

# YugabyteDB – Distributed SQL Database on Kubernetes

**Amey Banarse** VP of Product, Yugabyte, Inc.

**Taylor Mull** Senior Data Engineer, Yugabyte, Inc.



# Introduction – Amey

---



## Amey Banarse

VP of Product, Yugabyte, Inc.

Pivotal • FINRA • NYSE

University of Pennsylvania (UPenn)

 @ameybanarse

[about.me/amey](https://about.me/amey)

# Introduction – Taylor

---



## Taylor Mull

Senior Data Engineer, Yugabyte, Inc.

DataStax • Charter

University of Colorado at Boulder

# Kubernetes Is Massively Popular in Fortune 500s

---

- Walmart – Edge Computing

KubeCon 2019 <https://www.youtube.com/watch?v=sfPFrvDvdIk>



- Target – Data @ Edge

<https://tech.target.com/2018/08/08/running-cassandra-in-kubernetes-across-1800-stores.html>



- eBay – Platform Modernization

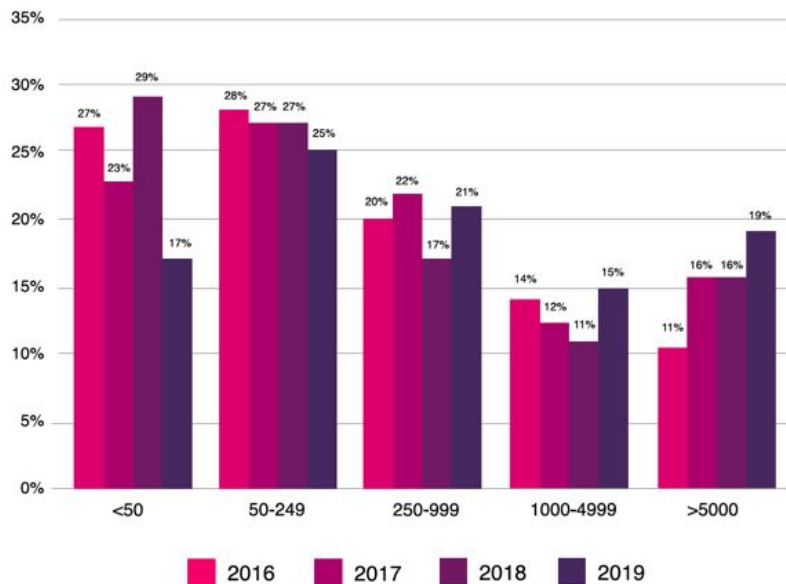
<https://www.ebayinc.com/stories/news/ebay-builds-own-servers-intends-to-open-source/>



# The State of Kubernetes 2020

- Substantial Kubernetes growth in Large Enterprises
- Clear evidence of production use in enterprise environments
- On-premises is still the most common deployment method
- Though there are pain points, most developers and executives alike feel Kubernetes is worth it

Number of Containers in Production



VMware The State of Kubernetes 2020 report

<https://tanu.vmware.com/content/ebooks/the-state-of-kubernetes-2020>

<https://containerjournal.com/topics/container-ecosystems/vmware-releases-state-of-kubernetes-2020-report/>

# Data on K8s Ecosystem Is Evolving Rapidly

---



# Why Data Services on K8s?

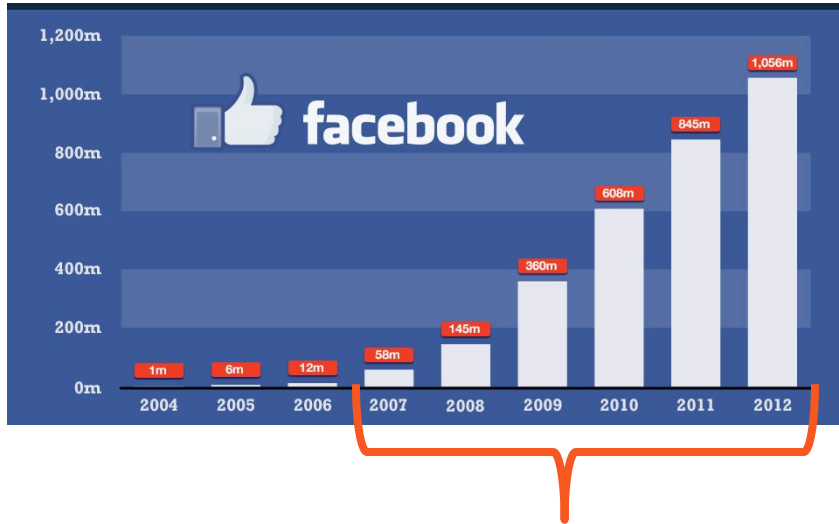
---

Containerized data workloads running on Kubernetes offer several advantages over traditional VM / bare metal based data workloads including but not limited to:

