



# StarlingX

A Cloud Platform for Your Distributed Edge

---

AUGUST 2022

# Ildiko Vancsa

Senior Manager, Community and Ecosystem  
Open Infrastructure Foundation

[ildiko@openinfra.dev](mailto:ildiko@openinfra.dev)  
Twitter: @IldikoVancsa

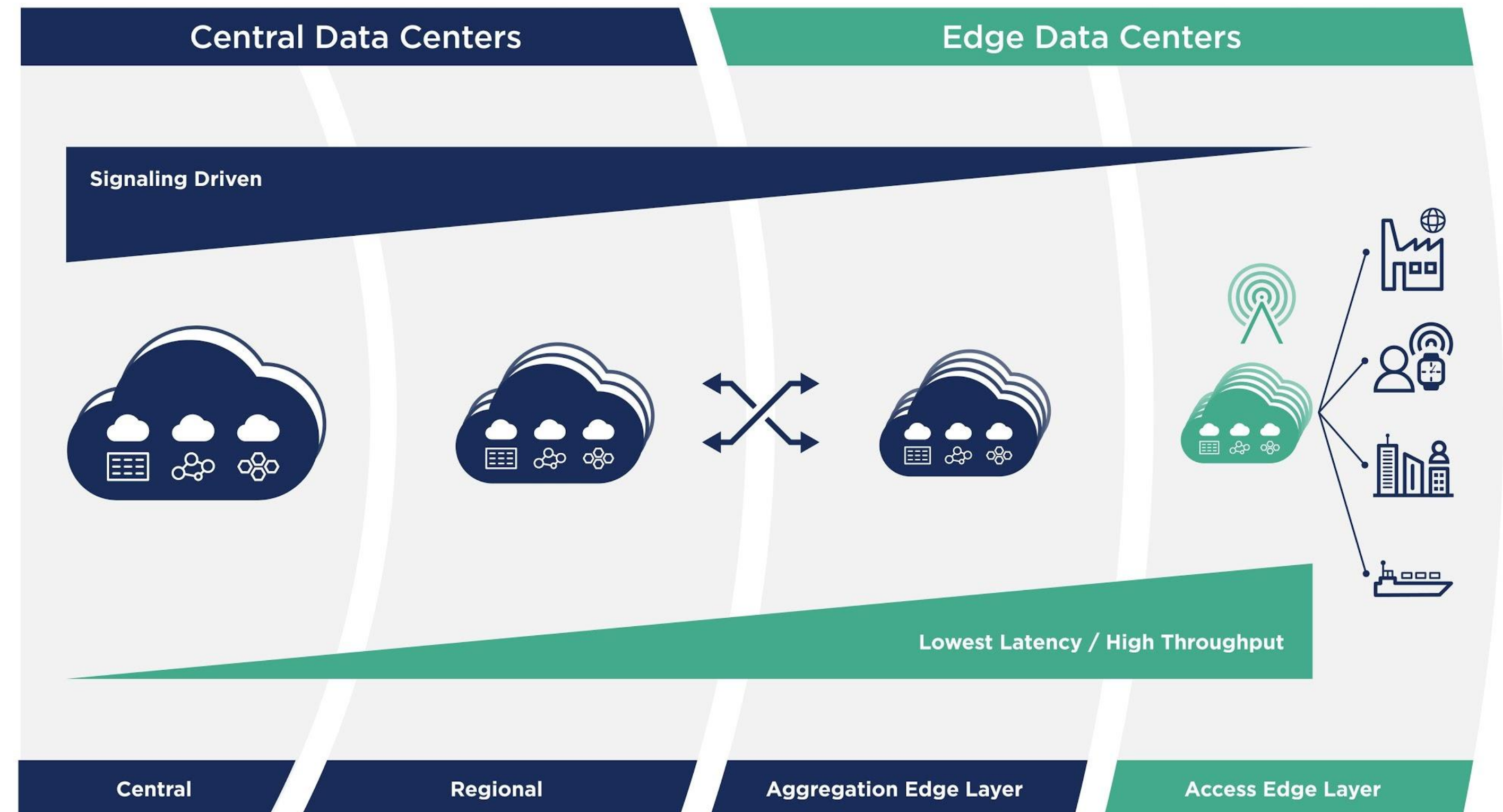
**01**

# Edge Computing

Use Cases, Challenges and Choices

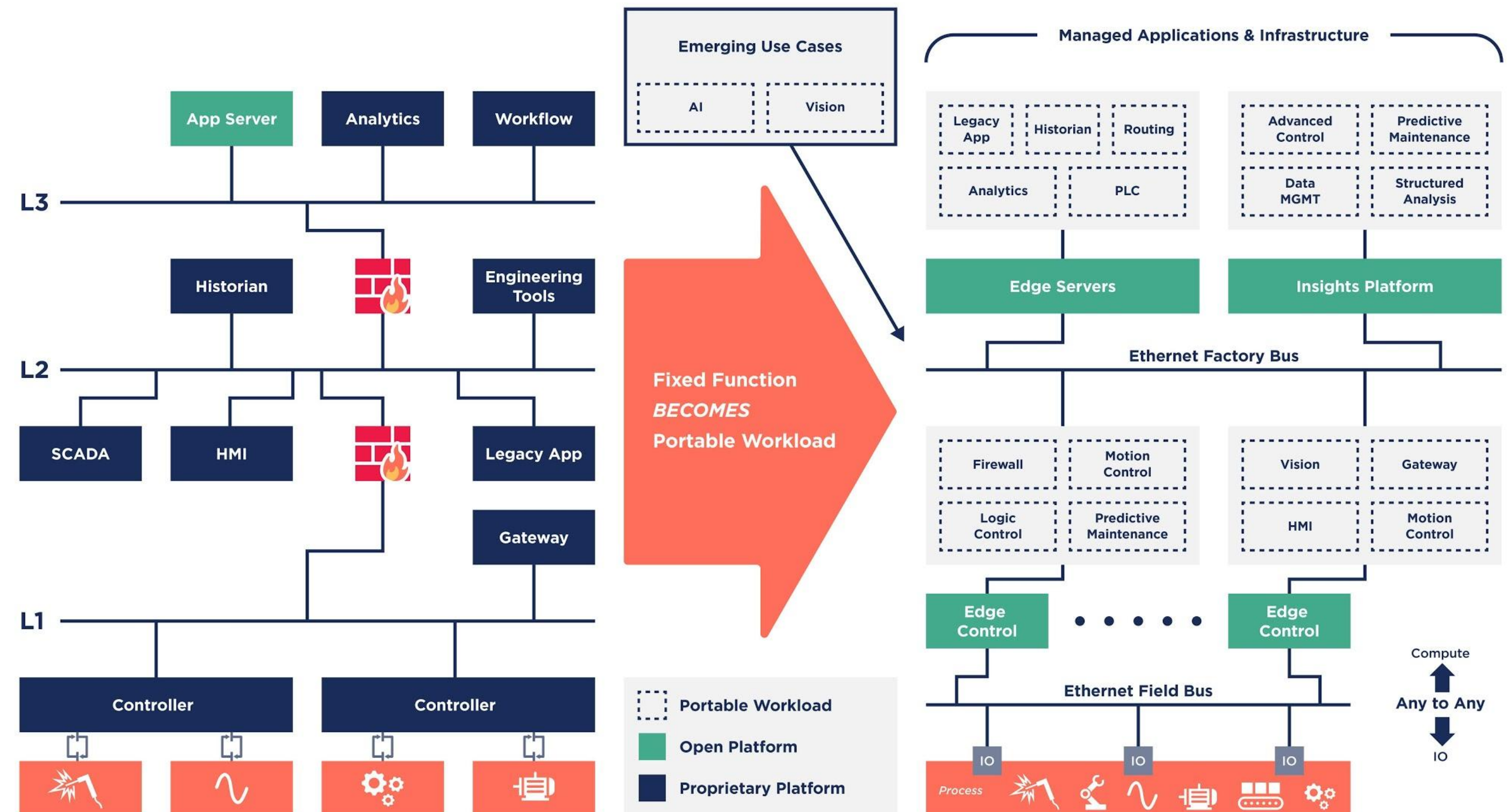
# Use Cases - Telecommunications and 5G

- Telecom providers are **pioneers** in the edge computing space
- Providing **connectivity**
- Offering **new services** by better utilizing resources at the edge
  - uCPE use case and more
- **Key requirements**
  - High bandwidth and low latency locally
  - Support for accelerators
  - Different datacenter and edge sizes/types
  - High number of edge datacenters
  - Remote 'Day 0', 'Day 1' and 'Day 2' operations



# Use Cases - Industrial IoT

- Industry 4.0 and 5.0
- **Digitalizing** factory automation
- Edge computing reaches **new industry segments**
- **Key requirements**
  - High bandwidth locally
  - Real-time operation
  - Tolerate network disruption to other locations
  - Support for GPU-s



# Emerging Use Cases

- **Green Energy**

- Moving from coal to wind and solar-generated power
- The **grid** is turning into a **distributed system**

- **Cars and Vehicles**

- Connected car
  - **Software upgrade as a CI/CD**-type workflow without taking the car to the shop
- V2V communication and **mission-critical workloads**

- **Cloud Robotics**

- **Factory floor** is becoming a **distributed cloud environment**
- Not just controlling the robots, but also **re-configuring** them **on the fly**

# Production Challenges

- **Mixed workloads on heterogenous platforms**
  - Complexity
  - Large and organically growing systems
- **Automation**
  - Deployment
  - Scale out and scale in
  - Patching
- **Infrastructure**
  - VMs, containers and bare metal
  - Interoperability

