StarlingX Project Overview

Learn, Try, Get Involved!

Let Me Introduce StarlingX

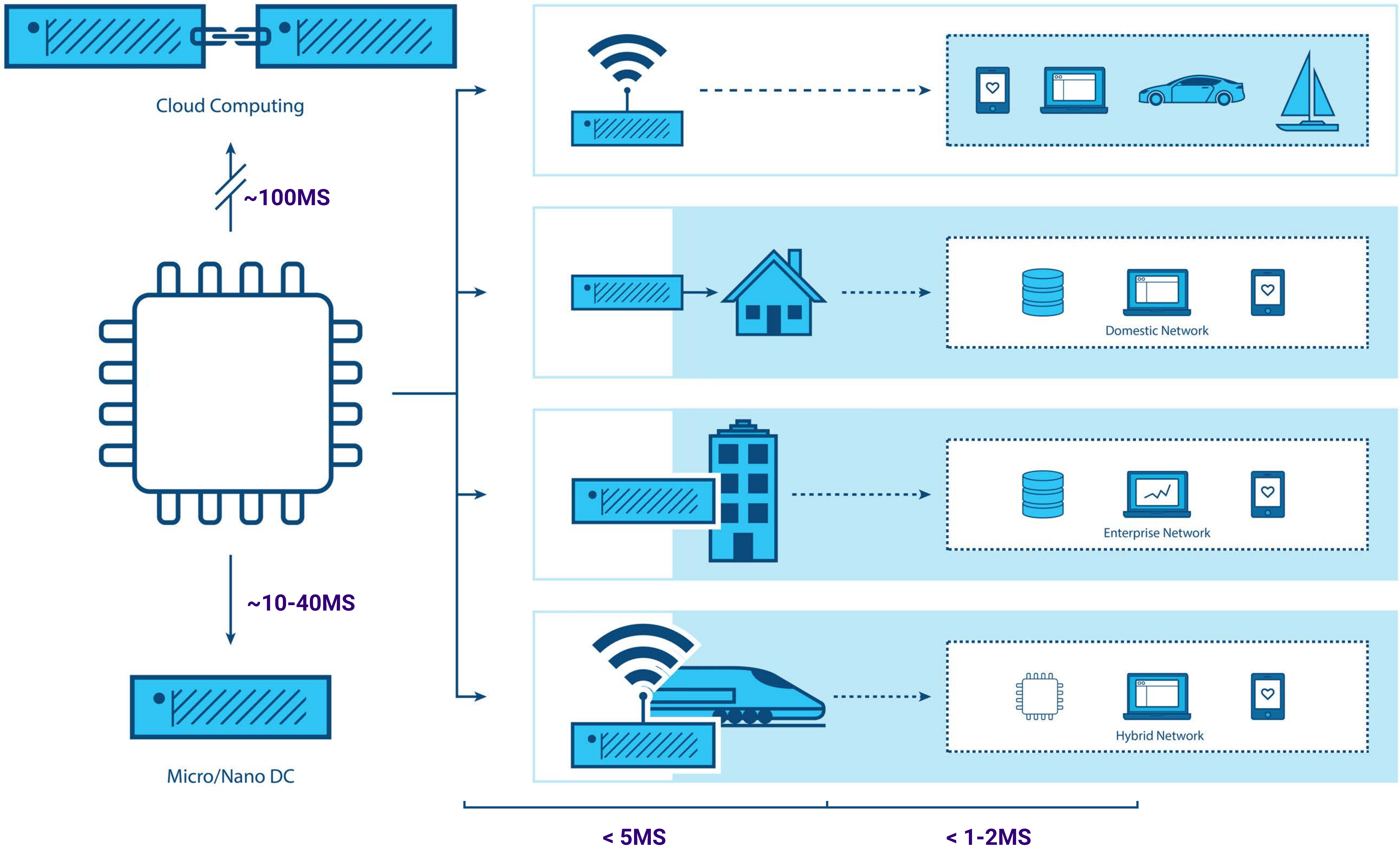
- New, top-level OpenStack Foundation pilot project
- Software stack providing high performance, low latency, and high availability for Edge Cloud applications
- The first release is coming out this week!
- Growing community
  - Inviting users, operators and developers to try out the software and participate in the community

Project Overview

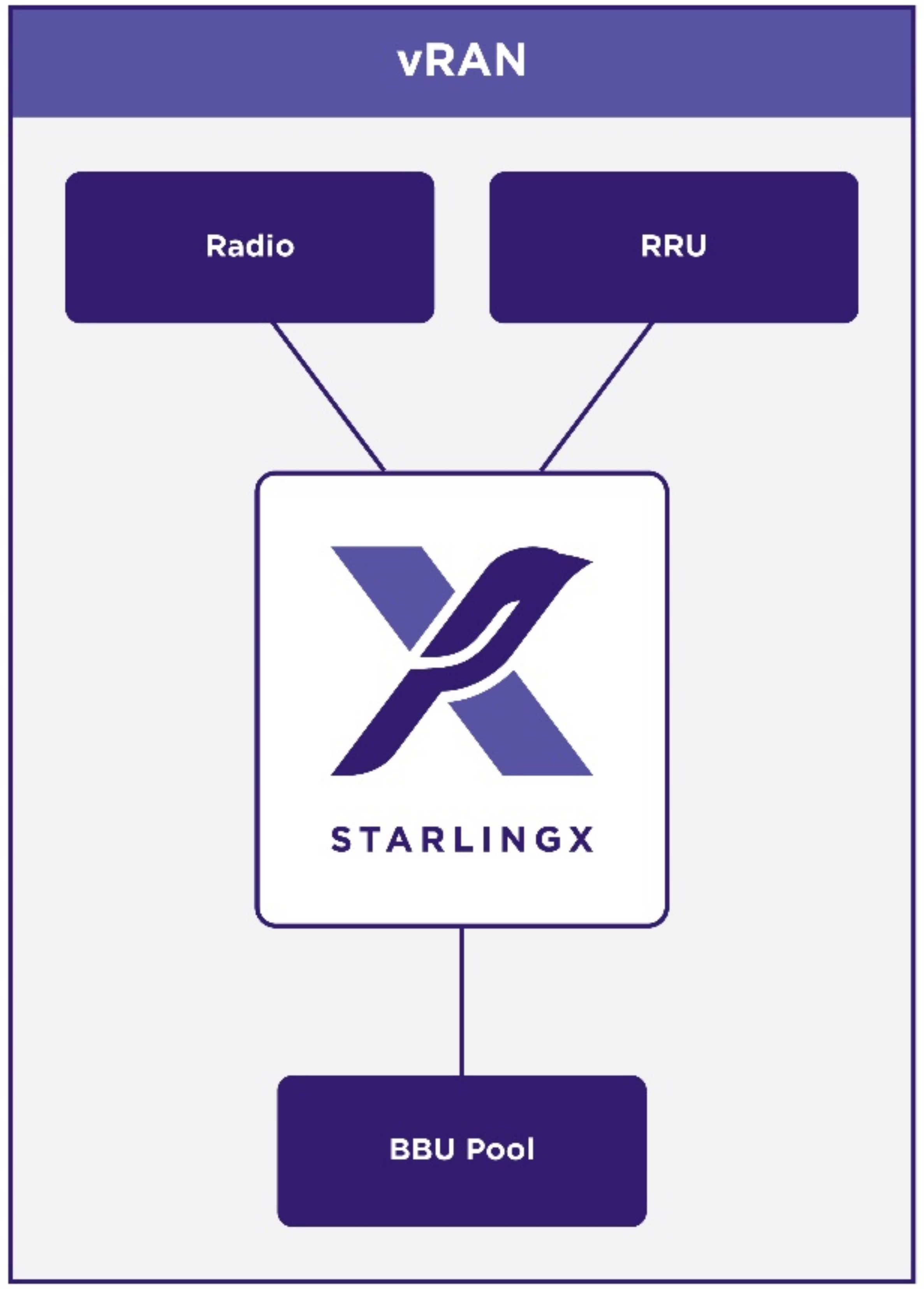
What Is Driving Edge Computing?