- Better cluster resource utilization
- Portability between cloud and on-premises
- Self Service Experience and seamlessly Scale on demand during peak traffic
- Robust automation framework can be embedded inside **CRDs** (Custom Resource Definitions) or commonly referred as 'K8s Operator'
- Simple and selective instant upgrades

# A Brief History of Yugabyte

Part of Facebook's cloud native DB evolution



- Yugabyte team dealt with this growth first hand
- Massive geo-distributed deployment given global users
- Worked with world-class infra team to solve these issues

Builders of multiple popular databases

ORACLE®



Yugabyte founding team ran Facebook's public cloud scale DBaaS

**+1 Trillion**

ops/day

**+100 Petabytes**

data set sizes





**Transactional, distributed SQL database designed for resilience and scale**

*100% open source, PostgreSQL compatible, enterprise-grade RDBMS  
.....built to run across all your cloud environments*



# Enabling Business Outcomes



- System of Record driving shopping list service
- Designed to be Multi-Cloud on GCP and Azure
- Scaling to 42 states, and 9m shoppers



For Financial Institutions and Fintech Firms

We Make Market Data Easy

With the Most Advanced Cloud-Native Market Data Solutions

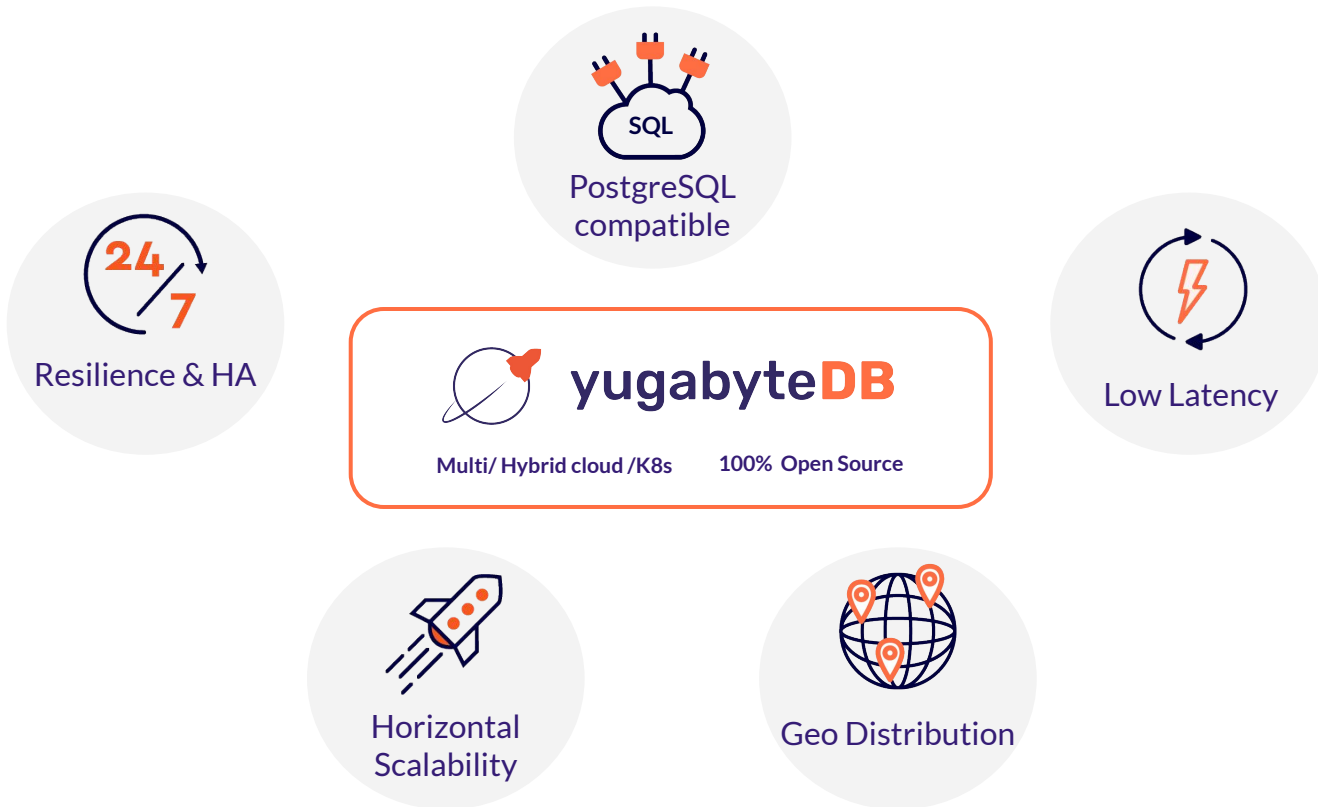
- Real-time APIs for financial services data
- 14 Billion requests per day with data from over 100 world-wide exchanges
- Serves major financial tech and services firms from Betterment to Schwab