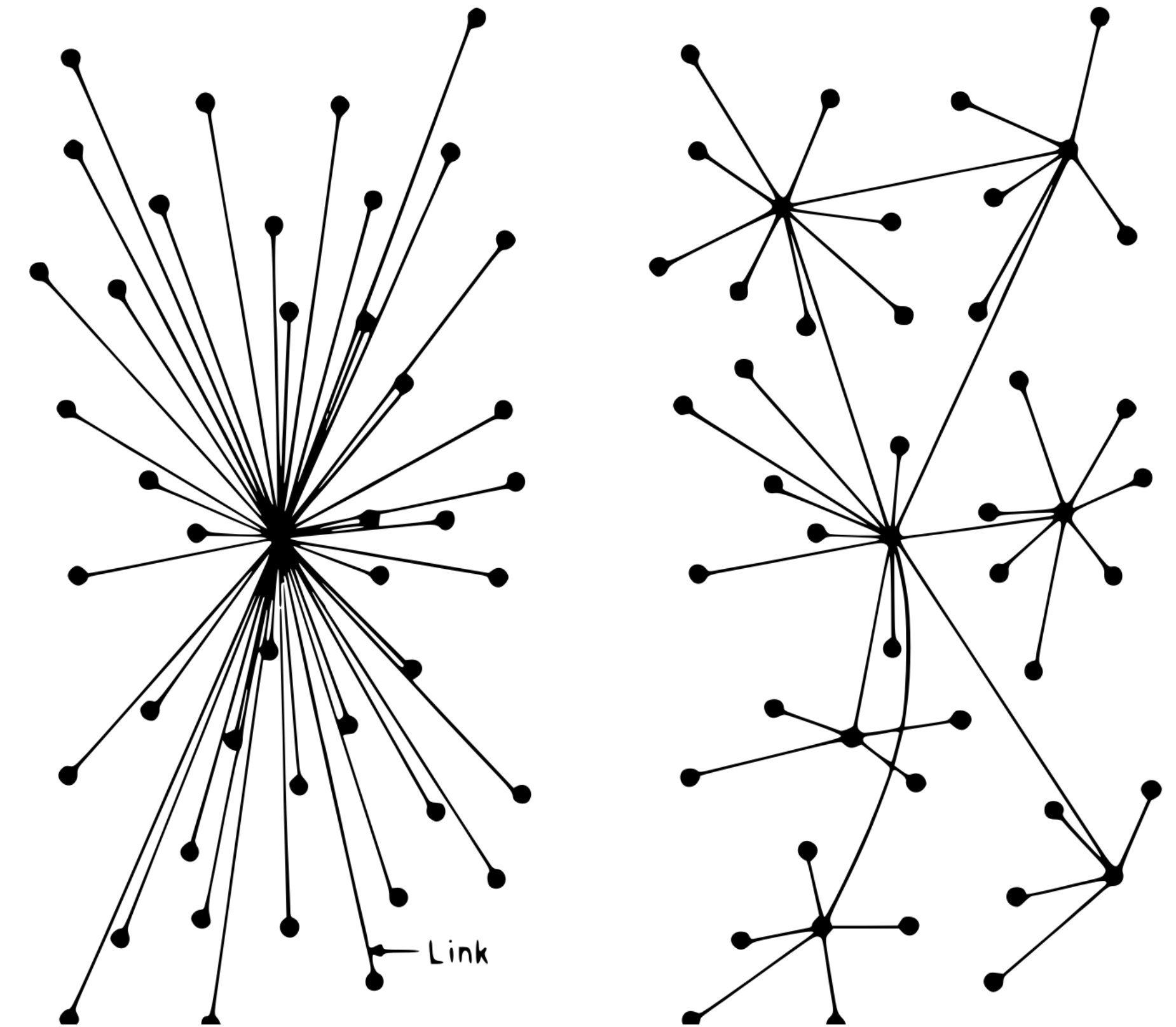


[https://vignette.wikia.nocookie.net/fifth-element/images/6/6f/1655344\\_10152164967867236\\_2110503604\\_o.jpg/revision/latest?cb=20180909001342](https://vignette.wikia.nocookie.net/fifth-element/images/6/6f/1655344_10152164967867236_2110503604_o.jpg/revision/latest?cb=20180909001342)

**Open source infrastructure** and **standards** are more crucial than ever

# Architecture Choices

- End-to-end architecture
  - From **Core** to the **Edge**
- There is **no one-size fits all** solution
  - Flexibility and scale drive implementation
- **Autonomy**
  - Unreliable networks
  - What are the critical functions?
- Accepting complexity
  - **Automation** is key!



**Centralized** and **distributed** models are both viable solutions



02

## StarlingX

Solve the operational problem of deploying and managing distributed networks

# Project Overview

# Let Me Introduce StarlingX

- Top-level Open Infrastructure Foundation project
- Software stack providing critical infrastructure and management services for edge cloud applications
- Frequent releases
  - <https://opendev.org/starlingx>
  - <http://mirror.starlingx.cengn.ca/mirror/starlingx/release/>
- Growing community
  - Inviting users, operators and developers to try out the software and participate in the community

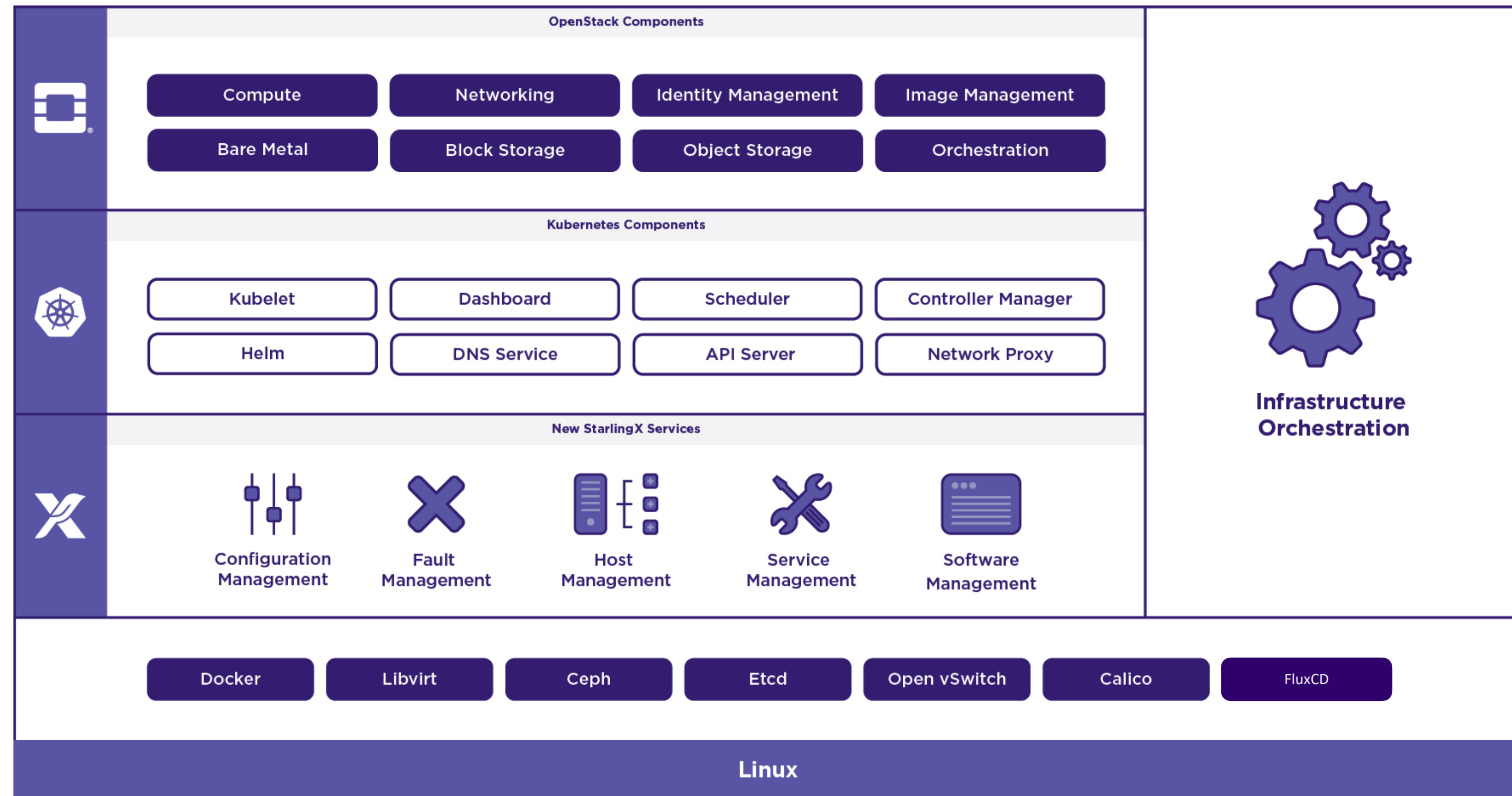
# StarlingX – Edge Virtualization Platform

StarlingX provides a **deployment-ready, scalable, highly reliable** Edge infrastructure software platform

Services from the StarlingX virtualization platform focus on

- **Easy deployment**
- **Low touch manageability**
- **Rapid response to events**
- **Fast recovery**

A complete edge orchestration platform for bare metal, VM and container workloads

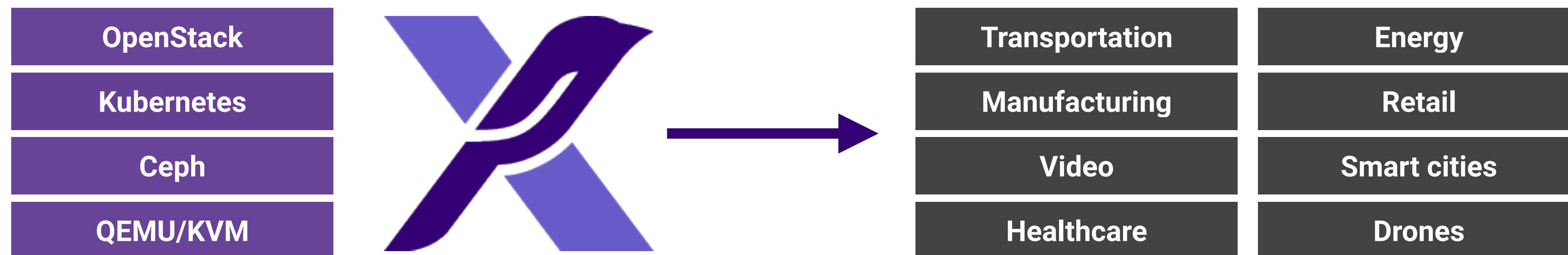


There are more OpenStack and Kubernetes components used than represented in this diagram.