A. **Latency**
B. Bandwidth
C. Security
D. Connectivity

"WHERE" MATTERS



Edge Computing Use Cases



vRAN

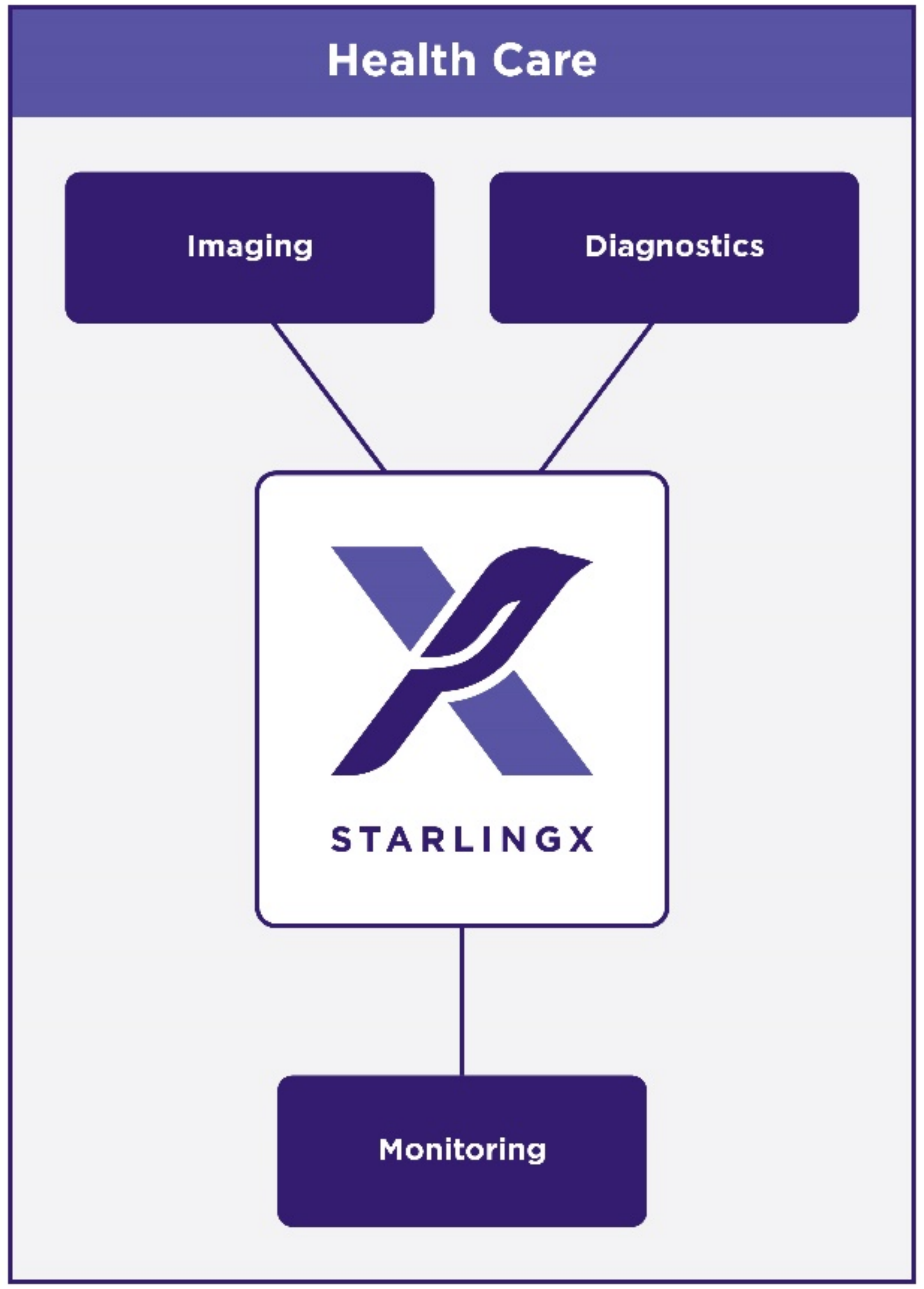
Radio

RRU

STARLINGX

BBU Pool

vRAN == virtual Radio Access Network



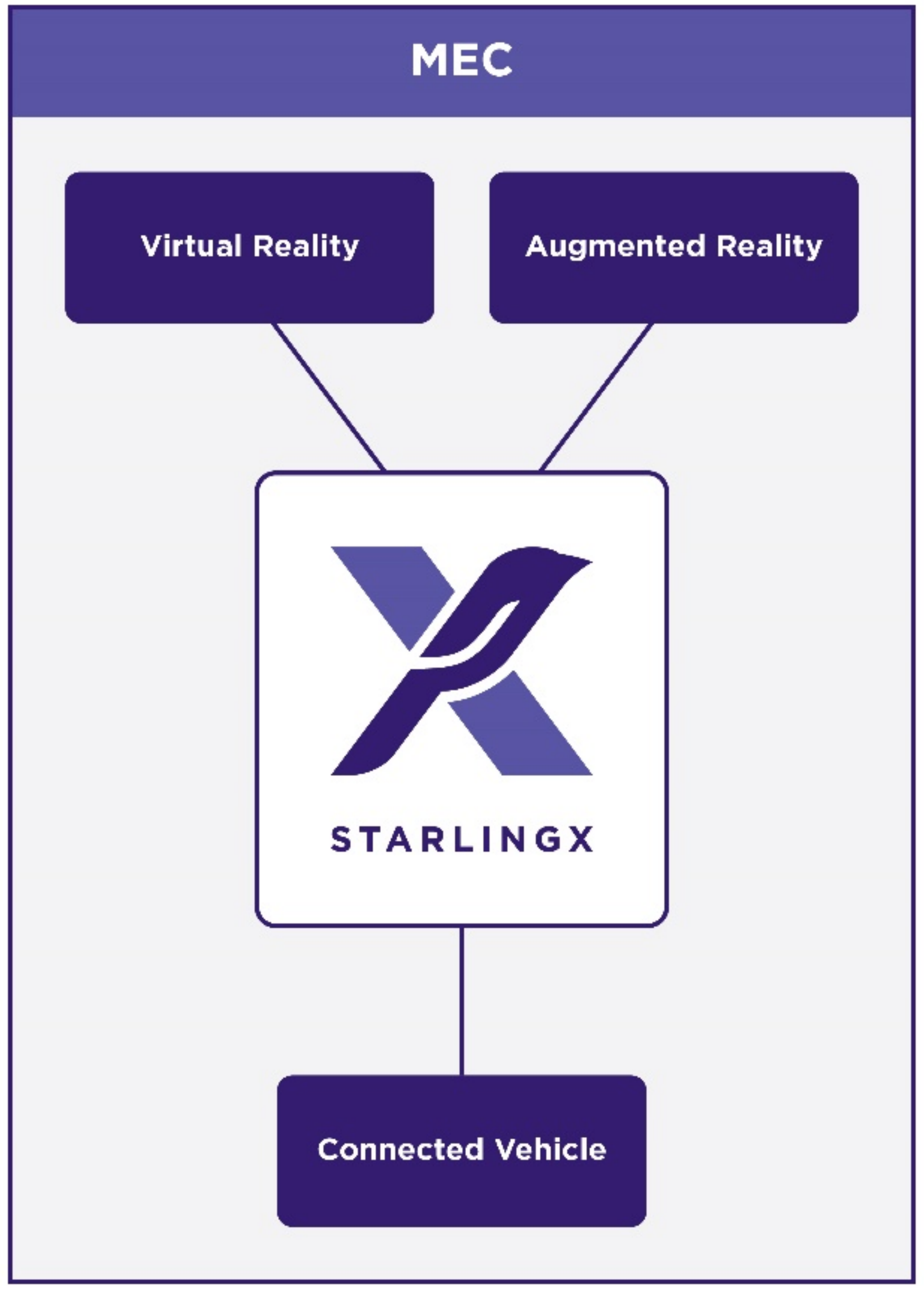
Health Care

Imaging

Diagnostics

STARLINGX

Monitoring



MEC

Virtual Reality

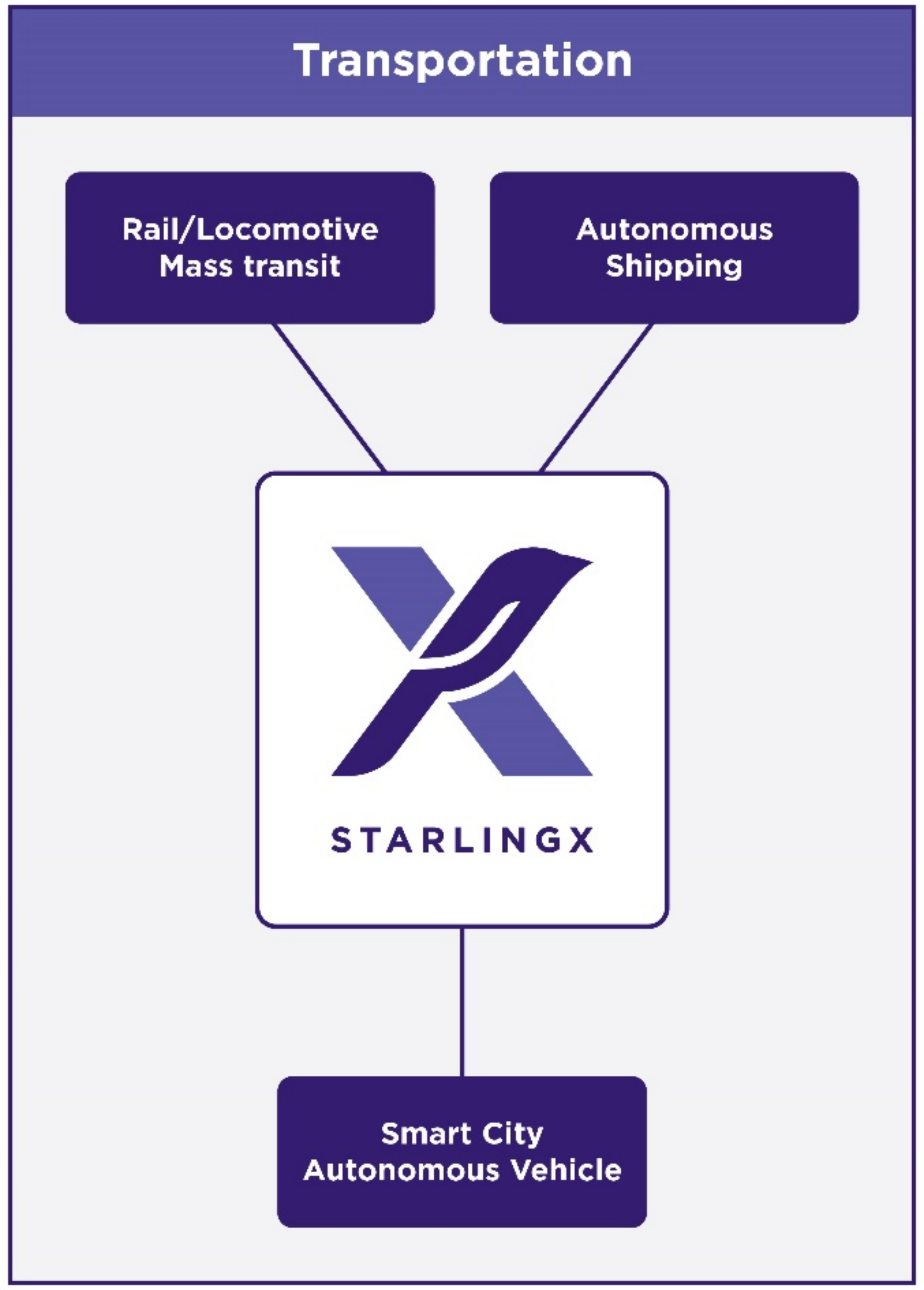
Augmented Reality

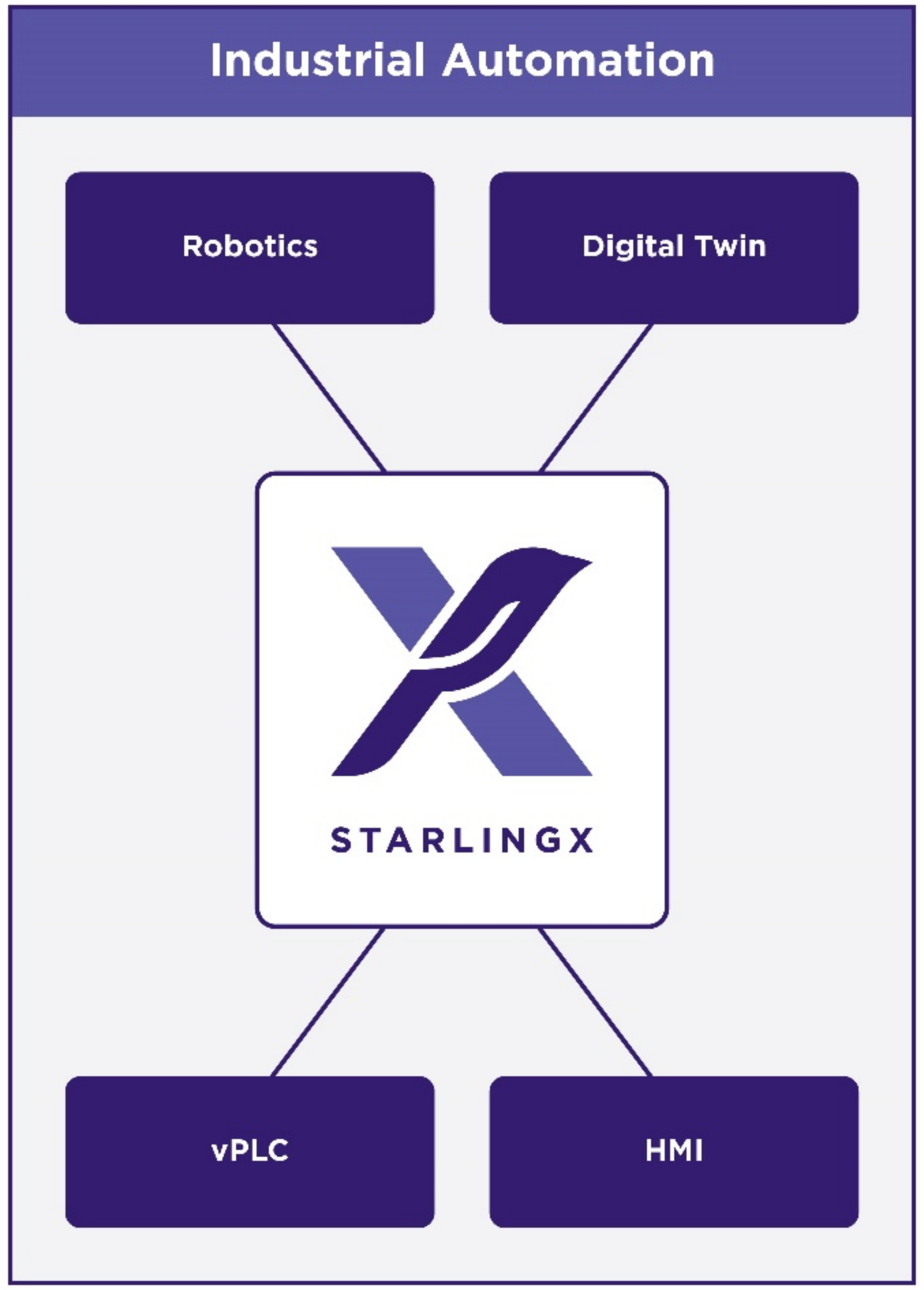
