

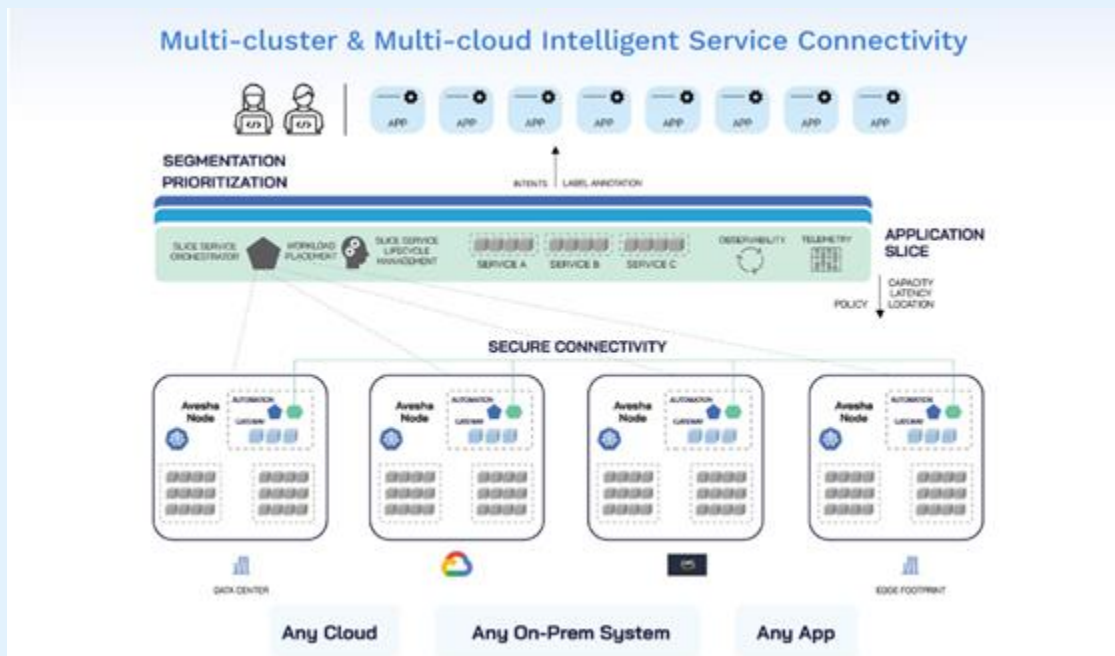
Overview

KubeSlice makes Kubernetes simple at scale for multi cluster/multi-tenant/ multi-region/multi-cloud application deployments. It is a platform that combines network, application, Kubernetes, and deployment services to bring uniformity across clusters for multi-cluster applications, thus dramatically increasing development velocity for platform and product teams.

KubeSlice bundles the following services into its architecture:

- **Deployment services:** location-based workload placements
- **Network services:** tenancy isolation, multi-cluster connectivity, policy propagation
- **Application services:** multi-cluster service discovery
- **Kubernetes services:** namespace management, RBAC, resource management

Control, Visibility & Management for Distributed Multi Cluster Applications



Features

Multi-Cluster Support

- **Application connectivity:** Enables application connectivity across clusters/clouds with zero touch provisioning
- **Overlay Network:** Constructs virtual clusters across physical clusters by establishing an overlay network
- **Traffic Prioritization:** Guarantees the ability to dependably run high-priority applications and traffic with QoS configuration for inter-cluster network connections.

Multi-Tenancy Support

- **App Segmentation:** Define QoS profiles on a per slice basis; thus providing the ability to isolate microservices on one Slice from another.
- **Network Policies:** Auto deploys network policies across clusters participating in the slice configuration, preventing configuration drift.
- **Resource management:** Enables configuration of native Kubernetes resource quotas and limit ranges across clusters on a per Slice basis

Namespace Sameness: Multi-Tenancy Support

- **Multi-cluster namespace:** Ensures namespace sameness on a Slice across multi-cluster/cloud
- **Multi-cluster:** Enables the aggregation of a group of namespaces across clusters thus allowing segmentation for multi-tenancy.
- **Access Controls:** RBAC functionality is propagated across all clusters participating in the Slice configuration.

Service Discovery

- **Auto discovery:** Enables automatic service discovery across clusters participating in the Slice configuration.
- **Service Export:** Inter-cluster service communication is achieved by exporting microservices on the Slice by installing a Service Export object. The Slice DNS propagates service the import objects across all clusters which are part of the Slice.

Advantages

Multi-Cluster Scalability

- **Simplified scaling:** Bring your own topology, clusters in a region, across regions or across clouds to KubeSlice. Create multiple virtual clusters, across physical clusters simplifying application deployment across heterogeneous environments.
- **No network planning:** Adding a cluster to a Slice is like adding additional capacity and requires no network planning to expand across additional clusters.
- **Resiliency:** Deploying workloads on Slice, across clusters, enables scaling for resiliency and disaster recovery design patterns.

Multi-Tenancy & Isolation

- **Inherently multi-tenant:** Tenancy for organizational separation; Isolation based on application priority with cluster redundancy, are the core benefits of KubeSlice.
- **Other tenancy needs:** Tenants of varying shapes and requirements can be achieved by using Taints and Tolerations of nodes across multiple clusters.
- **Targeted placements:** Ability to deploy pods which are dedicated to a cluster but attached to a Slice helps with targeted workload placements based on app needs.

Jurisdiction and Data Sovereignty:

- **Governance & Compliance:** Enables distribution of workloads across clusters where in-country data residency requirements need to be met for compliance reasons.
- **Low Latency:** Enables low latency applications to be at close proximity to the users and ensures high availability for better user experience.

Multi-Cluster Operational Efficiency

- **Hub Cluster:** Employs hub cluster architecture to manage all worker clusters- the Slice is maintained at the hub cluster. Maintaining a hub cluster ensures efficiency of distribution of policies across Slices and worker clusters.
- **Policy Management:** External integrations (OPA Gatekeeper / Kyverno) are integrated to the hub cluster and the policies propagate across clusters.