# Edge Use Case Frontiers

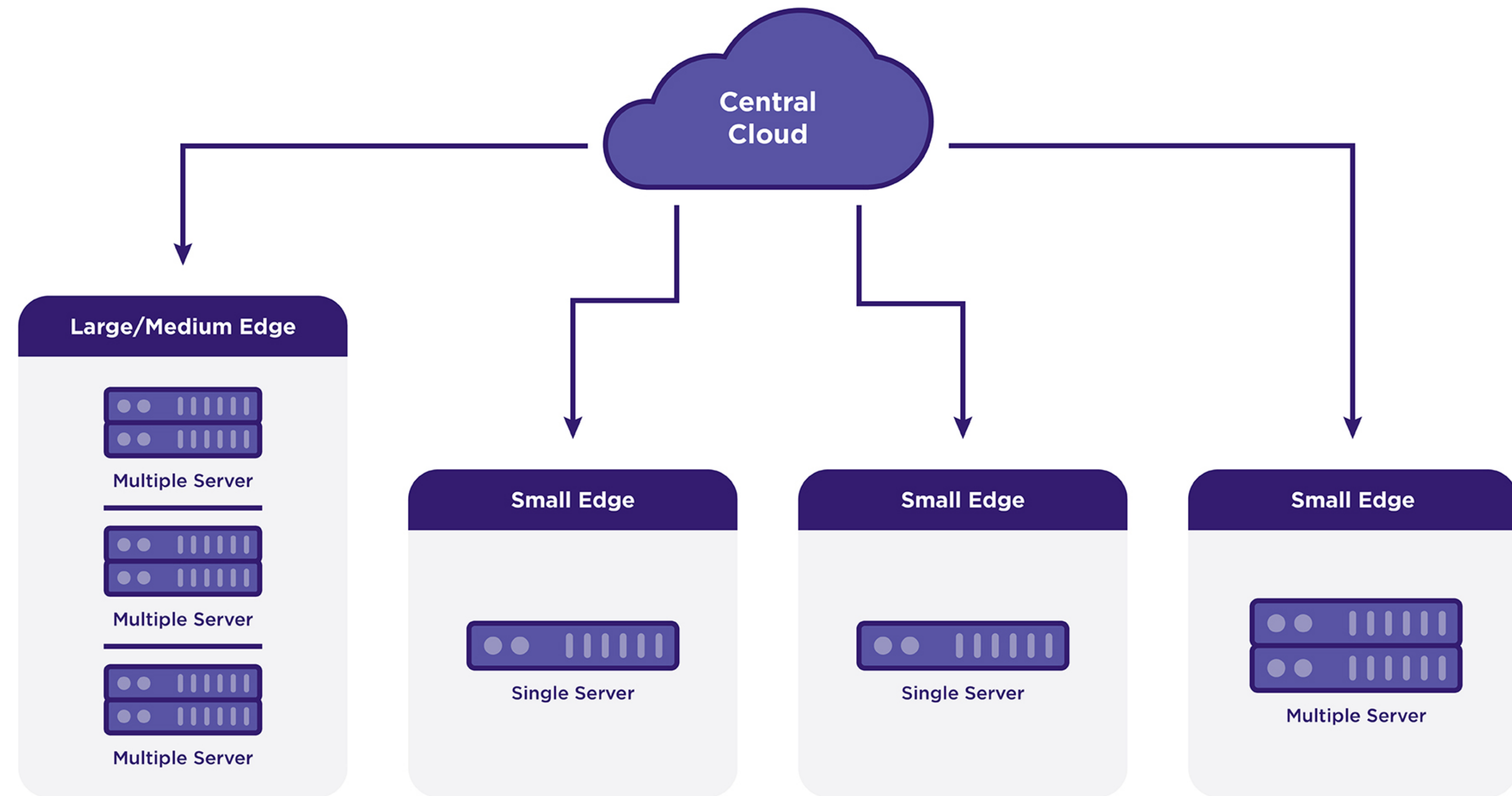
## Re-Configure Proven Cloud Technologies for Edge Compute

- Orchestrate system-wide
  - Deploy and manage Edge clouds, share configurations
- Simplify deployment to geographically dispersed, remote Edge regions



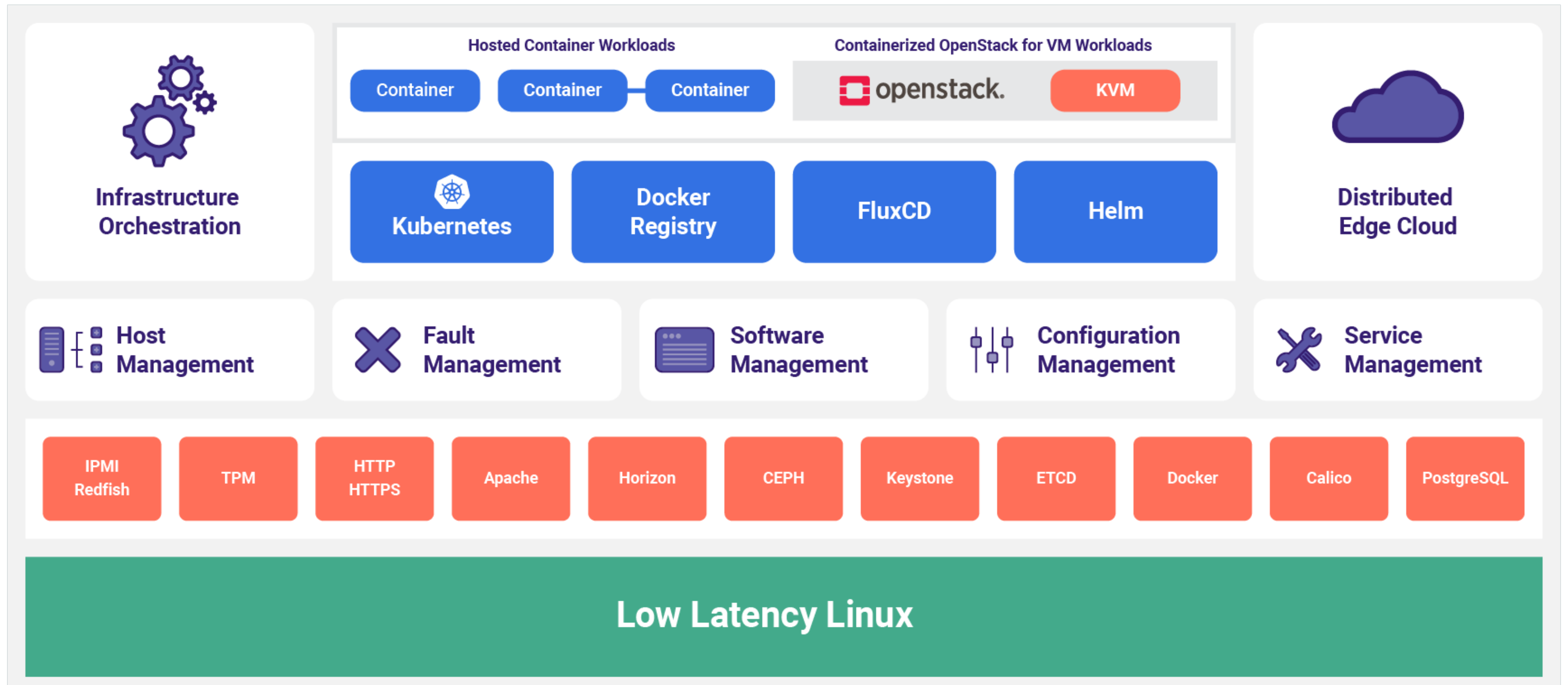
# StarlingX Edge Deployments

- Geographically distributed multi-region deployment,
- Central Datacenter providing Orchestration and Synchronization Services,
- Geographically distributed Edge Sites of various sizes



# StarlingX Technology

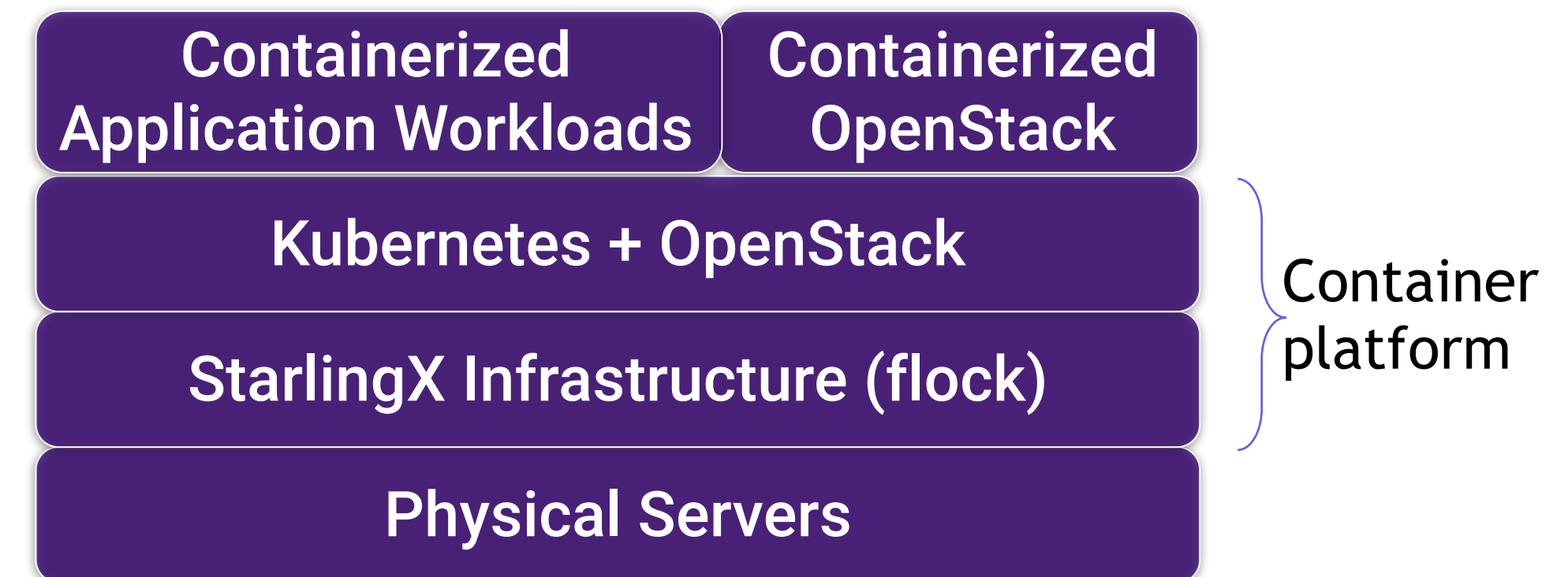
# Distributed Edge Cloud Native Platform