STARLINGX

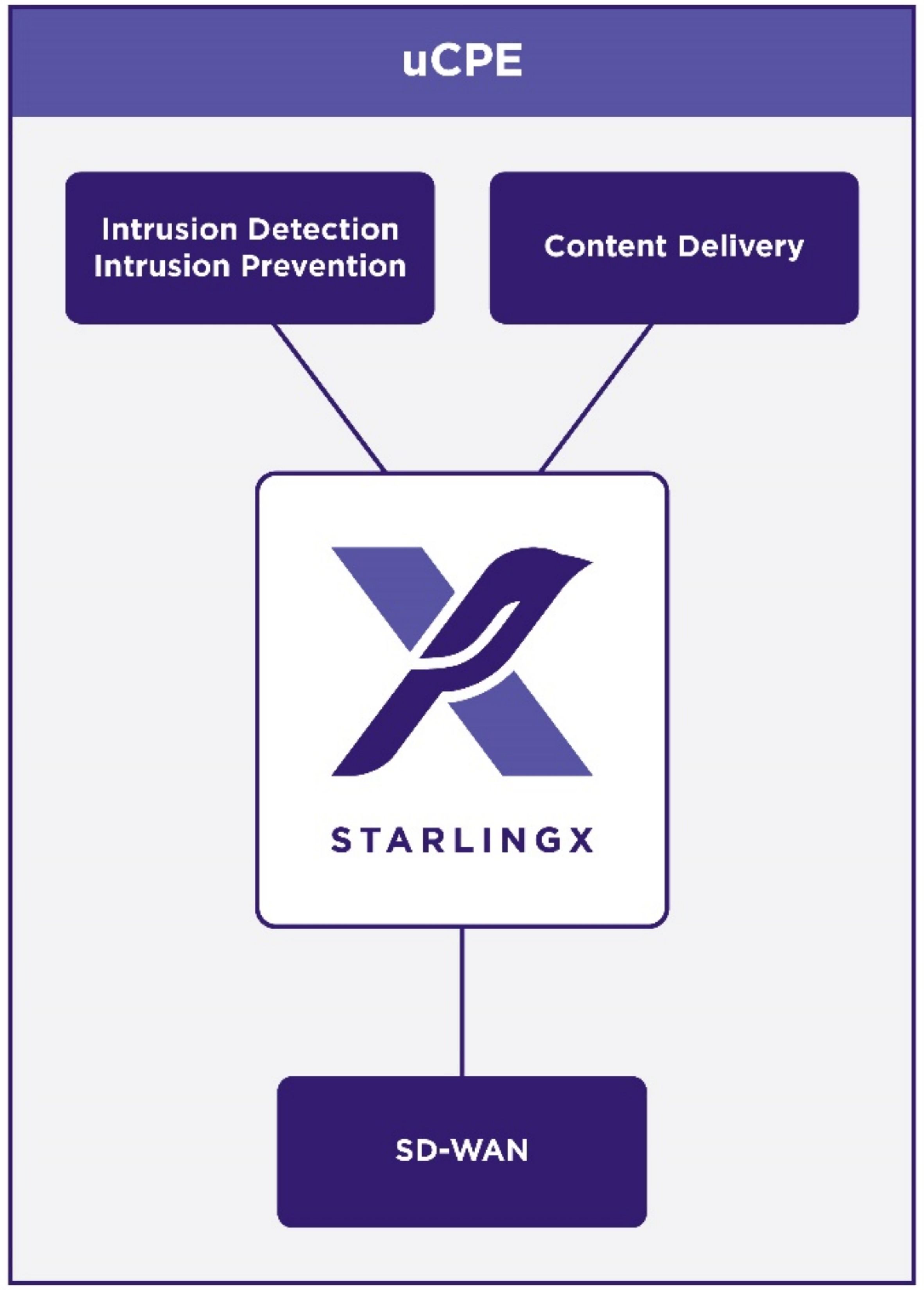
Connected Vehicle

MEC == Multi-access Edge Computing

Edge Computing Use Cases





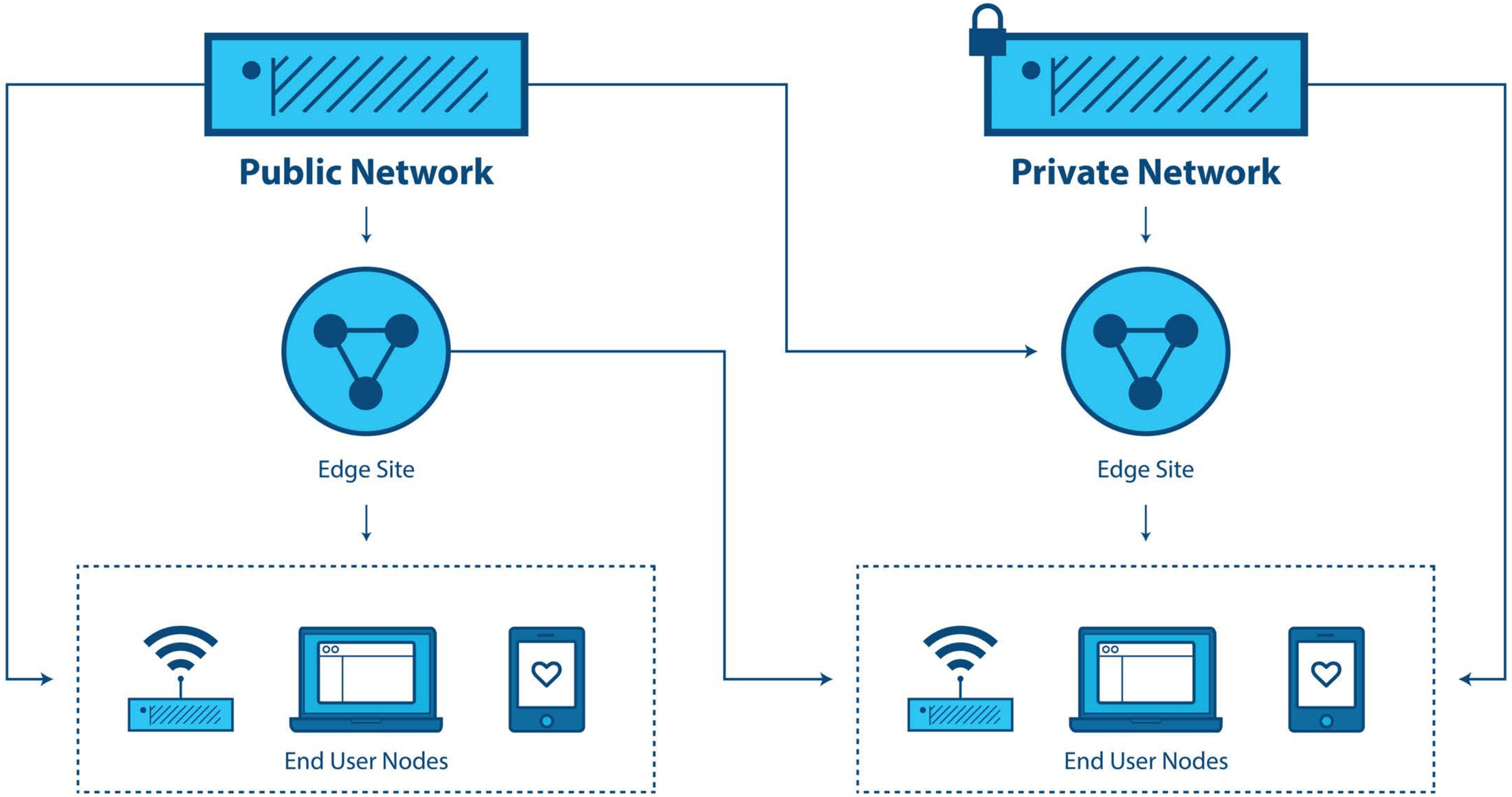


uCPE == universal Customer Premises Equipment

What Problems Is StarlingX Solving?

Massive data growth

Network needs to be smarter

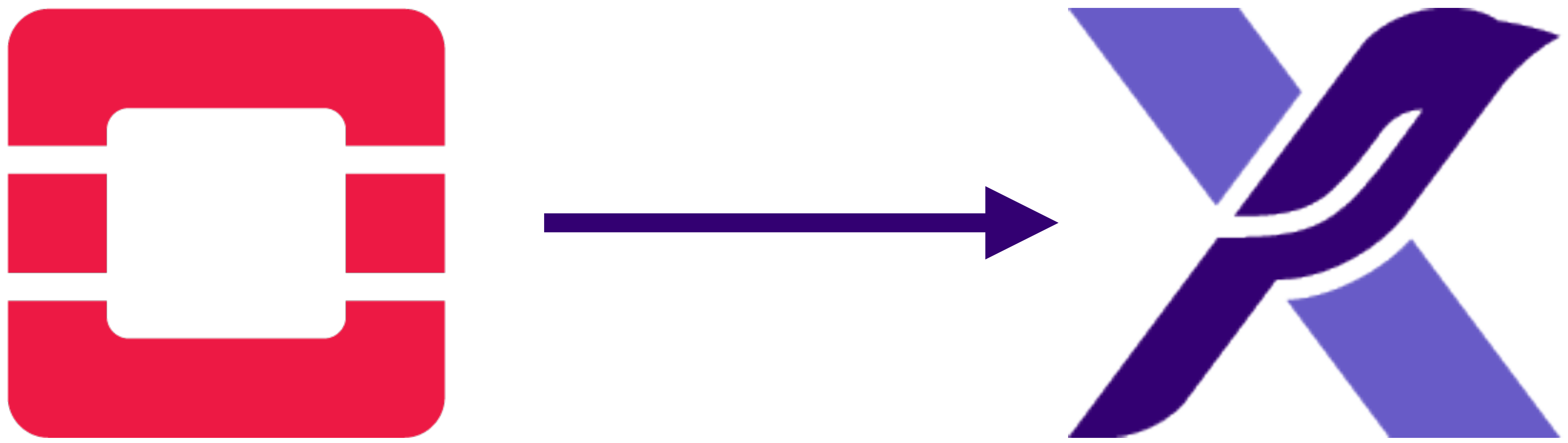


1. Distributed infrastructure demands a different architecture
