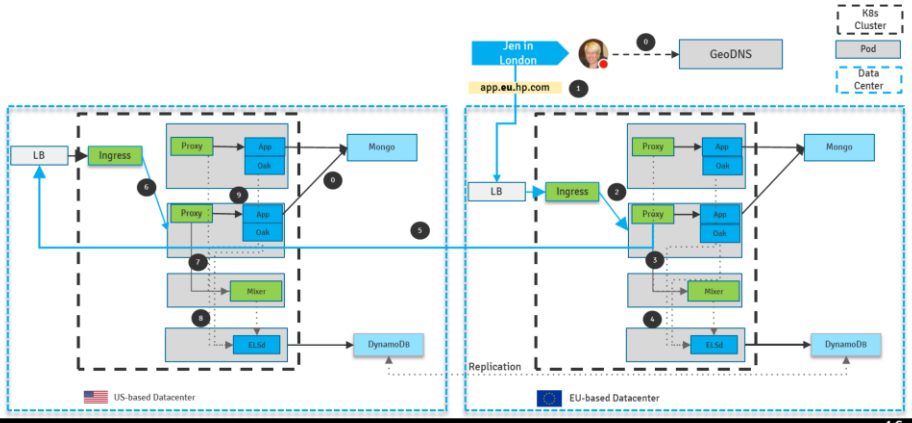

Proxy: Istio uses an extended version of the Envoy proxy, a high-performance proxy developed in C++, to mediate all inbound and outbound traffic for all services in the service mesh.

Mixer: Mixer is responsible for enforcing access control and usage policies across the service mesh and collecting telemetry data from the Envoy proxy and other services. The proxy extracts request level attributes, which are sent to Mixer for evaluation.

https://grpc.io/

adaptTo()

# Next Steps - Data Localization




16


# Thanks / Danke