# StarlingX Evolution

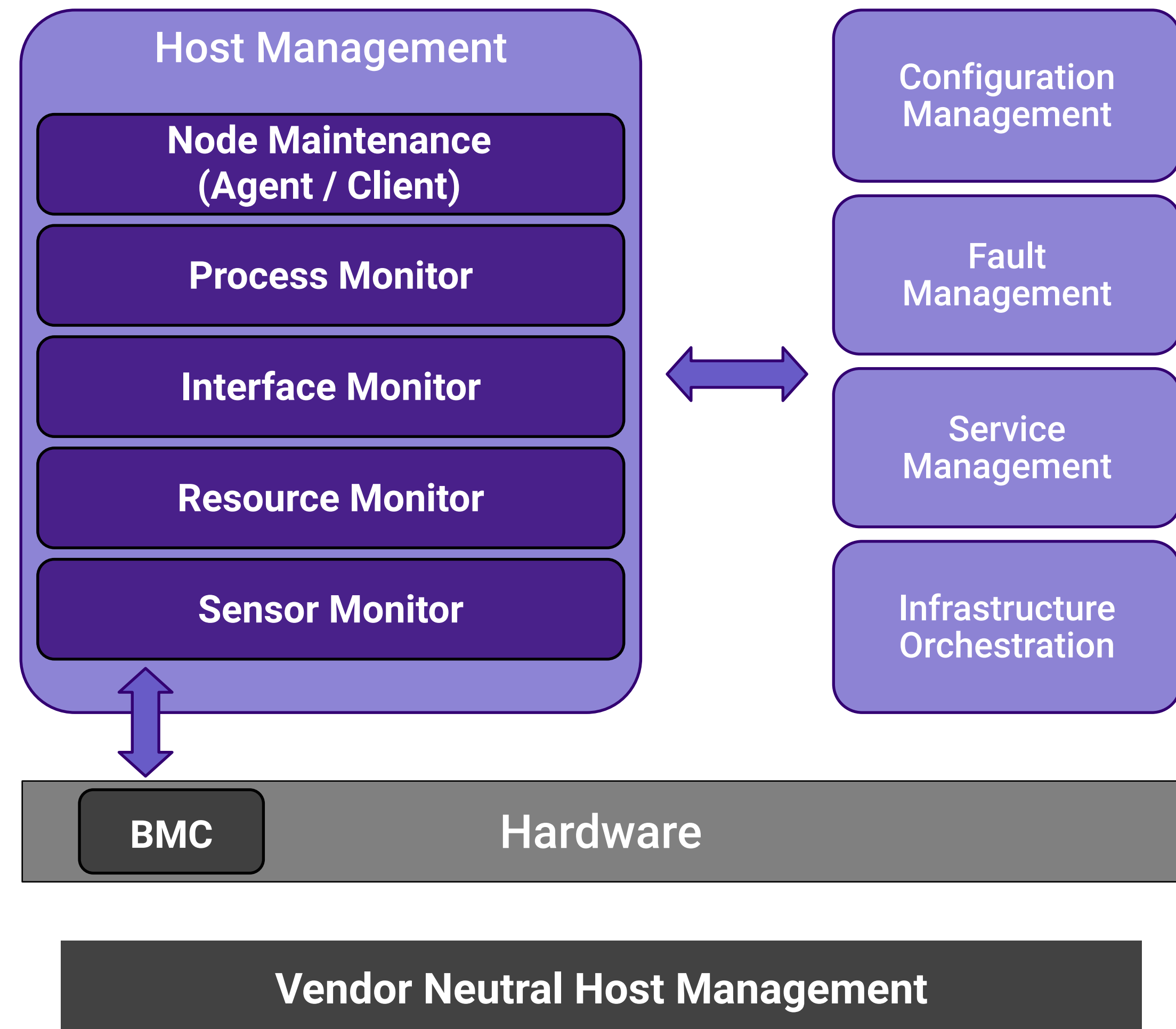
- A hardened cloud-native platform integrating OpenStack and Kubernetes on dedicated physical servers
- Containerized OpenStack services based on the latest release
- Closely aligned with the current OpenStack code base
  - The StarlingX and OpenStack communities are working together on Edge related enhancements
- Kubernetes-based edge sites for containerized workloads



# Infrastructure Management “The Flock”

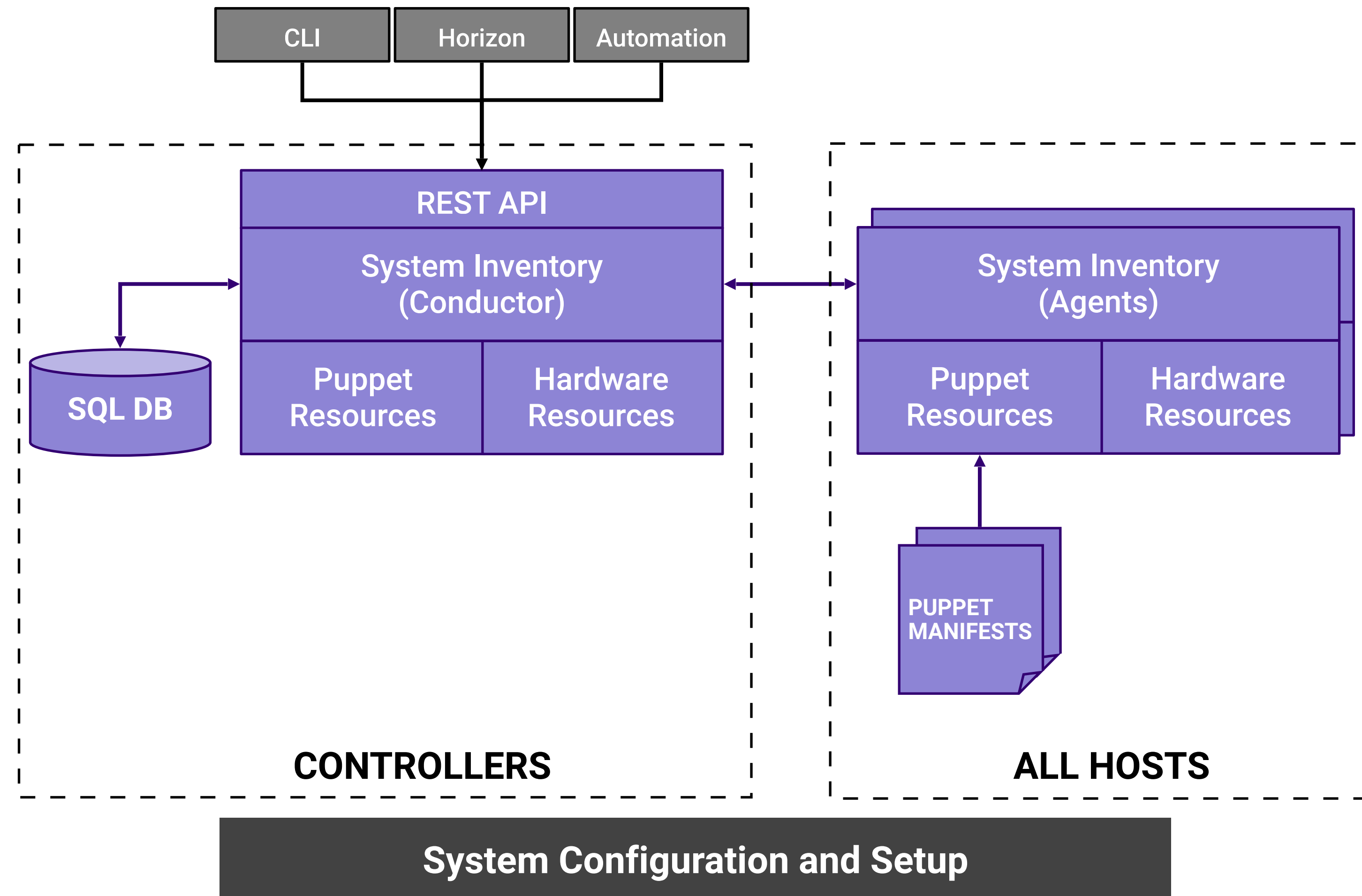
# Host Management

- Full life-cycle management of the host
- Detects and automatically handles host failures and initiates recovery
- Monitoring and fault reporting for
  - Cluster connectivity, critical process failures
  - Resource utilization thresholds, interface states
  - H/W fault / sensors, host watchdog
  - Activity progress reporting
- Interfaces with board management (BMC)
  - For out of band reset
  - Power-on/off
  - H/W sensor monitoring