2. The maturity and robustness of Cloud is required everywhere
3. Managing a massively distributed compute environment is hard

Intent of the StarlingX Project

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



Transportation	Energy
Manufacturing	Retail
Video	Smart cities
Healthcare	Drones

*Other names and brands may be claimed as the property of others

StarlingX Technology

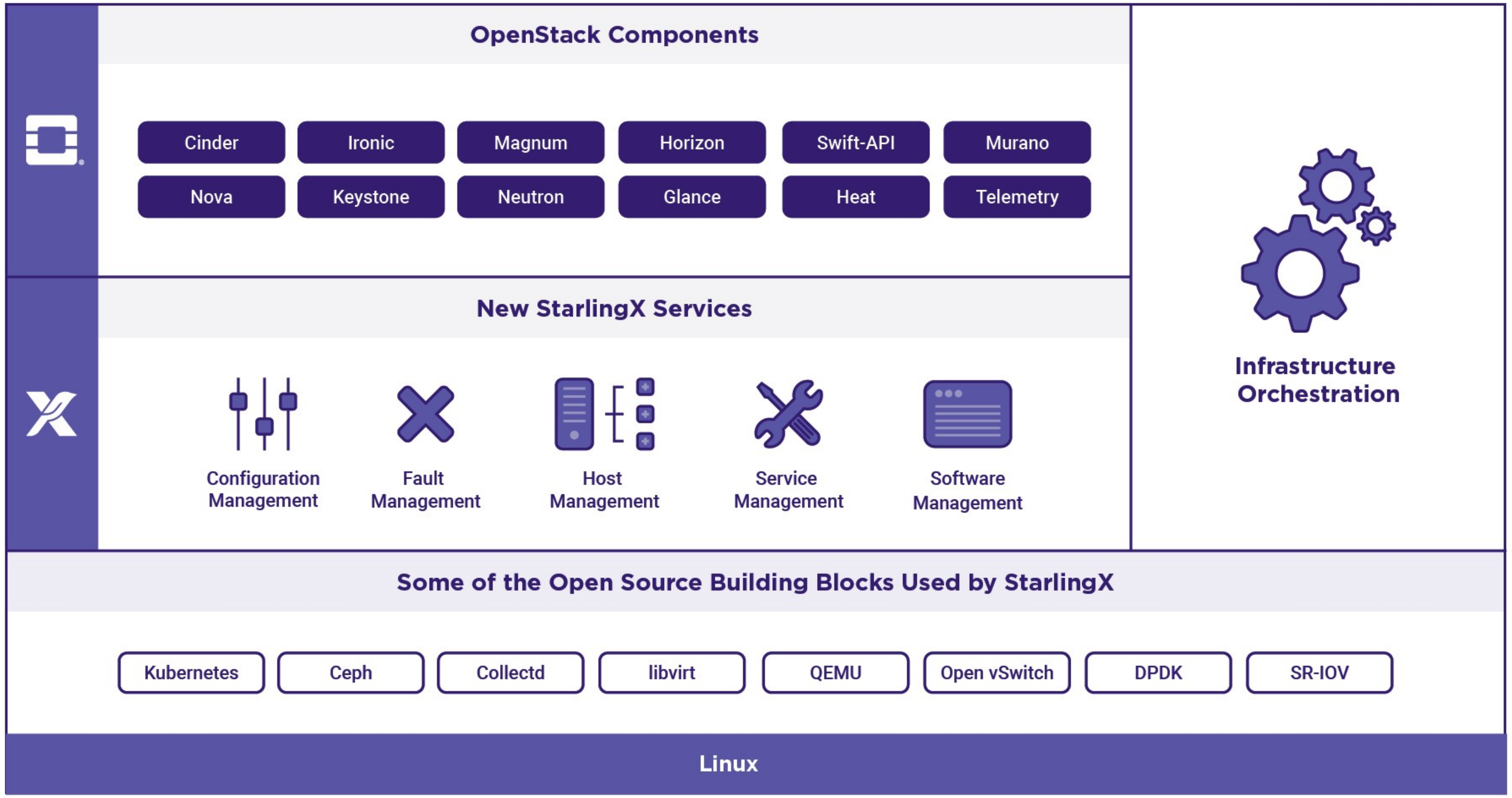
StarlingX – Edge Virtualization Platform