- Retail personalization platform serving 600+ retailers like Walmart and Nike
- Designed to be Multi-cloud/Multi-AZ and tuned to handle Black Friday

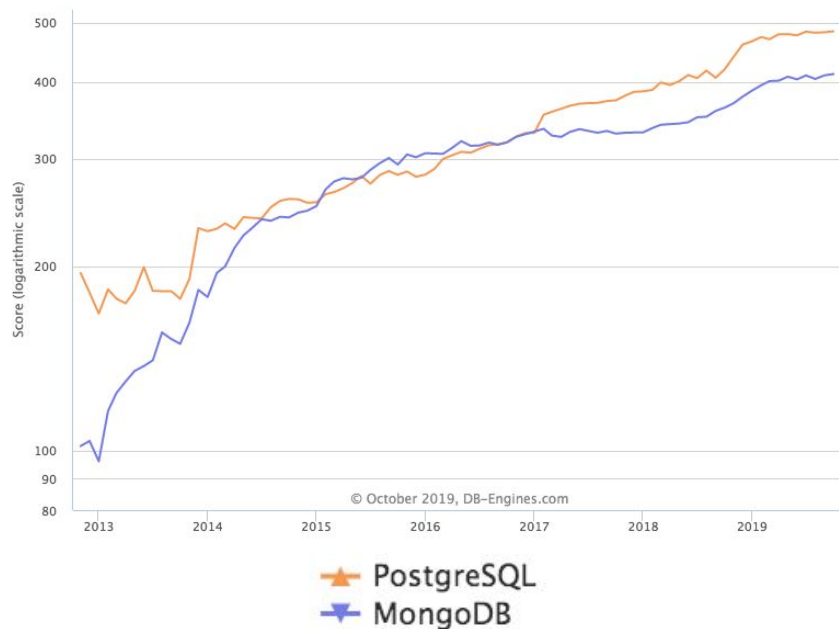


# YugabyteDB – The Promise of Distributed SQL



# Designing the Perfect Distributed SQL Database

PostgreSQL is more popular than MongoDB



Aurora much more popular than Spanner



## Amazon Aurora

A **highly available** MySQL and PostgreSQL-compatible relational database service

Not scalable but HA

All RDBMS features

PostgreSQL & MySQL



## Google Spanner

The first **horizontally scalable, strongly consistent**, relational database service

Scalable and HA

Missing RDBMS features

















New SQL syntax

# Comparing PostgreSQL and Google Spanner

Feature	PostgreSQL	Google Spanner
Query Layer	✓ Very well known	✗ New SQL flavor
SQL Feature Depth	High (many advanced features)	Low (basic features missing)
Fault Tolerance with HA	✗	✓
Horizontal Scalability	✗	✓
Distributed Txns	✗	✓
Replication	Async	Sync, Read Replicas

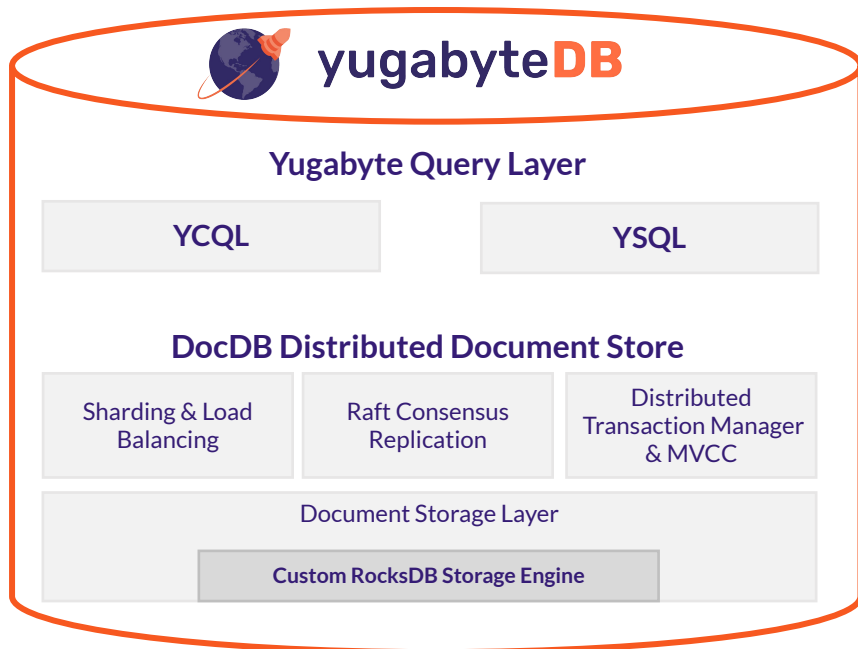
Feature set and ecosystem support critical for adoption