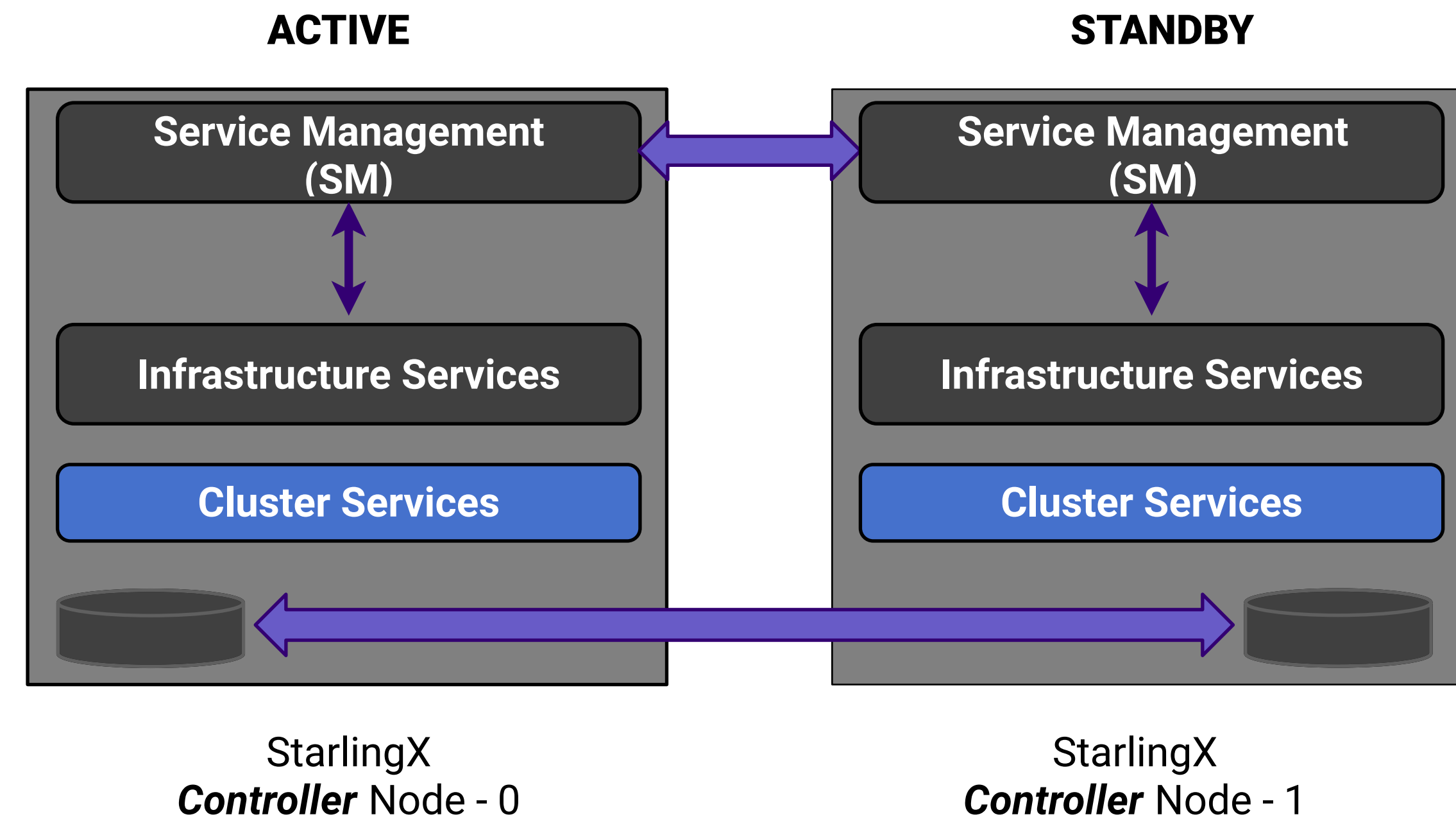
# Configuration Management

- **Manages installation**
  - Auto-discover new nodes
  - Manage installation parameters (i.e. console, root disks)
  - Bulk provisioning of nodes through XML file
- **Nodal Configuration**
  - Node role, role profiles
  - Core, memory (including huge page) assignments
  - Network Interfaces and storage assignments
- **Inventory Discovery**
  - CPU/cores, SMT, processors, memory, huge pages
  - Storage, ports
  - GPUs, storage, Crypto/compression H/W



# Service Management

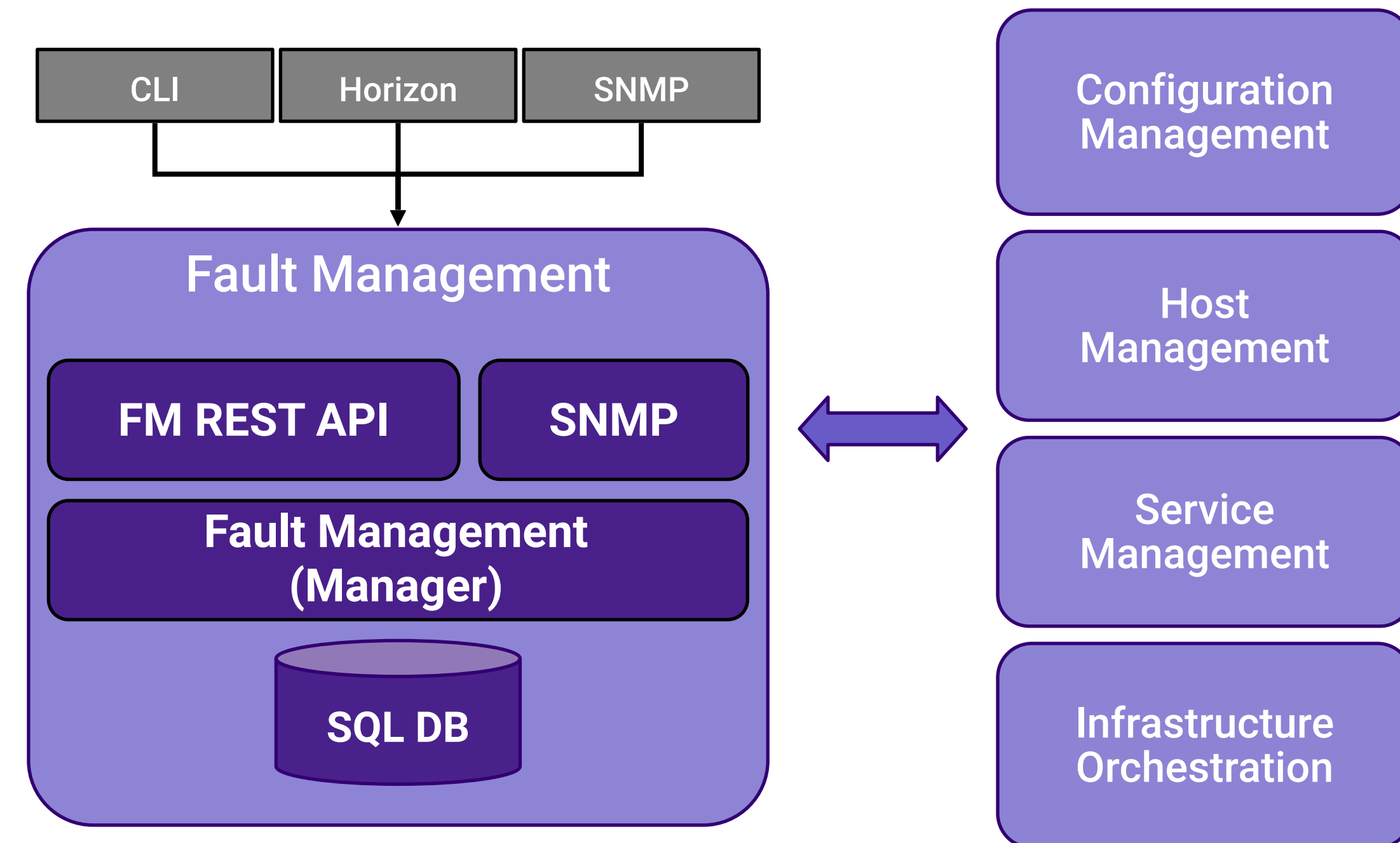
- High availability manager
  - Redundancy model can be N+M or N across multiple nodes
  - Currently 1+1 HA Controller Cluster
- Uses multiple messaging paths to avoid split-brain communication failures
  - Up to 3 independent communication paths
  - LAG can also be configured for multi-link protection of each path
  - Messages are authenticated using HMAC SHA-512 if configured / enabled on an interface-by-interface basis
- Active or passive monitoring of services
- Allows for specifying the impact of a service failure



**High Availability for Critical Infrastructure**

# Fault Management

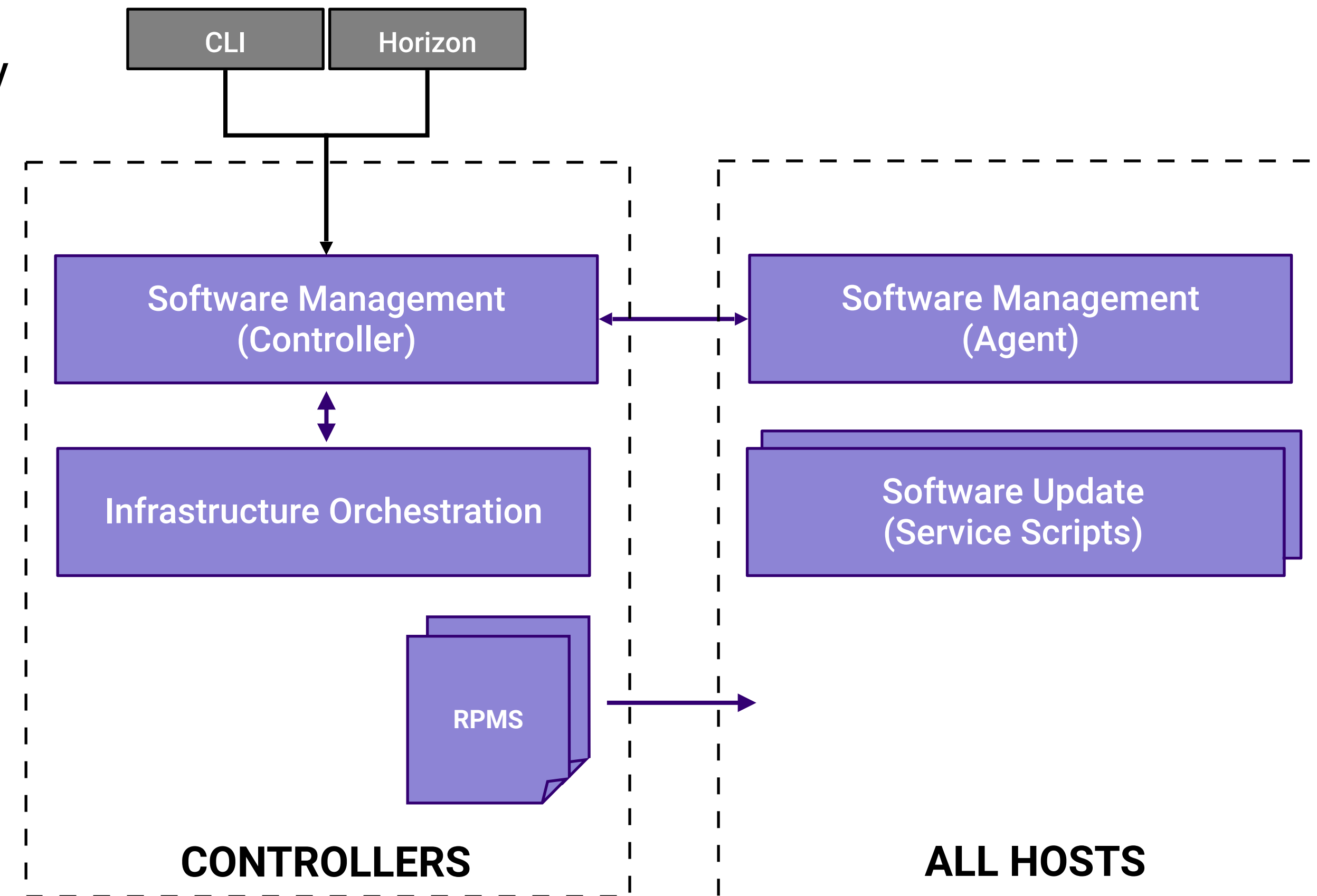
- Framework for infrastructure services via API
  - Set, clear and query customer alarms
  - Generate customer logs for significant events
- Maintains an Active Alarm List
- Provides REST API to query alarms and events
- Support for alarm suppression
- Operator alarms
  - On platform nodes and resources
  - On hosted virtual resources
- Operator logs - Event List
  - Logging of set/clear of alarms
  - Related to platform nodes and resources
  - Related to hosted virtual resources