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**

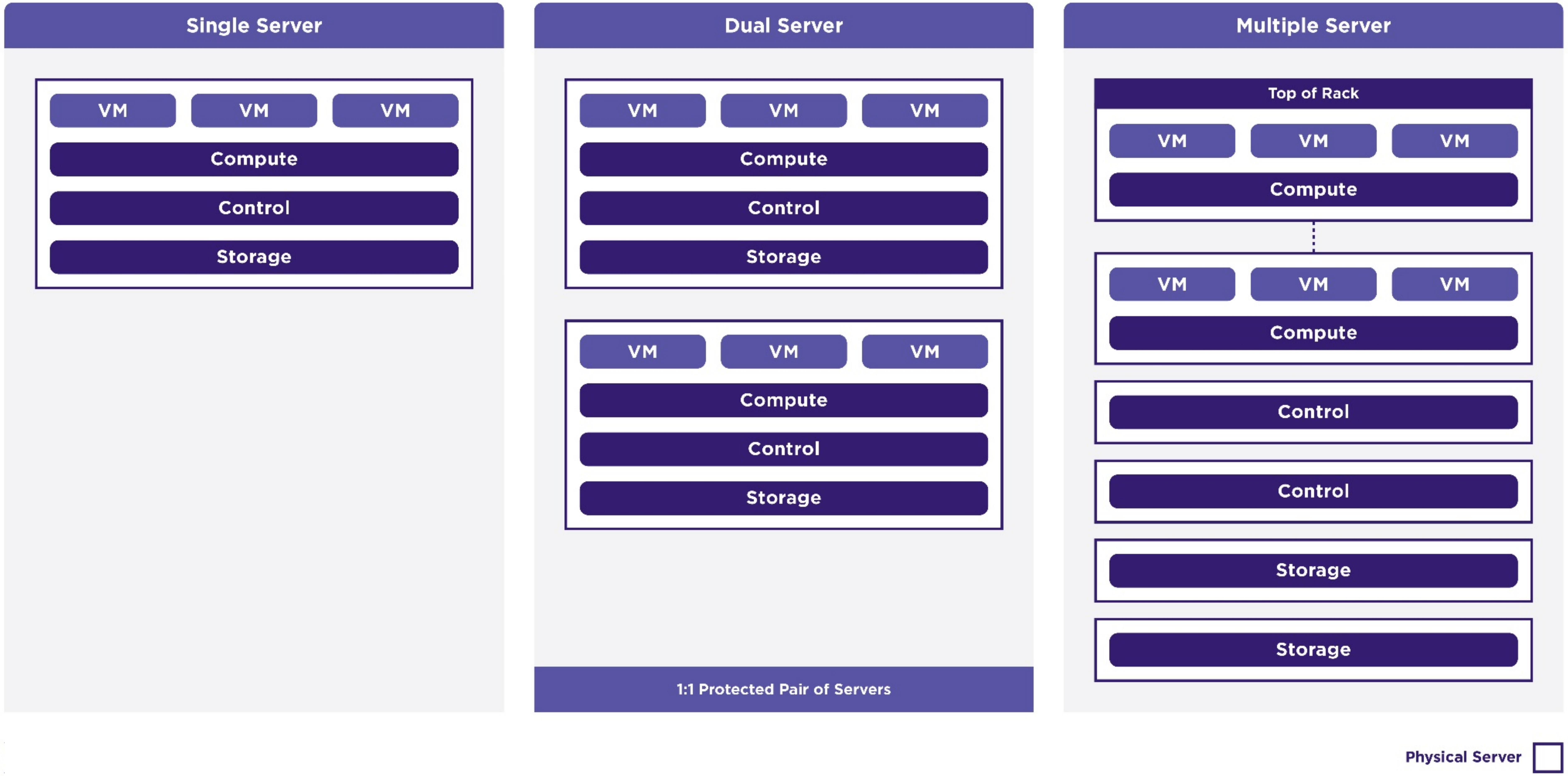
Think control at the Edge, control between IoT and Cloud, control over your virtual machines.



*Other names and brands may be claimed as the property of others

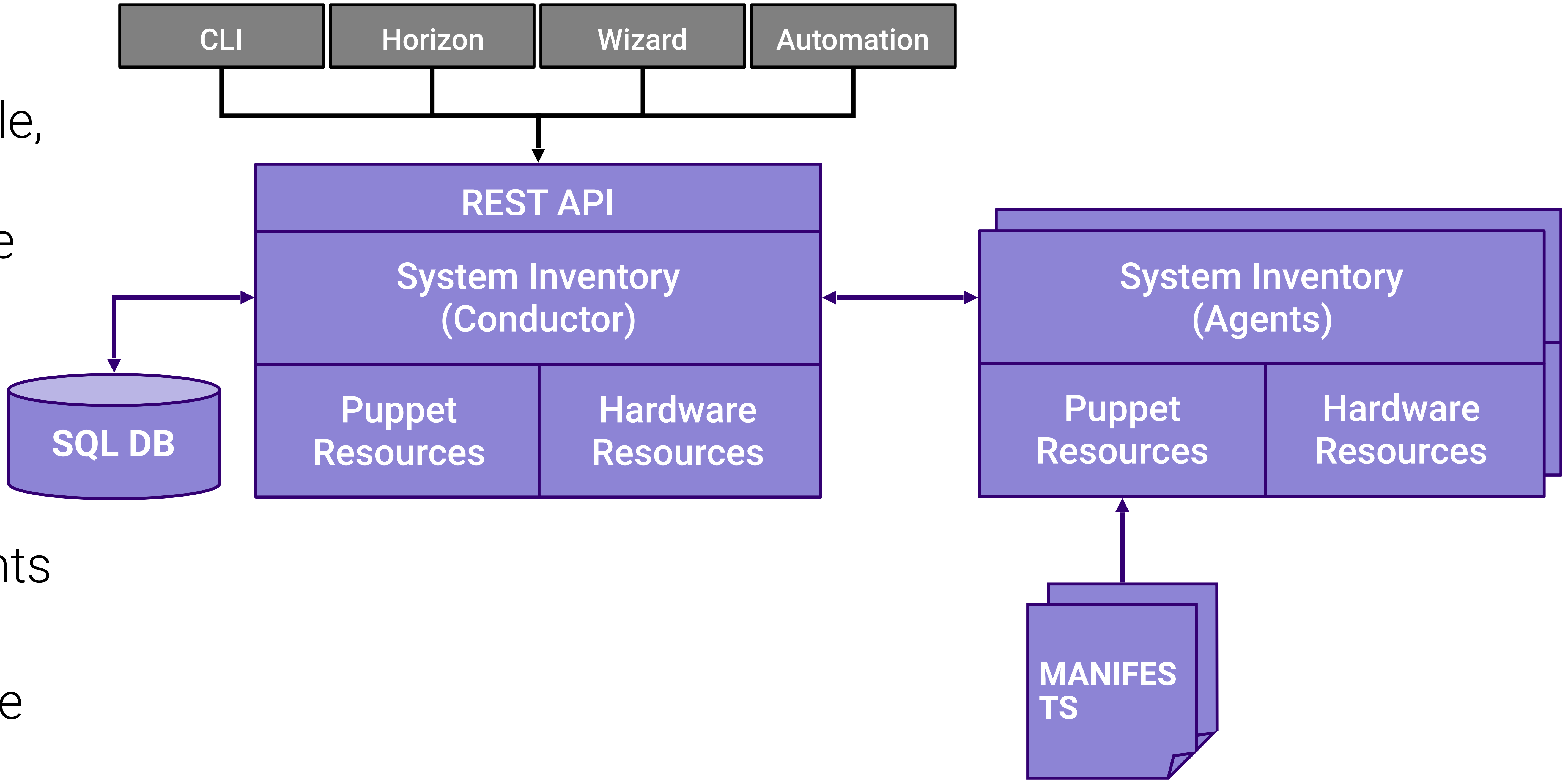
Scalability from Small to Large

- Single Server
  - Runs all functions
- Dual Server
  - Redundant design
- Multiple Server
  - Fully resilient and geographically distributable



Configuration Management

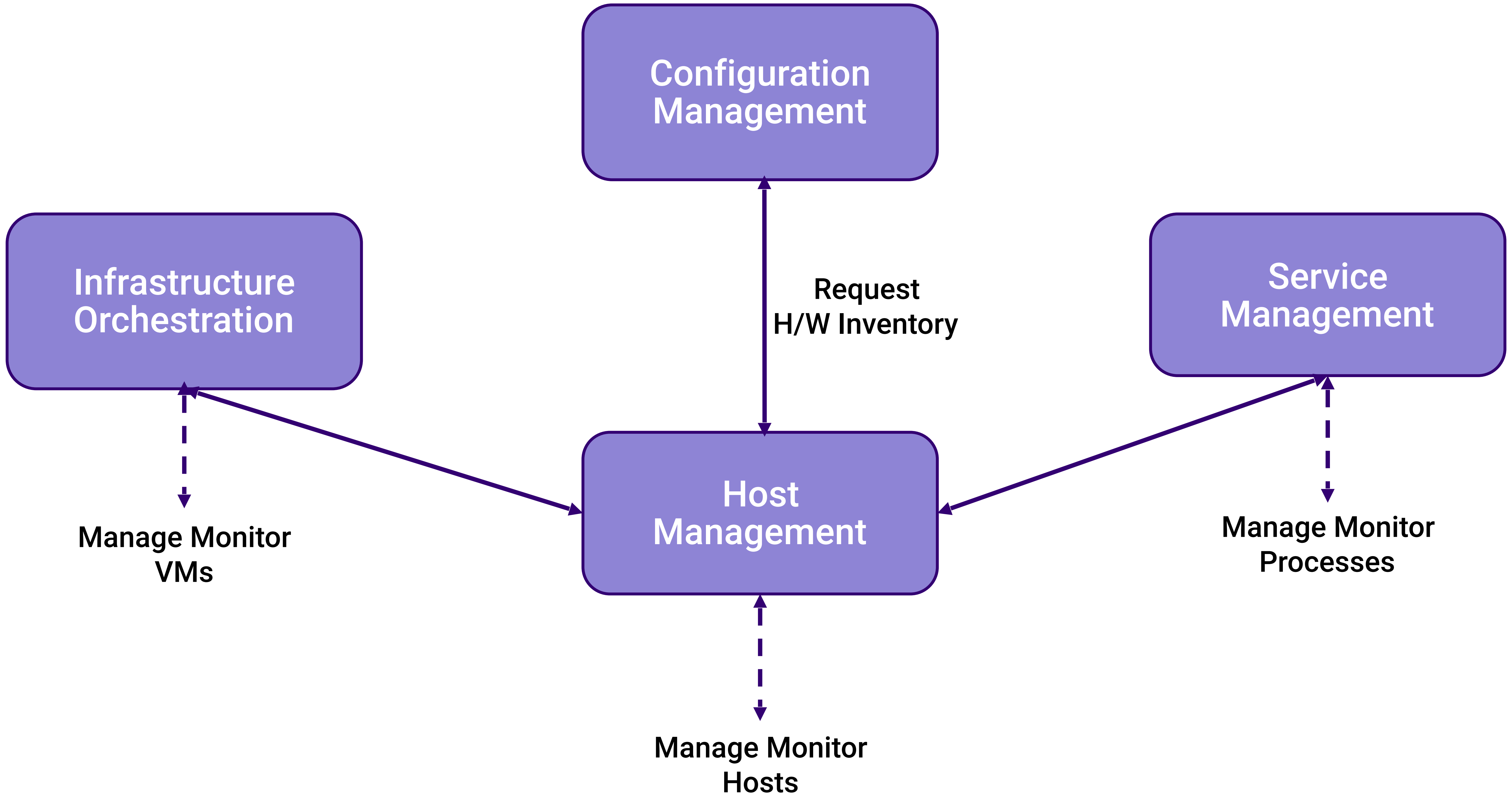
- Manages installation
  - Auto-discover new nodes
  - Manage installation parameters (i.e. console, root disks)
  - Bulk provisioning of nodes through XML file
- Nodal Configuration
  - Node role, role profiles
  - Core, memory (including huge page) assignments
  - Network Interfaces and storage assignments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



