Trent Waddington

Sr. Principal Software Engineer

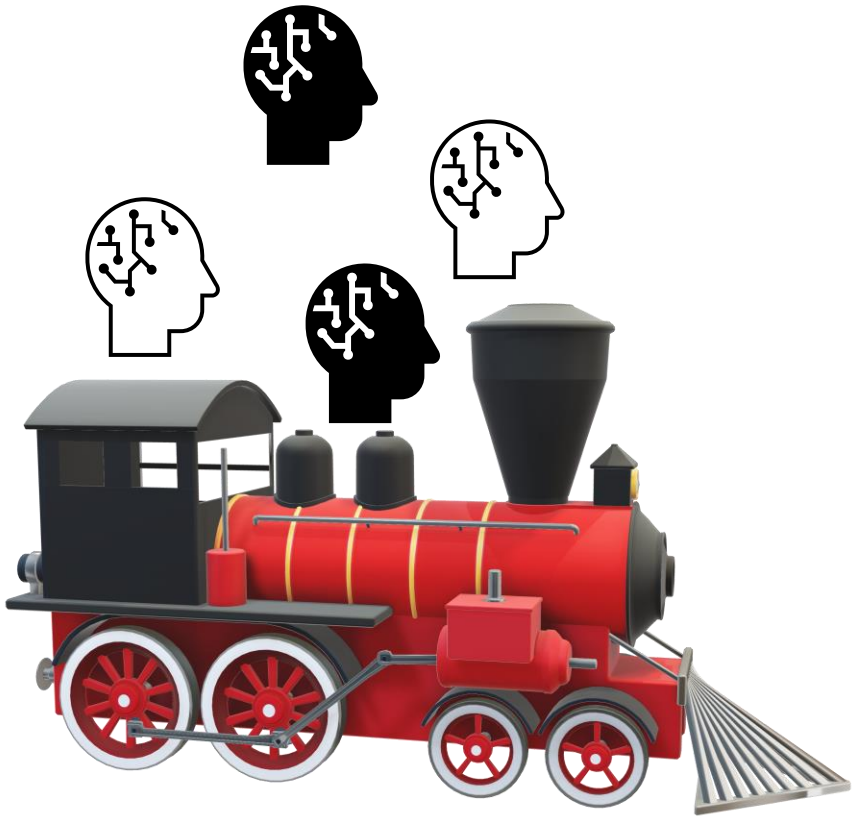
Agenda

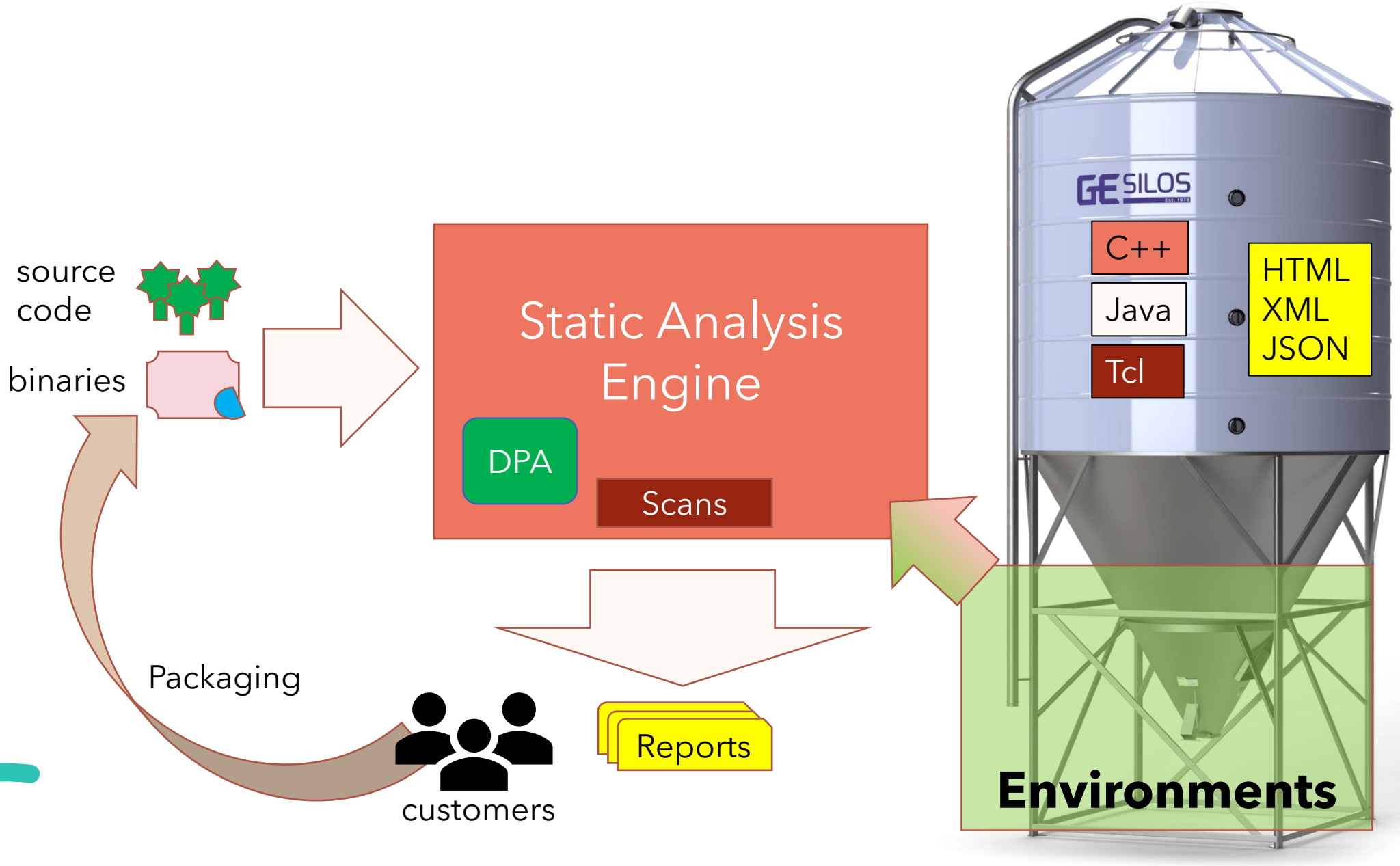
In the beginning…
The Engine.
Cloud migration.
Architecture.