**Fault Alarming and Logging**

# Software Management

- Automated deploy of software updates for security and/or new functionality
- Integrated end-to-end rolling upgrade solution
  - Automated, low number of steps
  - No additional hardware required for upgrade
  - Rolling upgrade across nodes
- In-service and reboot required patches supported
  - Reboot required for kernel replacement etc.
  - VM live migration is used for patches that require reboot
- Manages upgrades of all software
  - Host OS changes
  - New / upgraded StarlingX service software
  - New / upgraded OpenStack software

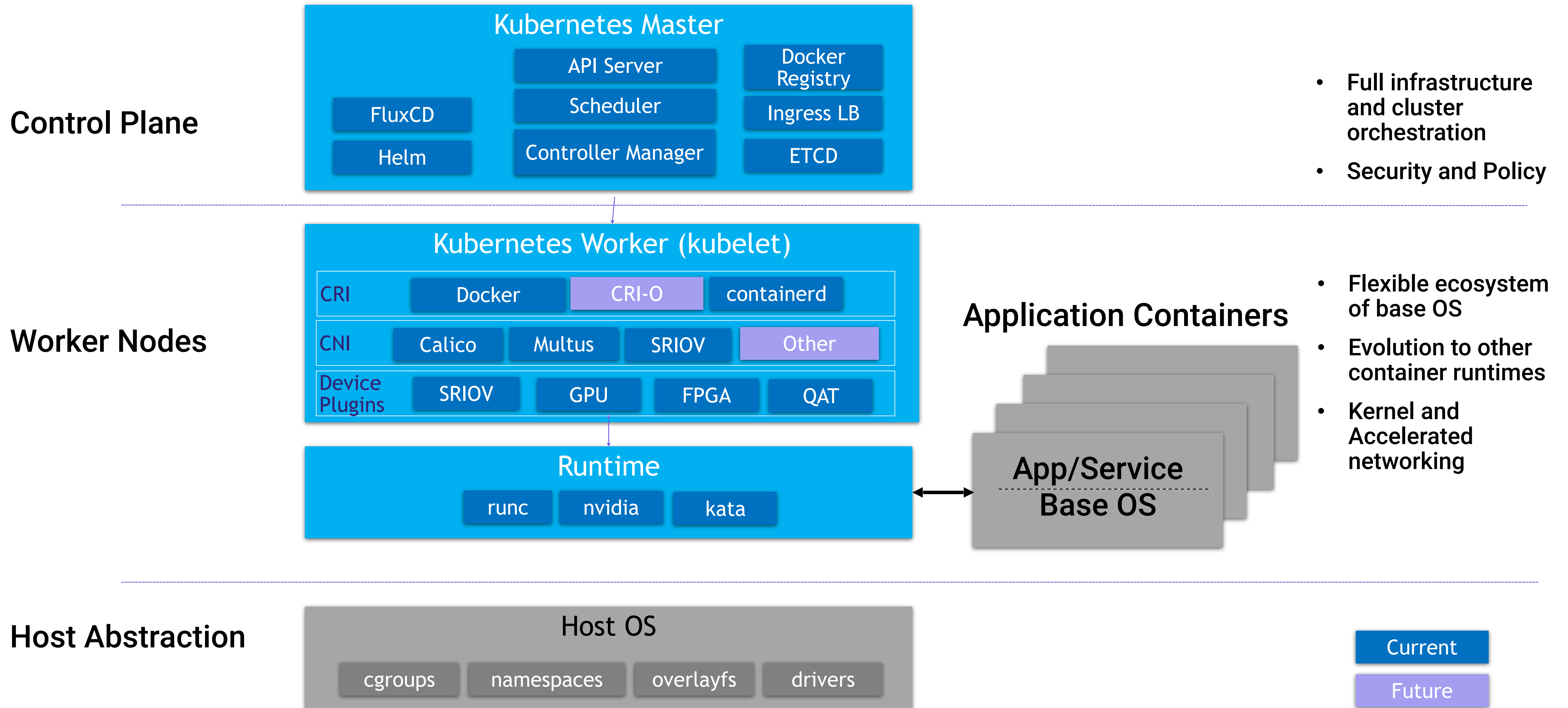


**Software Upgrades and Patching**

# Container Platform



# Kubernetes Cluster Software Components

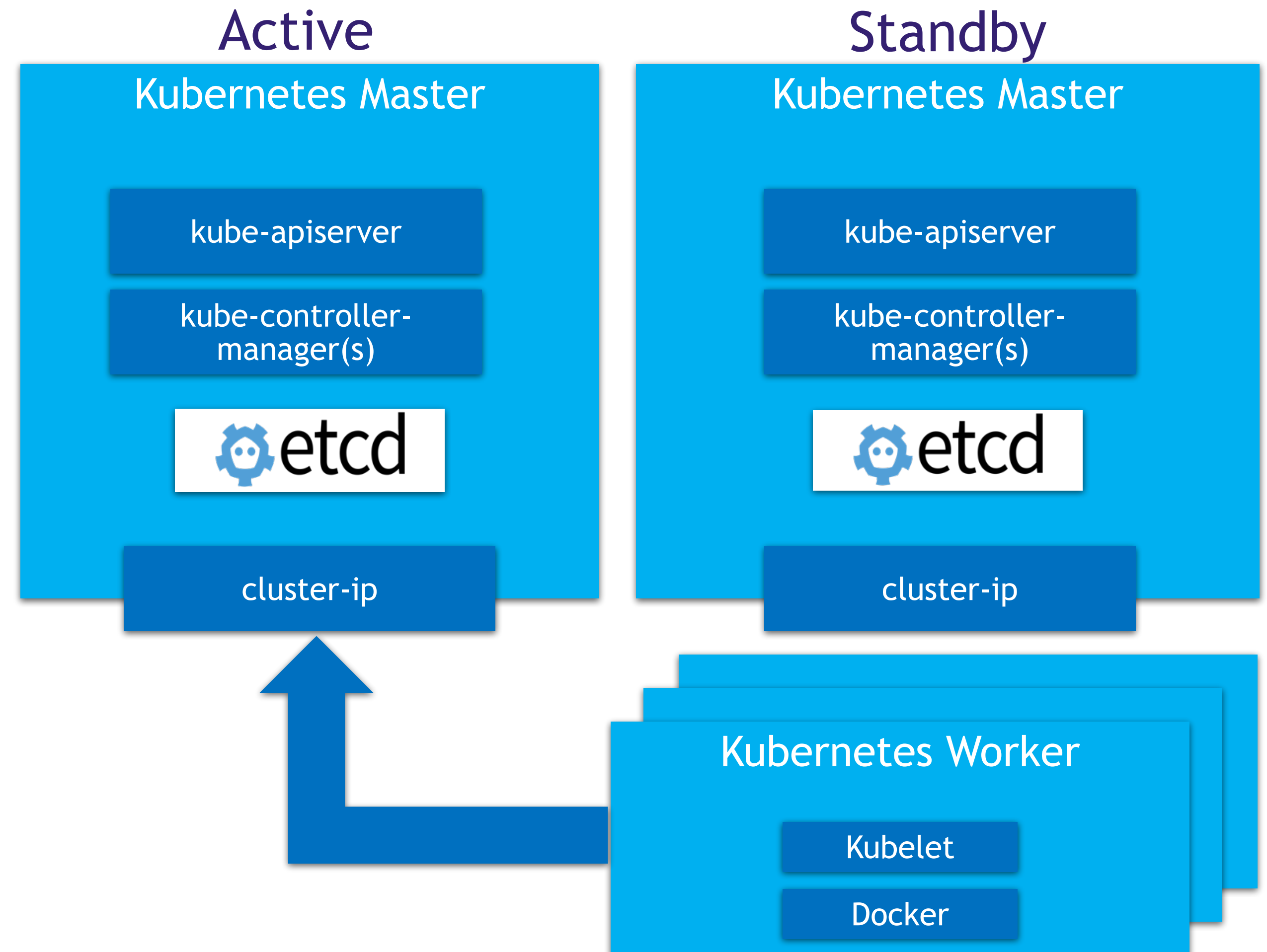


- Full infrastructure and cluster orchestration
- Security and Policy

- Flexible ecosystem of base OS
- Evolution to other container runtimes
- Kernel and Accelerated networking