System Configuration and Setup

Host Management

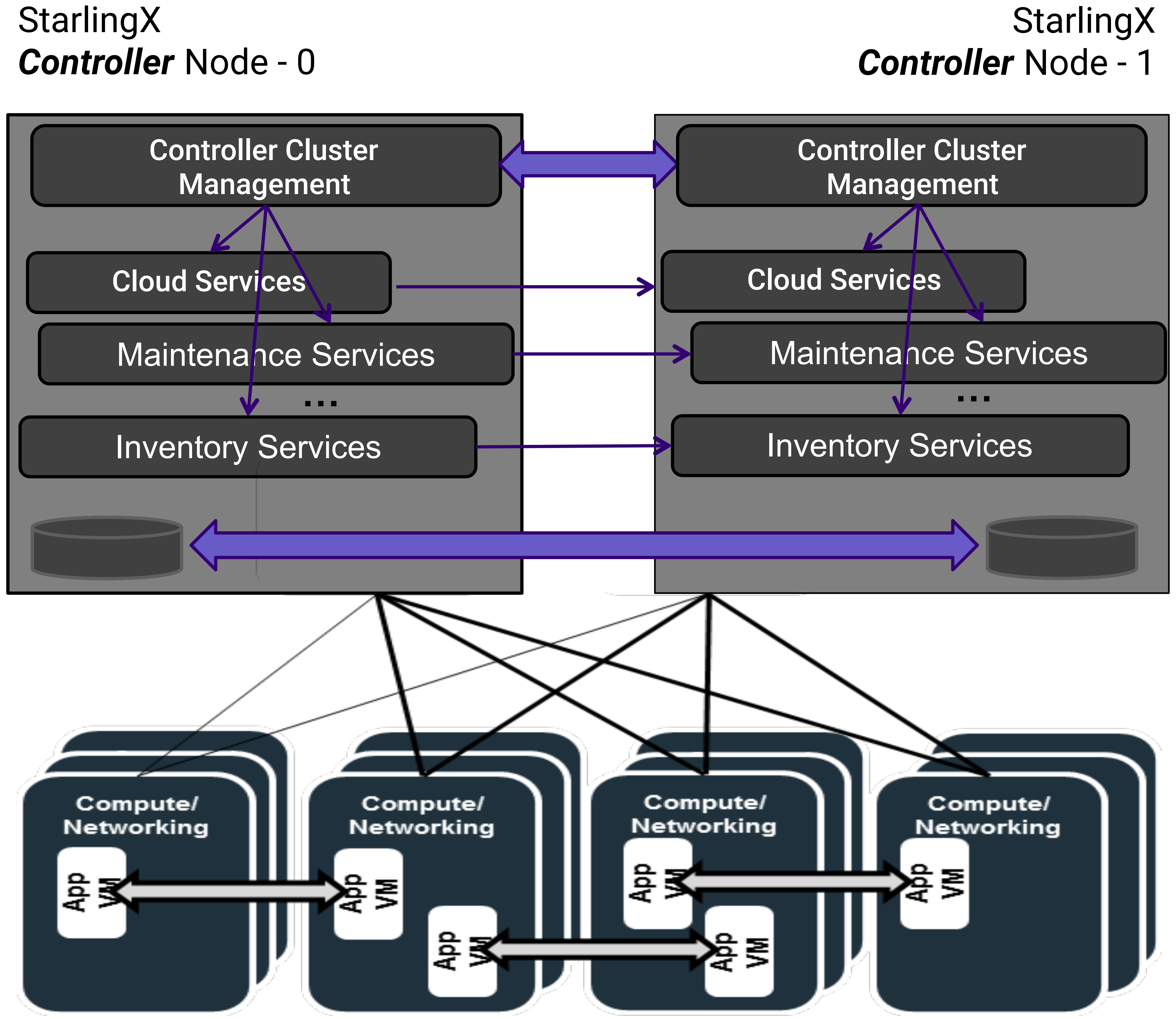
- Full life-cycle management of the host
- Detects and automatically handles host failures and initiates recovery
- Monitoring and alarms for
  - Cluster connectivity, critical process failures
  - Resource utilization thresholds, interface states
  - H/W fault / sensors, host watchdog
  - Activity progress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



Vendor Neutral Host Management

Service Management

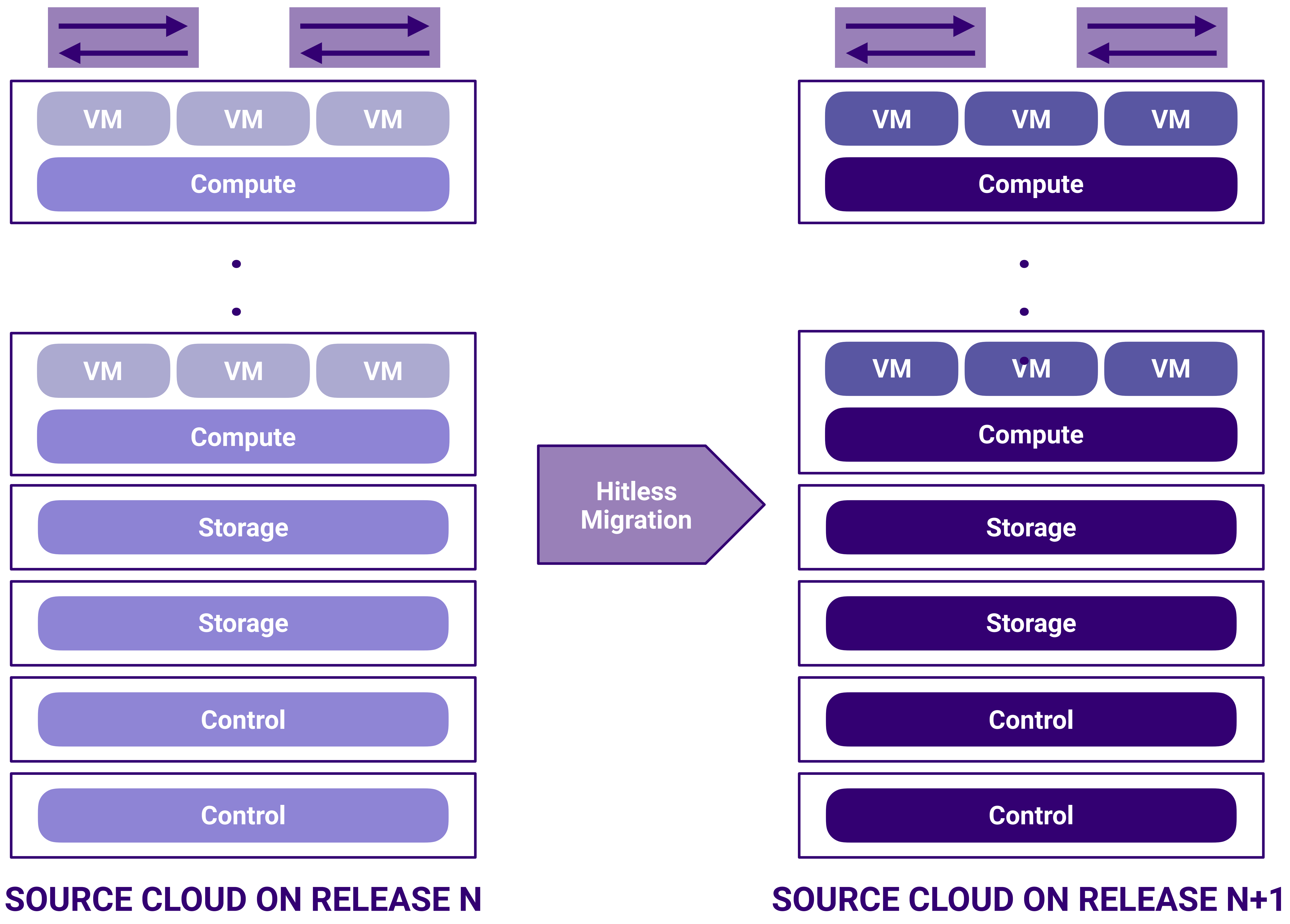
- High availability manager
  - Redundancy model can be N+M or N across multiple nodes
  - Currently 1+1 HA Controller Cluster
- Uses multiple messaging paths to avoid split-brain communication failures
  - Up to 3 independent communication paths
  - LAG can also be configured for multi-link protection 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