Who are we?

Veracode was founded by a group of concerned *InfoSec consultants* with the vision of *securing the world's software* by finding, reporting and remediating security flaws – hopefully before they become a security incident – using *automated* program analysis.

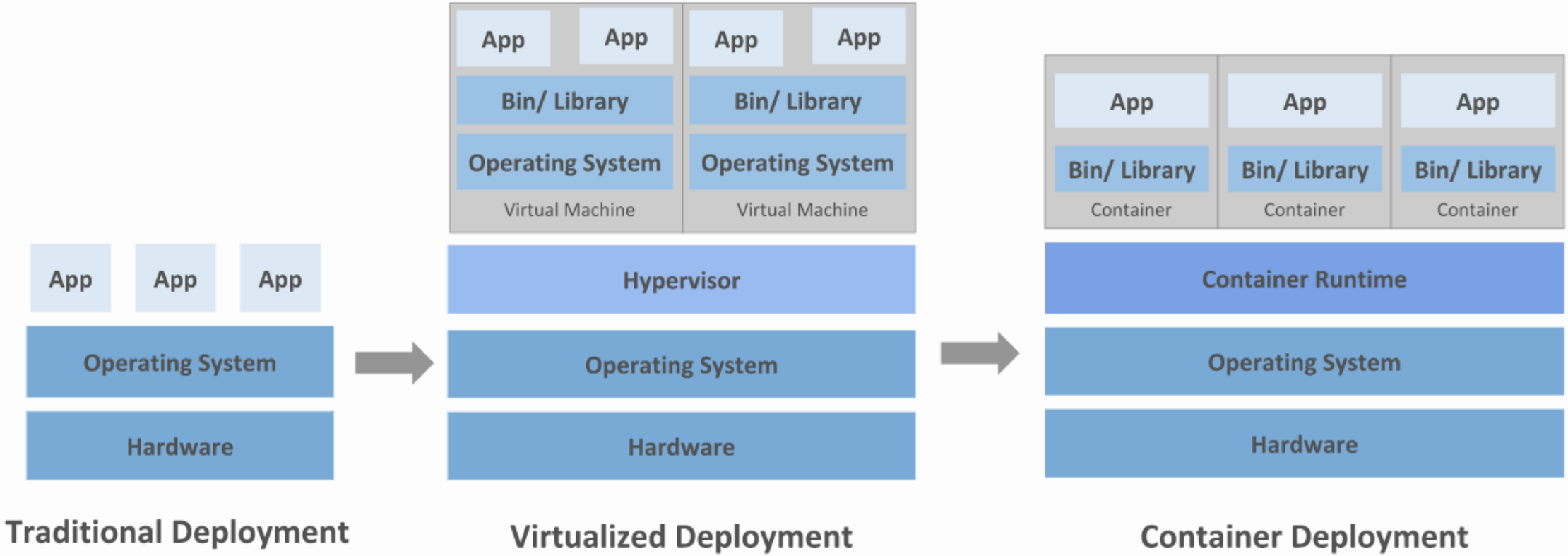

In the
beginning…




source code
binaries
Static Analysis Engine
DPA
Scans
Packaging
customers
Reports
C++
Java
Tcl
HTML
XML
JSON
Environments