# Spanner Design + PostgreSQL Compatibility

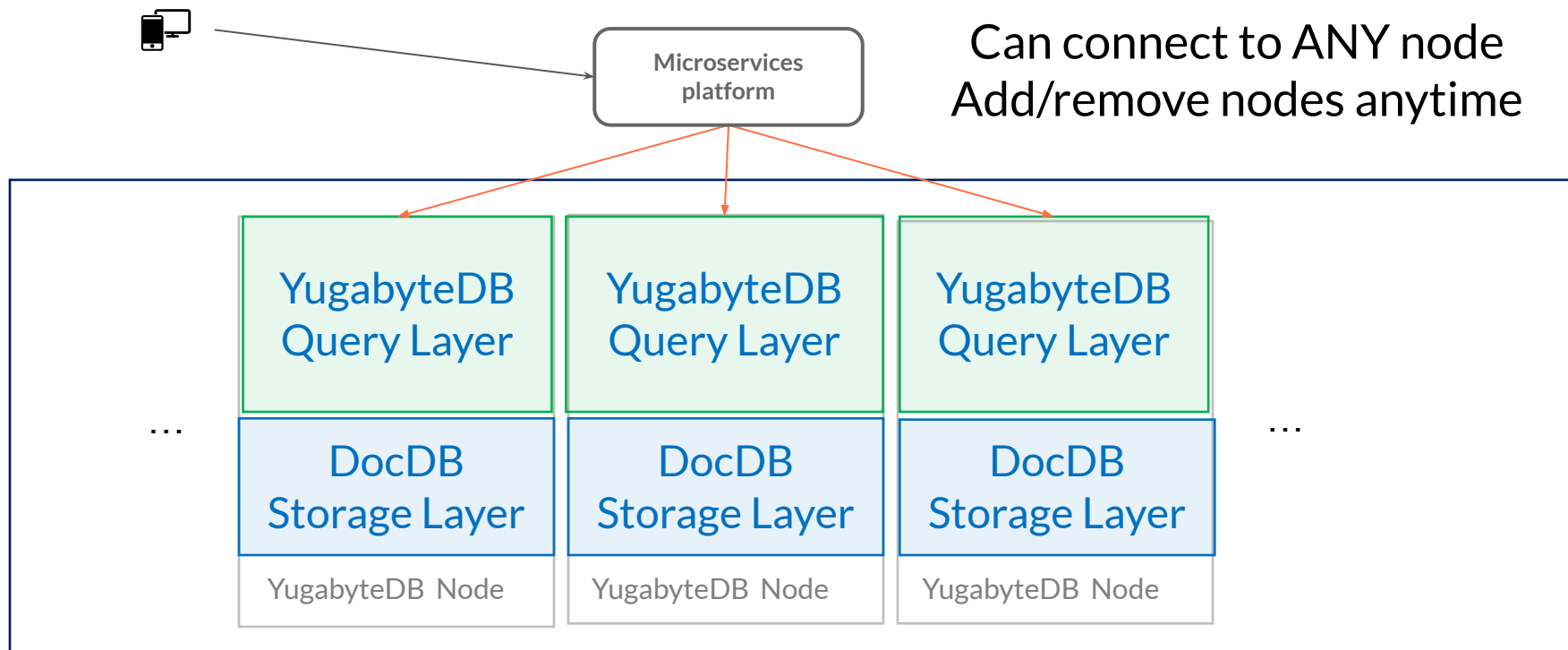
Feature	PostgreSQL	Google Spanner	 yugabyteDB
SQL Wire Protocol	 Very well known	 New SQL flavor	 Reuses PostgreSQL
RDBMS Feature Support	 High (many advanced features)	 Low (basic features missing)	 High (supports most Postgres features)
Fault Tolerance with HA			
Horizontal Scalability			
Distributed Txns			
Global Txn Replication	Async	Sync, Read Replicas	Sync, Read Replicas, Async