Software Management

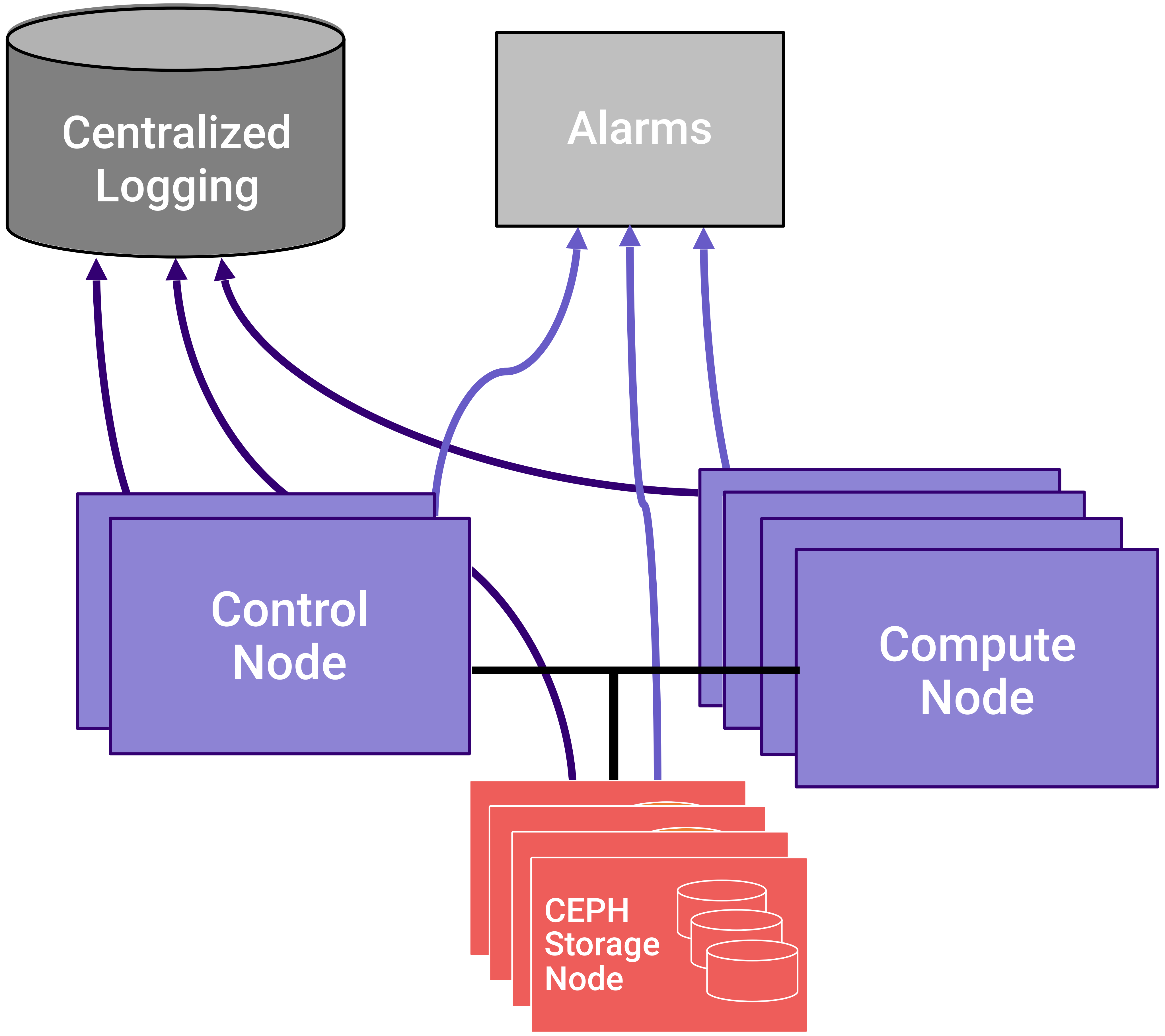
- Automated deploy of software updates for security and/or new functionality
- Integrated end-to-end rolling upgrade solution
  - Automated, low number of steps
  - No additional hardware required for upgrade
  - Rolling upgrade across nodes
- In-service and reboot required patches supported
  - Reboot 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

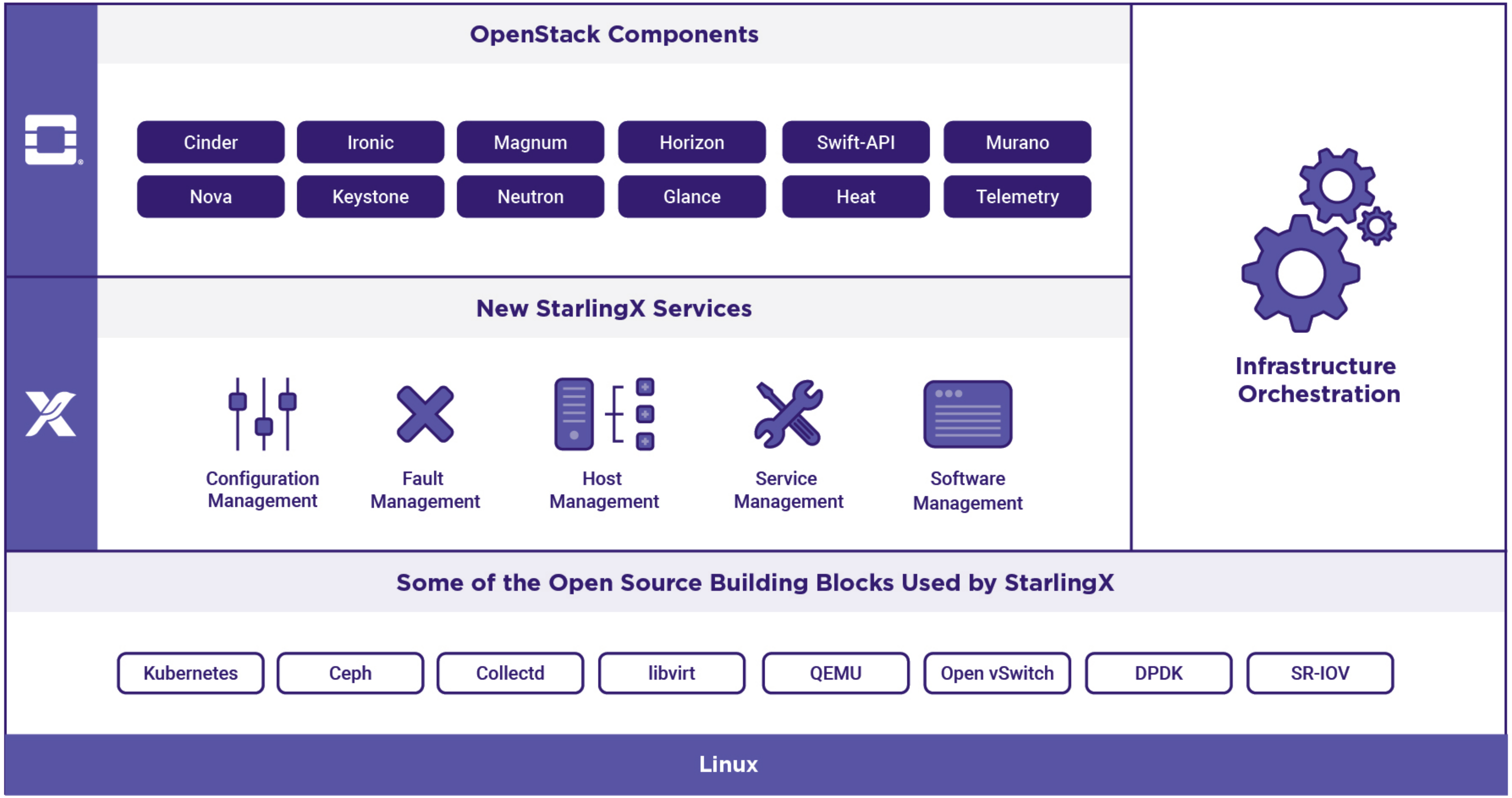
Fault Management

- 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/Clears of alarms
  - Related to platform nodes and resources
  - Related to hosted virtual resources



Fault Alarming and Logging

Current Architecture



OpenStack Components

Cinder, Ironic, Magnum, Horizon, Swift-API, Murano

Nova, Keystone, Neutron, Glance, Heat, Telemetry

New StarlingX Services

Configuration Management, Fault Management, Host Management, Service Management, Software Management

Infrastructure Orchestration

Some of the Open Source Building Blocks Used by StarlingX

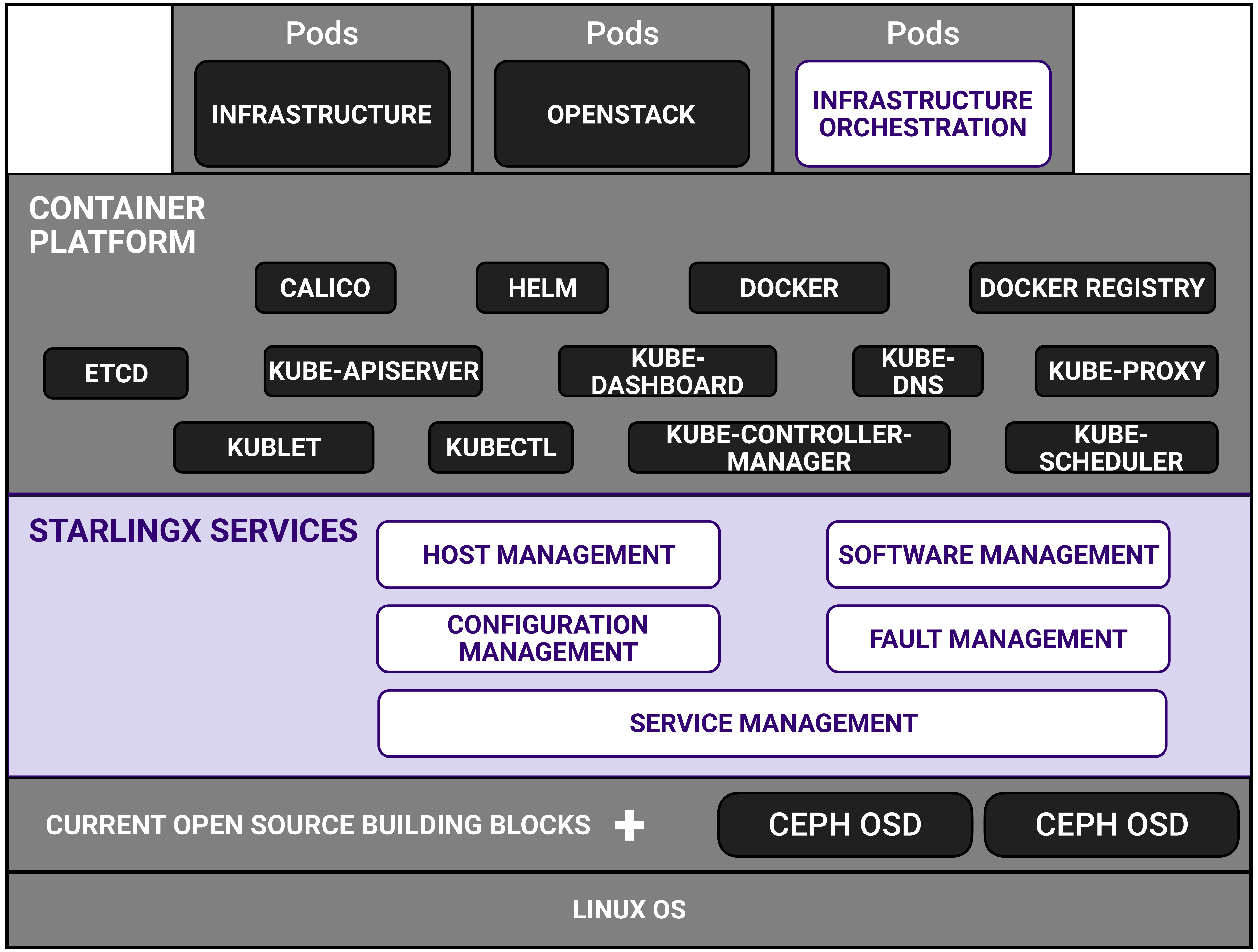
Kubernetes, Ceph, Collectd, libvirt, QEMU, Open vSwitch, DPDK, SR-IOV