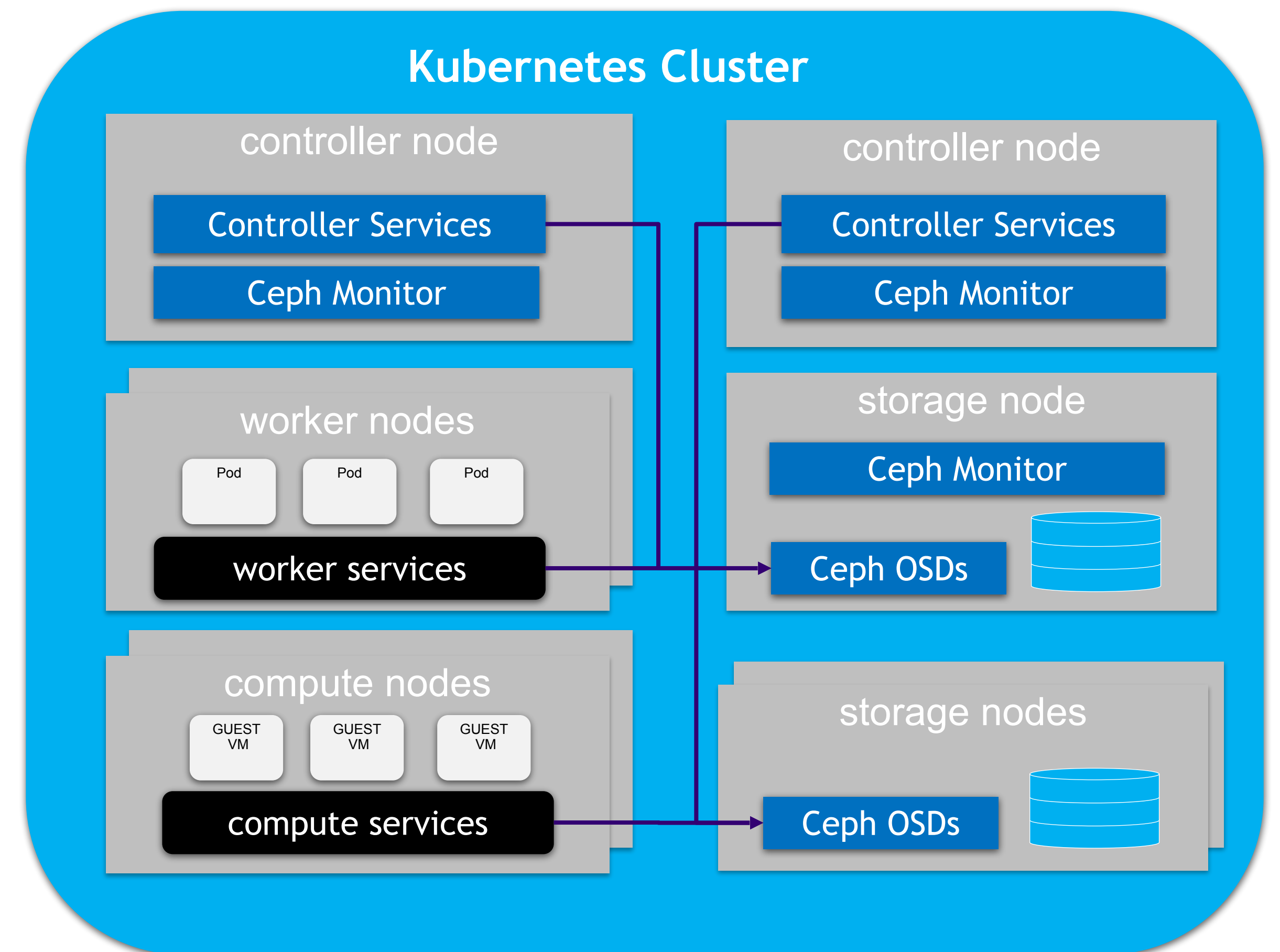
# Kubernetes Deployment Architecture

- Kubernetes deployed in a highly available configuration
  - Deployed in a 1:1 service model
  - All-in-One Simplex/Duplex deployments supported using same service management
- Requests directed to actives instances via cluster floating IP address
- DRBD backed file system for redundant persistent storage
- Service availability and activity managed by Service Management (SM)
  - Handles HA sparing of individual services
  - Actively monitors host, service and network availability
  - Mitigates split-brain scenarios



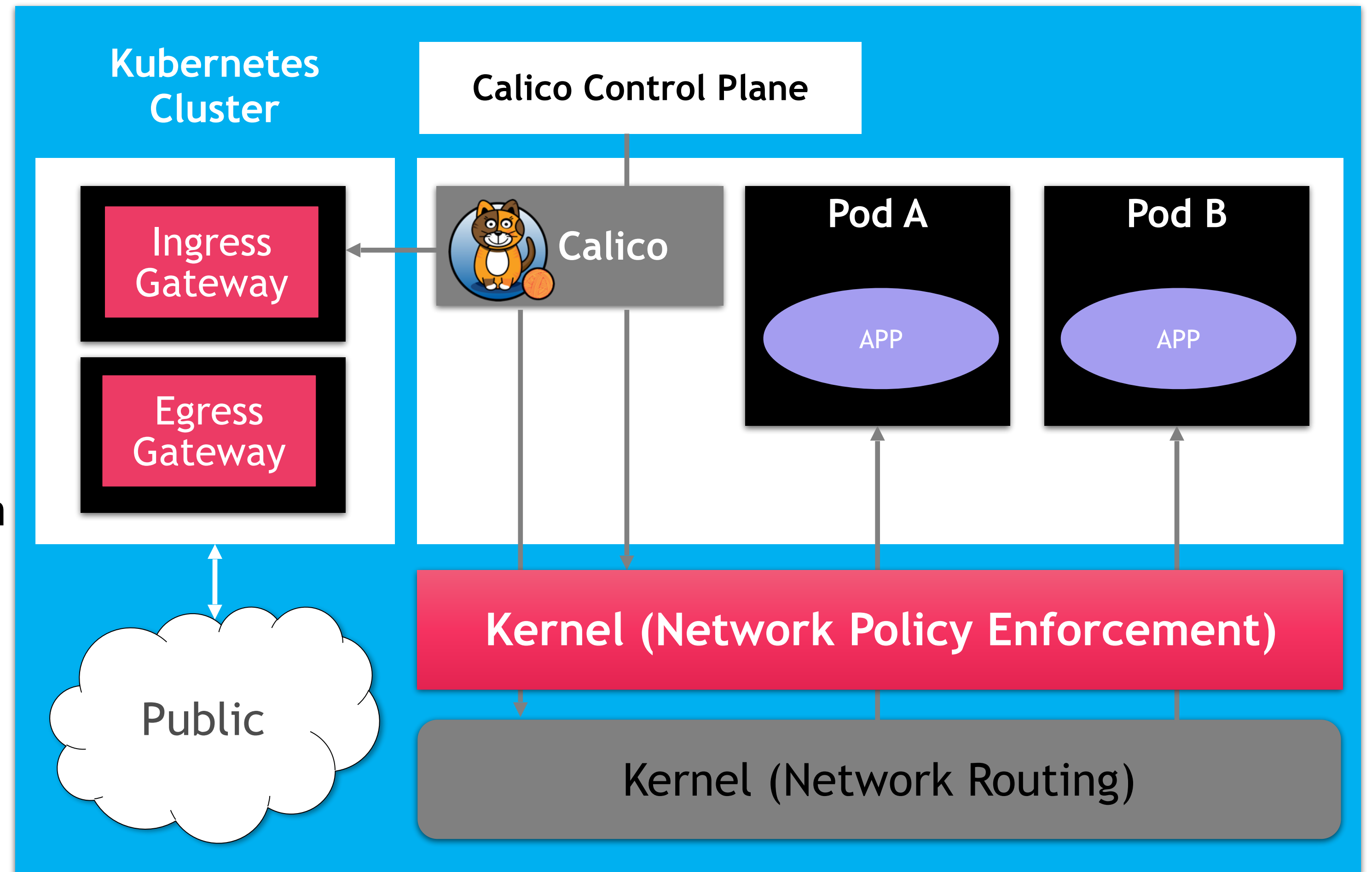
# Cluster Persistent Storage

- Ceph uniquely delivers object, block, and file storage in one unified system
- Highly scalable and highly available deployment with distributed Ceph monitors and Object Storage Devices (OSD) for data replication
  - Automatic cluster storage deployment and replication
  - Unified storage solution for all deployments: AIO-SX/DX, Standard, Multi-cloud
  - Fully managed Ceph Cluster Map (hyper-scale)
- **Kubernetes persistent storage provided by Ceph's RADOS Block Devices (RBD) provisioner**
  - Persistent Volumes (PVs) and Claims (PVCs)
  - Default Storage Class
- Support for Rook to add additional storage backend options
- OpenStack backend storage solution for services: Glance, Cinder, Swift, Nova



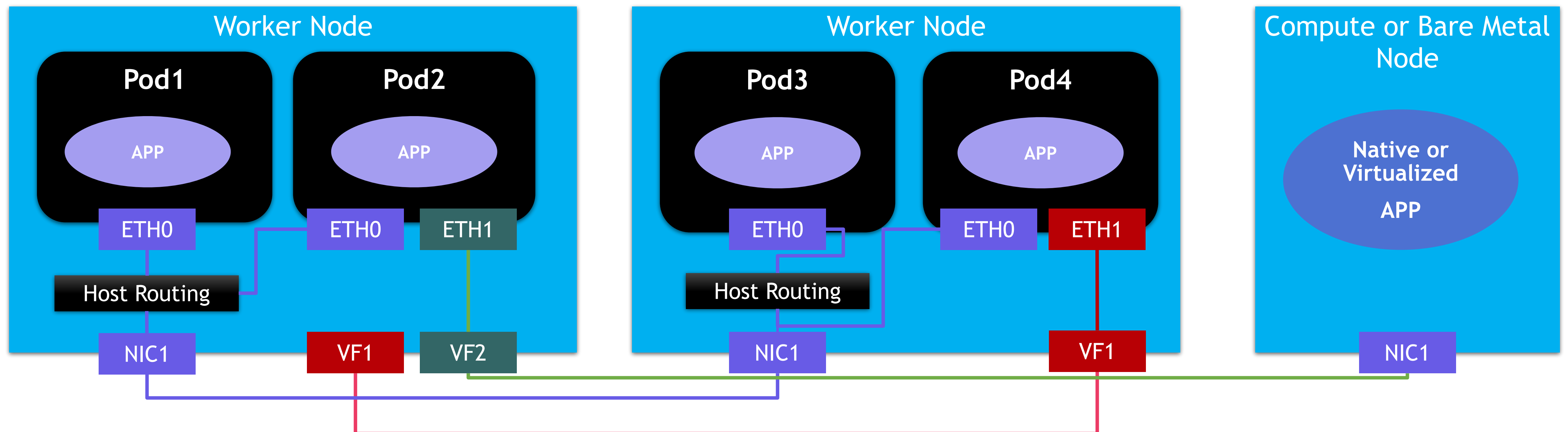
# Kubernetes Cluster Networking

- Calico provides a pure L3 fabric solution for interconnecting containers
- Calico leverages the Linux kernel for routing and policy enforcement
- Calico leverages Border Gateway Protocol (BGP) for control plane
- Calico leverages Open Standards and is a full Open Source network solution
- Calico is highly scalable, and is operator and policy friendly:
  - No overlay, no tunnelling, no VRF tables (no overhead) – pure routing
  - Access Controls enforced through L3/L4 security policies



# Kubernetes Accelerated Networking

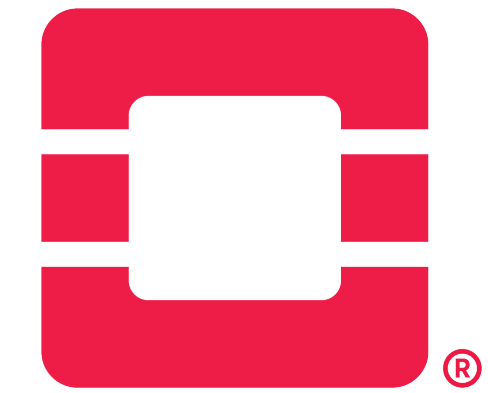
- Multus, SR-IOV, DPDK
- Kubernetes managed accelerated network devices (via Device and CNI plugins)
- Containers bind the driver to the Virtual Function (VF) or DPDK devices directly
- No host routing or switching is involved for SRIOV and provides the best direct IO