Cloud migration

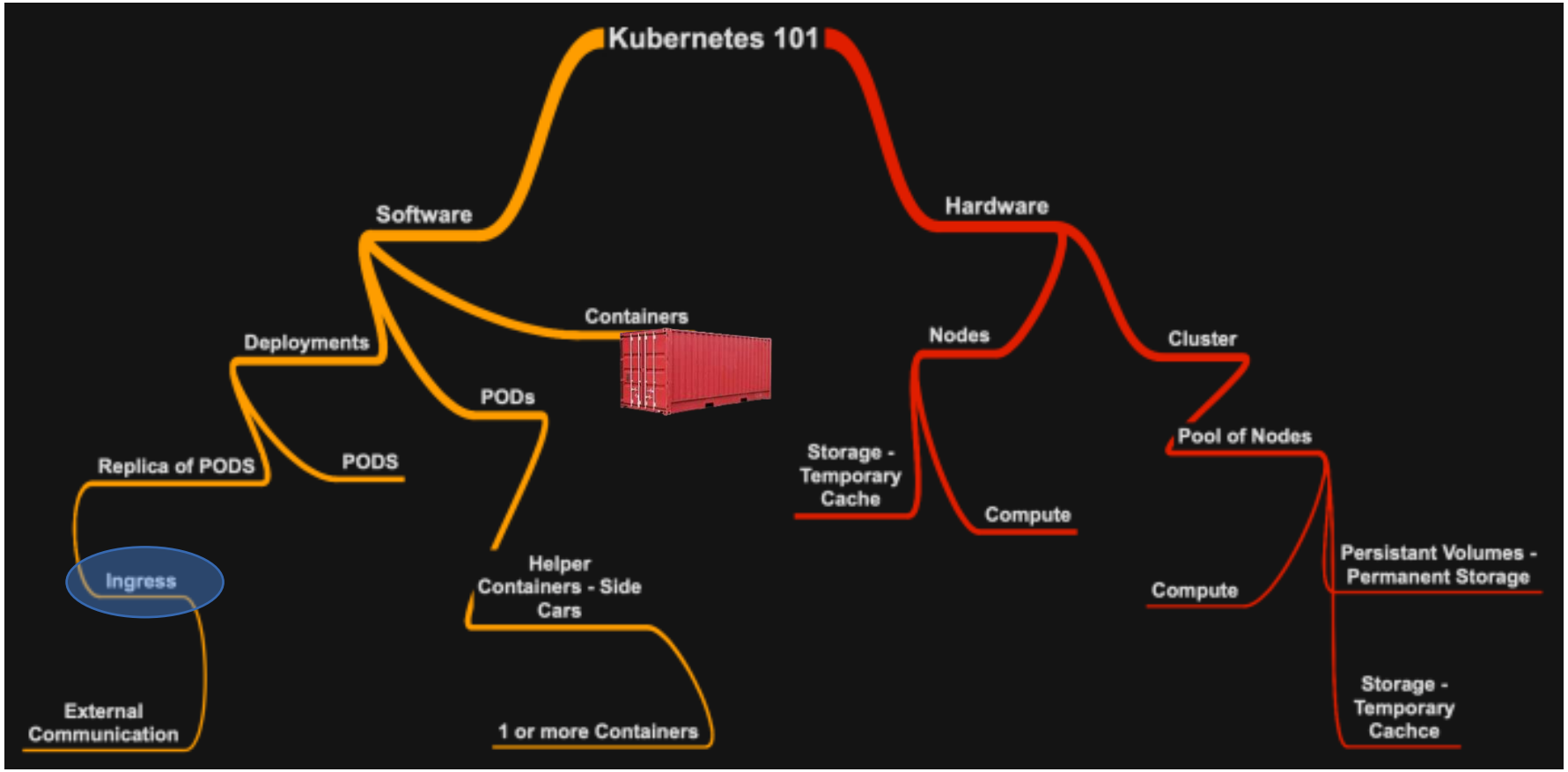
kubernetes

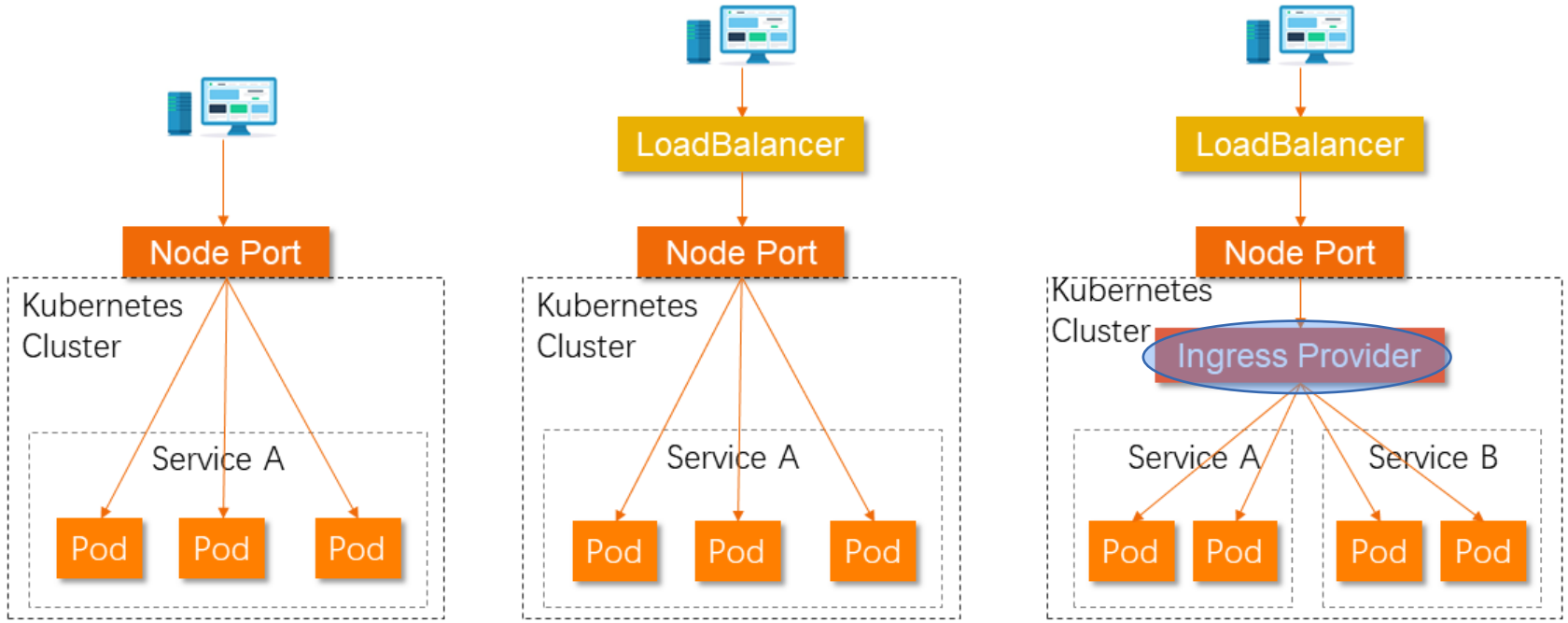

App
App
App
Operating System
Hardware
Traditional Deployment
App
App
App
App
Bin/ Library
Bin/ Library
Operating System
Operating System
Virtual Machine
Virtual Machine
Hypervisor
Operating System
Hardware
Virtualized Deployment
App
App
App
Bin/ Library
Bin/ Library
Bin/ Library
Container
Container
Container
Container Runtime
Operating System
Hardware
Container Deployment

Faisal Saleem

Published

May 8, 2020





NodePort

- Only one service can be mounted to each port.
- Nodes must have public IP address.
- The port range can only be from 30000 to 32767.