Linux

*Other names and brands may be claimed as the property of others

Next Generation Container Architecture

- StarlingX is evolving to
  - Running OpenStack containerized
  - On top of a bare metal Kubernetes cluster
  - With OpenStack Helm managing the lifecycle of the OpenStack cluster
- With Kubernetes Cluster initially supporting
  - Docker runtime
  - Calico CNI plugin
  - CEPH as persistent storage backend
  - HELM as the package manager
  - Local Docker Image Registry
- Along with Kubernetes cluster available for non-OpenStack end user applications



Pods

INFRASTRUCTURE

Pods

OPENSTACK

Pods

INFRASTRUCTURE ORCHESTRATION

CONTAINER PLATFORM

CALICO

HELM

DOCKER

DOCKER REGISTRY

ETCD

KUBE-APISERVER

KUBE-DASHBOARD

KUBE-DNS

KUBE-PROXY

KUBLET

KUBECTL

KUBE-CONTROLLER-MANAGER

KUBE-SCHEDULER

STARLINGX SERVICES

HOST MANAGEMENT

SOFTWARE MANAGEMENT

CONFIGURATION MANAGEMENT

FAULT MANAGEMENT

SERVICE MANAGEMENT

CURRENT OPEN SOURCE BUILDING BLOCKS +

CEPH OSD

CEPH OSD

LINUX OS

Full Support for VMs and Containers

*Other names and brands may be claimed as the property of others

The Road to the Edge

- Build it yourself from open source components
  - Building blocks need refinement
  - Time consuming
  - Gaps to fill
- Use StarlingX
  - New services provide **improved manageability** for the platform and **high availability for your applications** to meet **Edge Cloud requirements**
  - Tested and available as a **complete stack**
  - **Mission-ready** for your applications

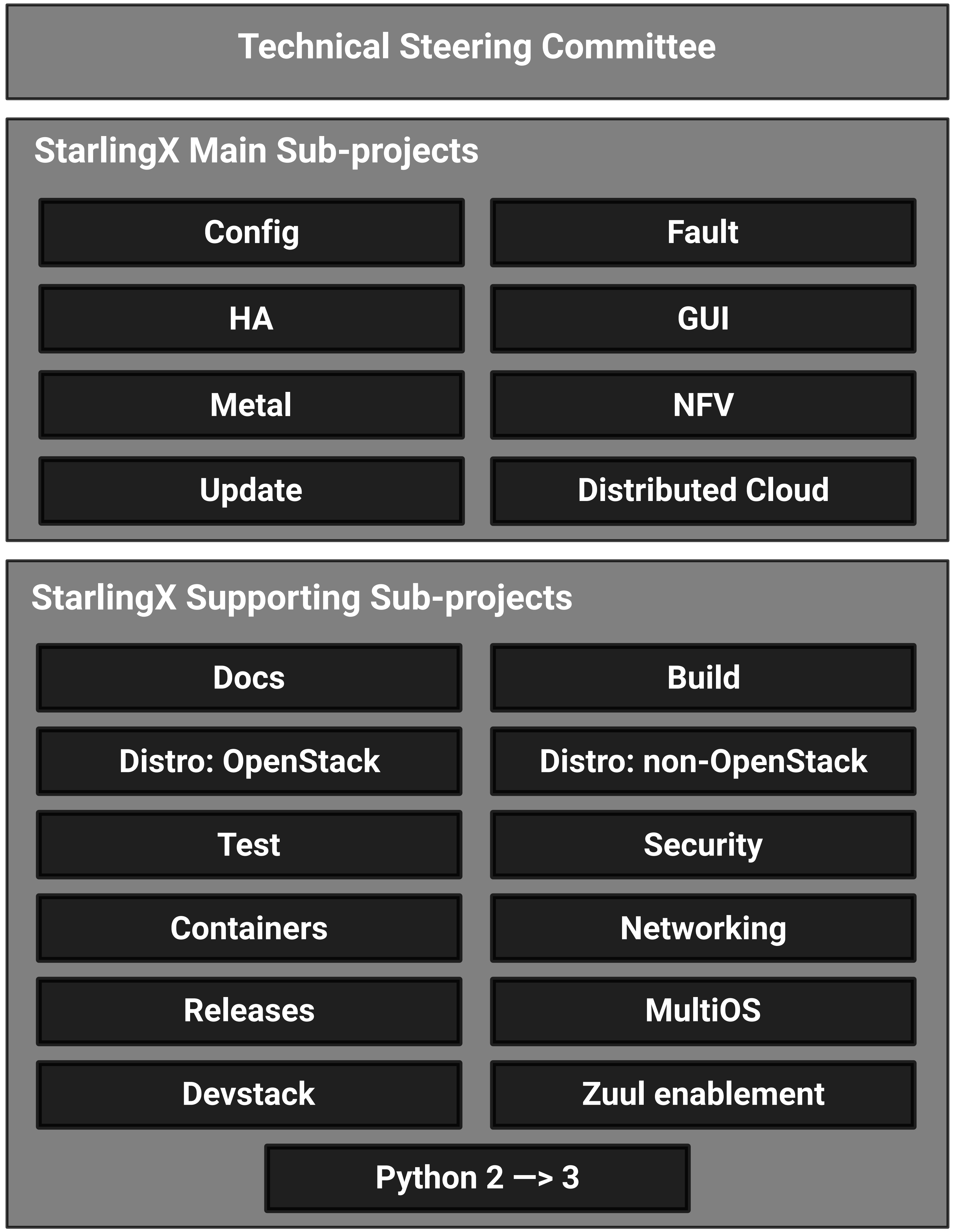
Community and Contributing

Principles

- The StarlingX project follows the “four opens,”
  - Open Collaboration
  - 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 Team Lead
  - 1 Project Lead
  - Core Reviewers
  - Contributors



Technical Steering Committee

StarlingX Main Sub-projects

Config, Fault, HA, GUI, Metal, NFV, Update, Distributed Cloud

StarlingX Supporting Sub-projects

Docs, Build, Distro: OpenStack, Distro: non-OpenStack, Test, Security, Containers, Networking, Releases, MultiOS, Devstack, Zuul enablement, Python 2 —> 3

Governance Roles

- Contributor
  - Someone who made a contribution in the past 12 months
    - Code, test or documentation
    - Serving in a leadership role
  - Can run and vote for elected positions
- Core Reviewer
  - Active contributors to a sub-project, appointed by fellow core reviewers
  - Responsible for reviewing changes and specifications
  - Can merge code and documentation changes

Governance Roles

- Technical Lead
  - Per sub-project
  - Core Reviewer with additional duties
  - Helps guiding the technical direction of a sub-project
- Project Lead
  - Sub-project level coordination work
    - Tracks and communicates progress and priorities
  - Sub-project ambassador

Governance Bodies

- Technical Steering Committee (TSC)
  - Responsible for overall project architectural decisions
  - Managing the sub-project life-cycle
  - Making final decisions if sub-project Core Reviewers, Technical Leads or Project Leads disagree
  - It will be comprised of 7 people, where the initial group will be appointed; the project will move to an election based system within the first year
  - The initial TSC members are Brent Rowsell (Wind River), Ian Jolliffe (Wind River), Dean Troyer (Intel) and Saul Wold (Intel)

Get Involved

- Code and documentation are available through git
  - git.starlingx.io
- Apache 2 license
- IRC: #starlingx@Freenode
- 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://git.openstack.org/cgit/openstack/stx-specs/
- Design and implementation work is tracked in StoryBoard
  - https://storyboard.openstack.org/#!/project_group/86

Community

- You do not need to be an Individual Member of the OpenStack Foundation in order to contribute, but if you want to vote in the annual OpenStack Foundation Board of Directors election, you may join: openstack.org/join
- If you are contributing on behalf of an employer, they will need to sign a corporate contributor license agreement, which now covers all projects hosted by the OpenStack Foundation (same model such as Apache and CNCF)

Communication

- #starlingx@Freenode, IRC channel for online discussions
- Mailing Lists: lists.starlingx.io
- Email: info@starlingx.io
- Weekly meetings:
  - Zoom calls
  - https://wiki.openstack.org/wiki/Starlingx/Meetings
- Twitter handle: @StarlingX

Thank You!