# OpenStack

# OpenStack Deployment

- OpenStack is deployed as a containerized Kubernetes application
  - OpenStack control plane running in pods
  - OpenStack virtual machines running on host
  - Leverages Kubernetes' strengths to manage, scale and update the OpenStack services
- Deployed using Helm (using OpenStack-Helm charts)
- StarlingX provides application APIs to install and configure the containerized OpenStack application
  - Application tarball contains helm charts and manifest files for StarlingX
  - Automatic generation of helm configuration values based on system configuration
  - User can easily customize helm configuration of OpenStack Services



# Supported OpenStack Services

- Configuration optimized and system validated within StarlingX
  - Keystone, Nova, Neutron, Glance, Cinder, Horizon, Heat, Barbican, Ironic
  - Telemetry
    - Ceilometer, Gnocchi, Panko, Aodh



# Day 2 Configuration Changes

- Configuration changes can be applied after the application has been deployed
- Update the helm chart overrides
  - system helm-override-update ...
- Reapply the application
  - system application-apply ...
- Only charts impacted by the configuration change will be updated

# Distributed Cloud

# Distributed Cloud Overview

- Introduced in StarlingX 3.0
- Heterogeneous Distribution of Kubernetes and OpenStack Clouds
- Central Cloud (System Controller)
  - Hosting shared services
  - System-wide infrastructure orchestration functions
- Remote, geographically dispersed edge clouds
  - Communication with the System Controller node through REST APIs/L3
  - Running a control plane for autonomous operation
- In line with the [Distributed Control Plane](#) reference architecture model defined by the OpenInfra Edge Computing Group

# Distributed Cloud - System Controller

- Centralized deployment of container platform on sub-clouds
  - Automated and declarative configuration
- Sub-cloud health monitoring and management
- Synchronized User Authentication & Authorization with Keystone
- Centralized Docker registry for infrastructure and applications
- Centralize Horizon dashboard - single pane of glass
- Configuration portal for shared platform data
  - DNS, NTP/PTP, API Firewall, SNMP, ...

# Releases

# 7.0 Release Overview

- Released in August, 2022
- Focus on scalability, security and real-time functions
  - Precision Time Protocol enhancements
  - Distributed Cloud:
    - Local installation support
    - Increased scalability
  - Security:
    - Audit logging and more
- Code: <https://opendev.org/starlingx/>
- ISO image: <http://mirror.starlingx.cengn.ca/mirror/starlingx/release/7.0.0/>

# Precision Time Protocol (PTP)

- Standard protocol (IEEE 1588-2008, IEEE 1588-2019)
  - Synchronize clocks in a computer network
  - On a Local Area Network (LAN) the clock accuracy is in the sub-microsecond range
  - Leader - Follower architecture
- **Widely used to synchronize**
  - financial transactions
  - mobile phone tower transmissions
  - and more

# PTP Features and Enhancements

- **Supported since StarlingX 3.0**
  - Fulfills the needs of mission-critical workloads
  - Configuration options include
    - 'Time Stamping Mode'
    - 'Network Transport'
    - 'Delay Mechanism'
- **7.0 additions**
  - 3.1.1 version of the base linuxptp package
  - Host-specific configurations
  - Support for NICs that provide Synchronous Ethernet (SyncE) function



# Security - Features and Enhancements

- Initial implementation of Pod Security Admission Controller
  - It will replace Pod Security Policies in a future release
- **Security Audit Logging**
  - Platform commands
  - Kubernetes API calls
- **cert-manager upgrade to 1.7.1**

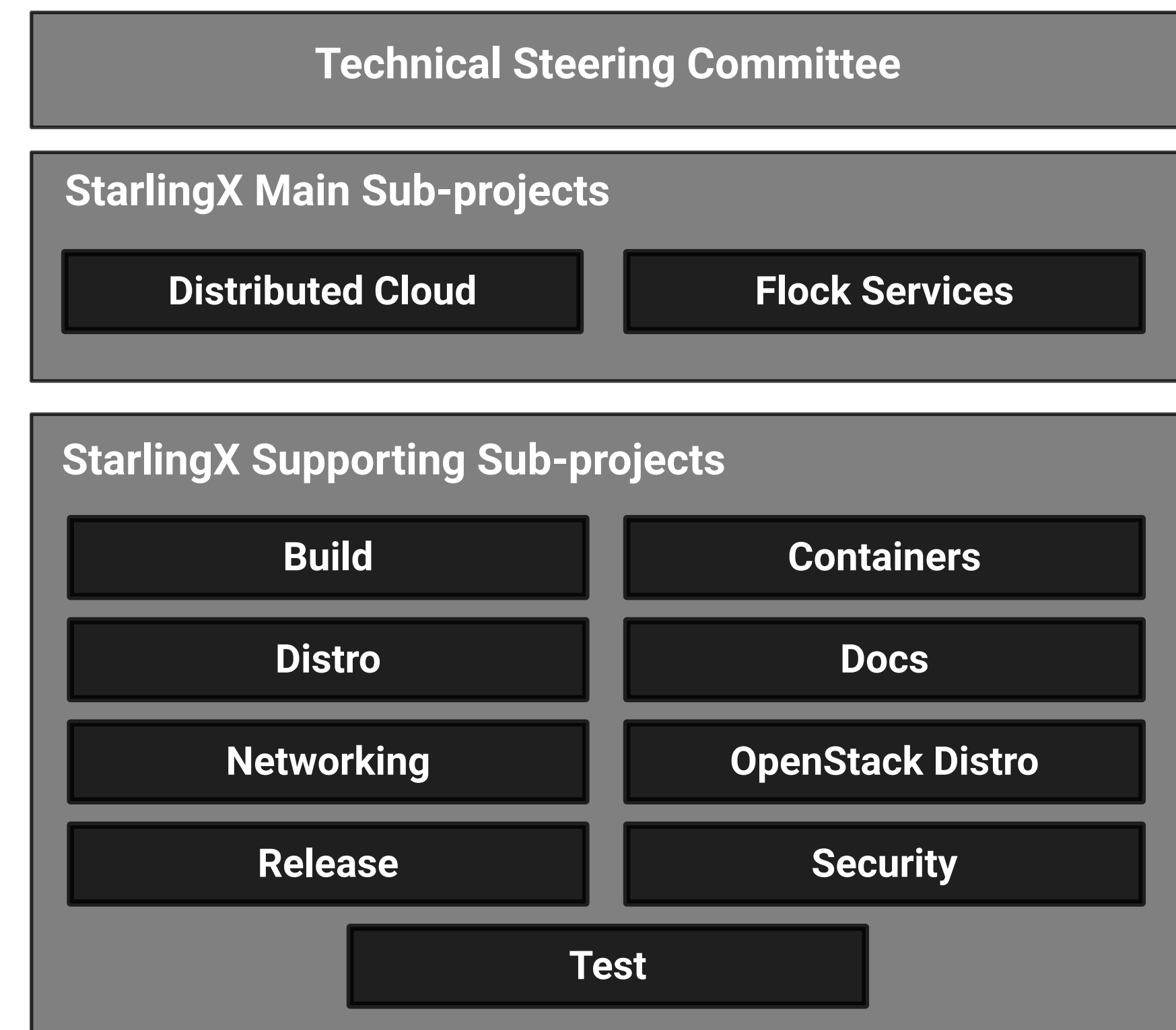
# Community and Contributing

# Principles

- The StarlingX project follows the “Four Opens,”
  - Open Community
  - Open Design
  - Open Development
  - Open Source
- Technical decisions will be made by technical contributors and a representative Technical Steering Committee.
- The community is committed to diversity, openness, encouraging new contributors and leaders to rise up.

# Sub-project Structure

- Main sub-projects
  - New functionality and services
- Supporting sub-projects
  - Supporting services, test and infrastructure
- Sub-project team structure
  - 1 Technical Lead
  - 1 Project Lead
  - Core Reviewers
  - Contributors



# Get Involved

- Code and documentation are available through git
  - <https://opendev.org/starlingx>
- Apache 2 license
- IRC: #starlingx@OFTC
- Mailing List for daily discussions
  - <http://lists.starlingx.io/cgi-bin/mailman/listinfo/starlingx-discuss>

# Where to Contribute?

- Bugs are tracked in Launchpad
  - <https://bugs.launchpad.net/starlingx>
- New ideas are introduced in the specs repository
  - <https://opendev.org/starlingx/specs>
- Design and implementation work is tracked in StoryBoard
  - [https://storybook.openstack.org/#!/project\\_group/86](https://storybook.openstack.org/#!/project_group/86)
- Further information about sub-teams and processes
  - <https://wiki.openstack.org/wiki/StarlingX>

# Communication

- #starlingx@OFTC, IRC channel for online discussions
- Mailing Lists: [lists.starlingx.io](https://lists.starlingx.io)
- Email: [info@starlingx.io](mailto:info@starlingx.io)
- Weekly meetings:
  - Zoom calls
  - <https://wiki.openstack.org/wiki/Starlingx/Meetings>
- Twitter handle: @StarlingX

**03**

## **Meet the Community**

Upcoming OpenInfra Events



# OpenInfra Live on Thursday

- An interactive, live show
- Featuring panel discussions with industry experts, OpenInfra Community updates and more!
- Submit episode ideas at [ideas.openinfra.live](https://ideas.openinfra.live)!
- Popular topics include:
  - Global connectivity
  - Cloud economics
  - Sustainable computing
  - Automation
  - Large scale deployments



Find out more at [openinfra.live](https://openinfra.live)

# OpenInfra Project Teams Gathering

→ October 17-20, Columbus, OH

- Project Teams Gathering (PTG) is back in person!
- Contributor-focused event to plan roadmap and discuss release priorities and further technical topics
- Registration is already open!
- Schedule will be announced soon



Event information at <https://openinfra.dev/ptg/>

A large flock of birds is flying in a circular pattern over a landscape. The scene is set at sunset, with a warm orange and yellow sky. The birds are silhouetted against the bright sky, creating a dense, swirling pattern. The landscape below consists of rolling hills and a body of water in the distance.

**Thank you!**

[openinfra.dev](https://openinfra.dev)