LoadBalancer

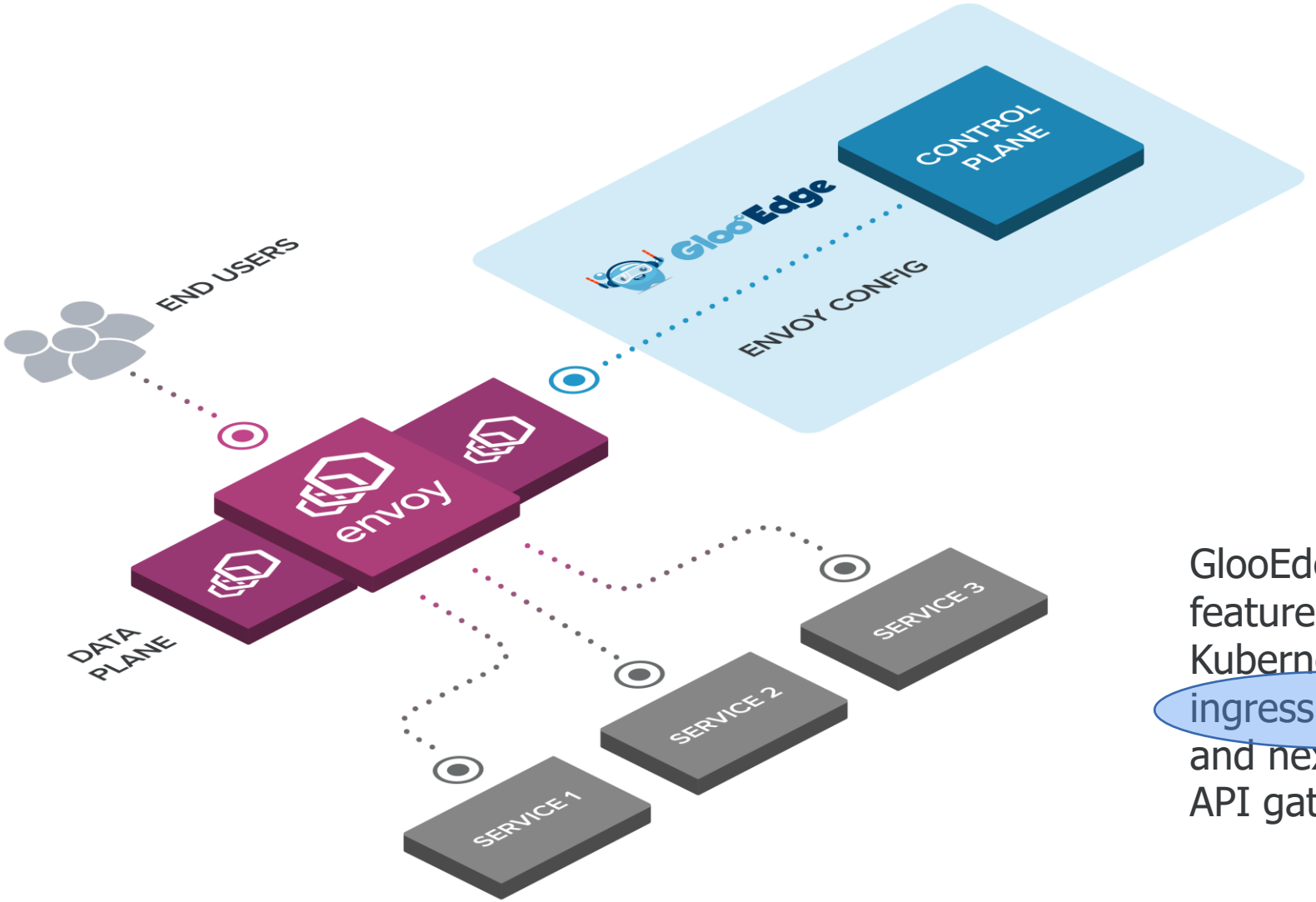
- Four-layer traffic forwarding
- Each LoadBalancer can correspond to only one service and cannot expose multiple services.

Ingress

- Seven-layer traffic management, HTTP/HTTPS
- Multiple services sharing Ingress

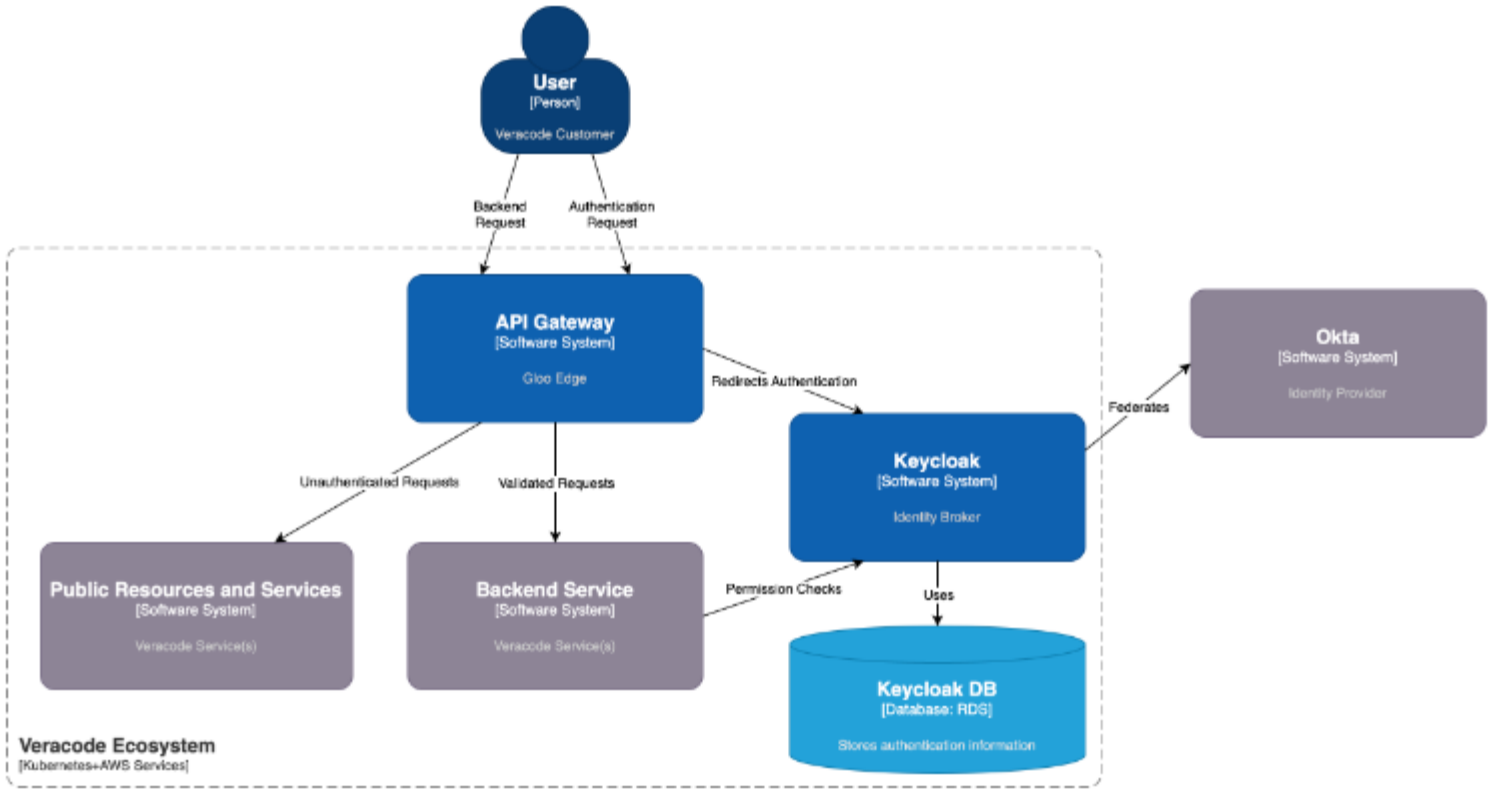
Note: Ingress ≠ Ingress Provider. Ingress is the resource for Kubernetes to define routing rules, and Ingress Provider is the implementer and executor of Ingress rules.

Fan Yang (Yangshao)

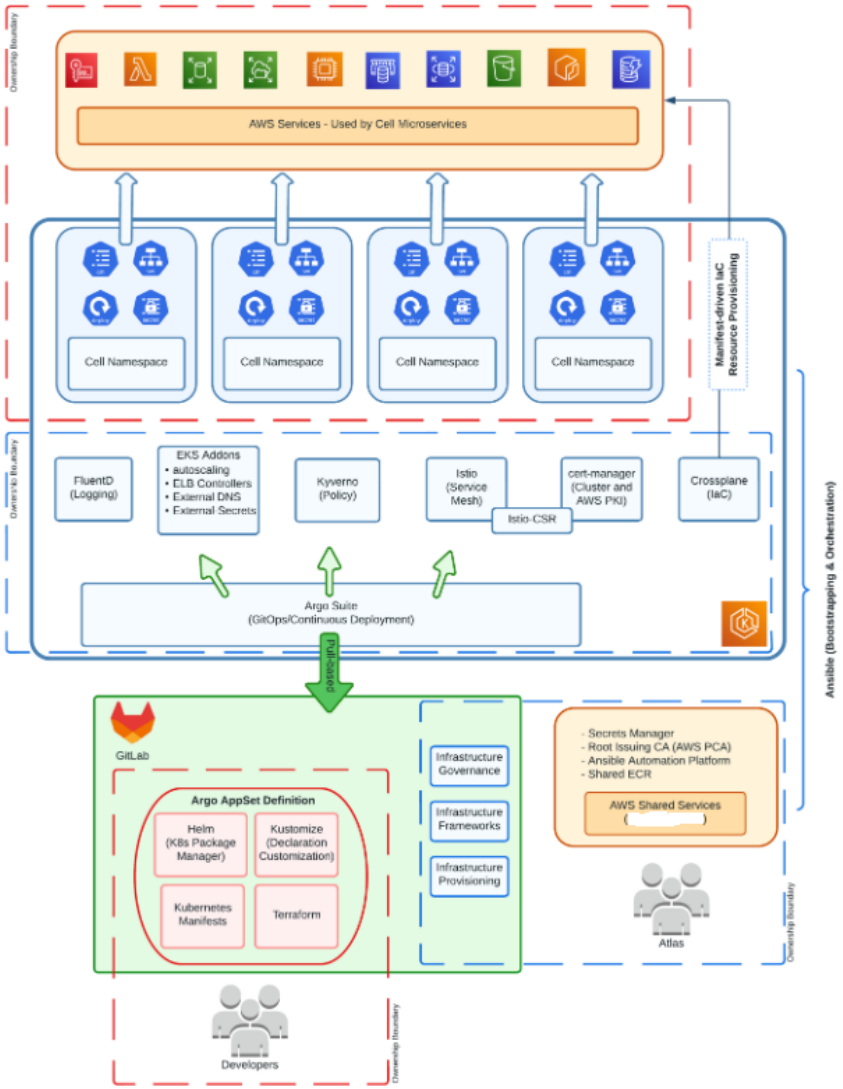


GlooEdge is a feature-rich, Kubernetes-native ingress controller, and next-generation API gateway

Architecture


User
[Person]
Veracode Customer
Backend Request
Authentication Request
API Gateway
[Software System]
Gloo Edge
Redirects Authentication
Okta
[Software System]
Identity Provider
Federates
Keycloak
[Software System]
Identity Broker
Unauthenticated Requests
Validated Requests
Public Resources and Services
[Software System]
Veracode Service(s)
Backend Service
[Software System]
Veracode Service(s)
Permission Checks
Uses
Keycloak DB
[Database: RDS]
Stores authentication information
Veracode Ecosystem
[Kubernetes+AWS Services]

System Architecture



Deployment

Deployments should be entirely automated through ArgoCD/Gitops and existing mechanisms. (See Voltron projects.)

Keycloak's database is required to be a relational database, and therefore the Mongo DB service being pioneered by the data team in this initiative is not sufficient. We will solve for this by using RDS for a Postgres DB. <Database updates?>

Lower environments will use the built in H2 database built into Keycloak. QA data will be seeded into this H2 database by scripting or Helm based custom resources.

Keycloak will run in the same cluster as the workloads to facilitate service mesh discovery. Back end services should be able to use provided Keycloak Adapters to pull claims for the principal associated with the request.

Keycloak versions should be controlled through ArgoCD/Helm configurations. This will need to be coordinated, as it will likely require downtime, and we'll need to make sure that we have a fully recoverable database before doing the upgrade.

The API Gateway versions should be controlled through ArgoCD/Helm configurations. Upgrades and downgrades should be online without downtime, if possible.

The API Gateway will be deployed into the same cluster as the workloads and services. This should be the sole point of ingress into the cluster and all other requests should be locked down to only requests from the Istio service mesh network.

Configuration

We should use Helm custom resources and Keycloak Operator to define our Realms and Clients configurationally.

The API Gateway will validate the incoming requests for authentication, but will delegate the configuration of the route tables to the back end service Helm files.

Thank you

Trent Waddington

Sr. Principal Software Engineer

TWaddington@veracode.com