# Designed for Cloud Native Microservices

	PostgreSQL	Google Spanner	YugabyteDB
SQL Ecosystem	✓ Massively adopted	✗ New SQL flavor	✓ Reuse PostgreSQL
RDBMS Features	✓ Advanced Complex	✗ Basic cloud-native	✓ Advanced Complex and cloud-native
Highly Available	✗	✓	✓
Horizontal Scale	✗	✓	✓
Distributed Txns	✗	✓	✓
Data Replication	Async	Sync	Sync + Async

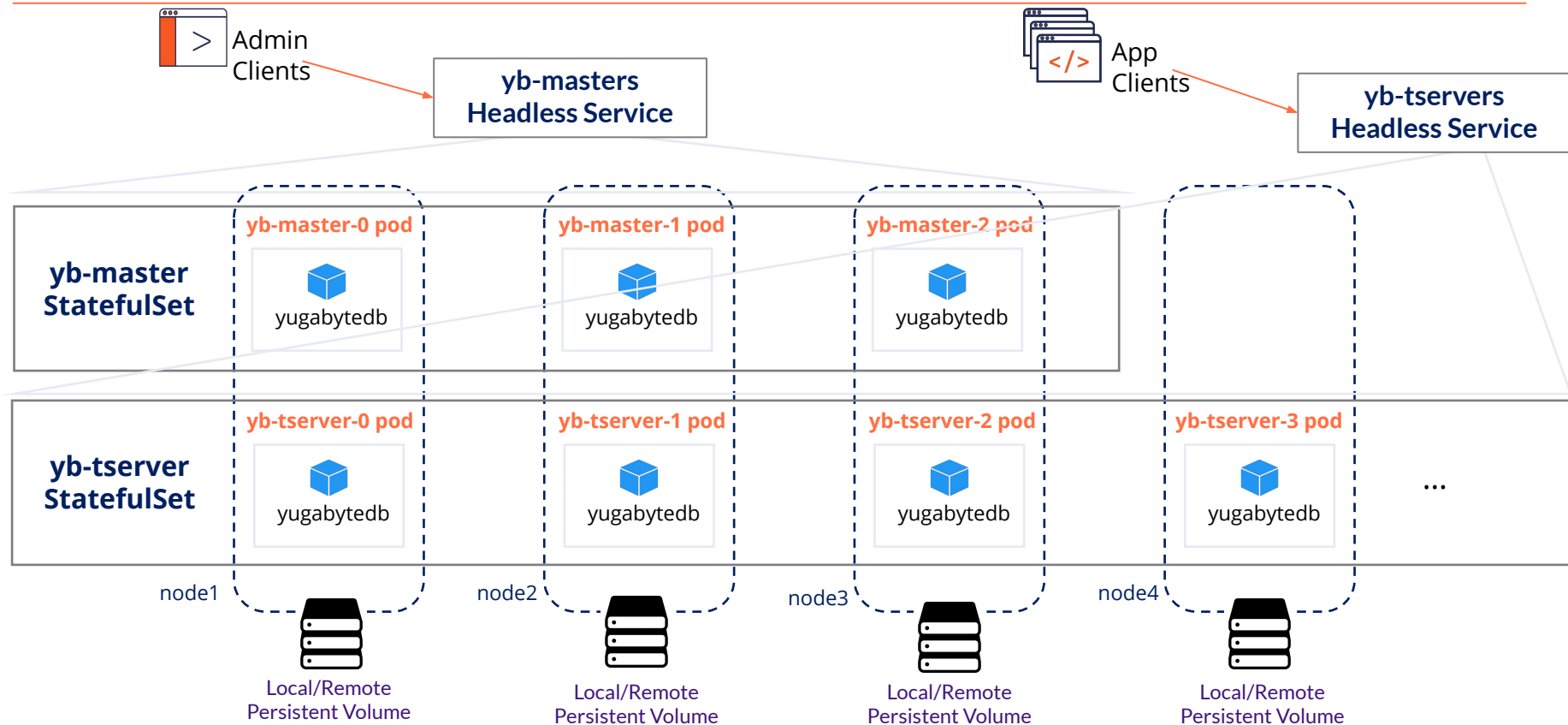


# All Nodes Are Identical

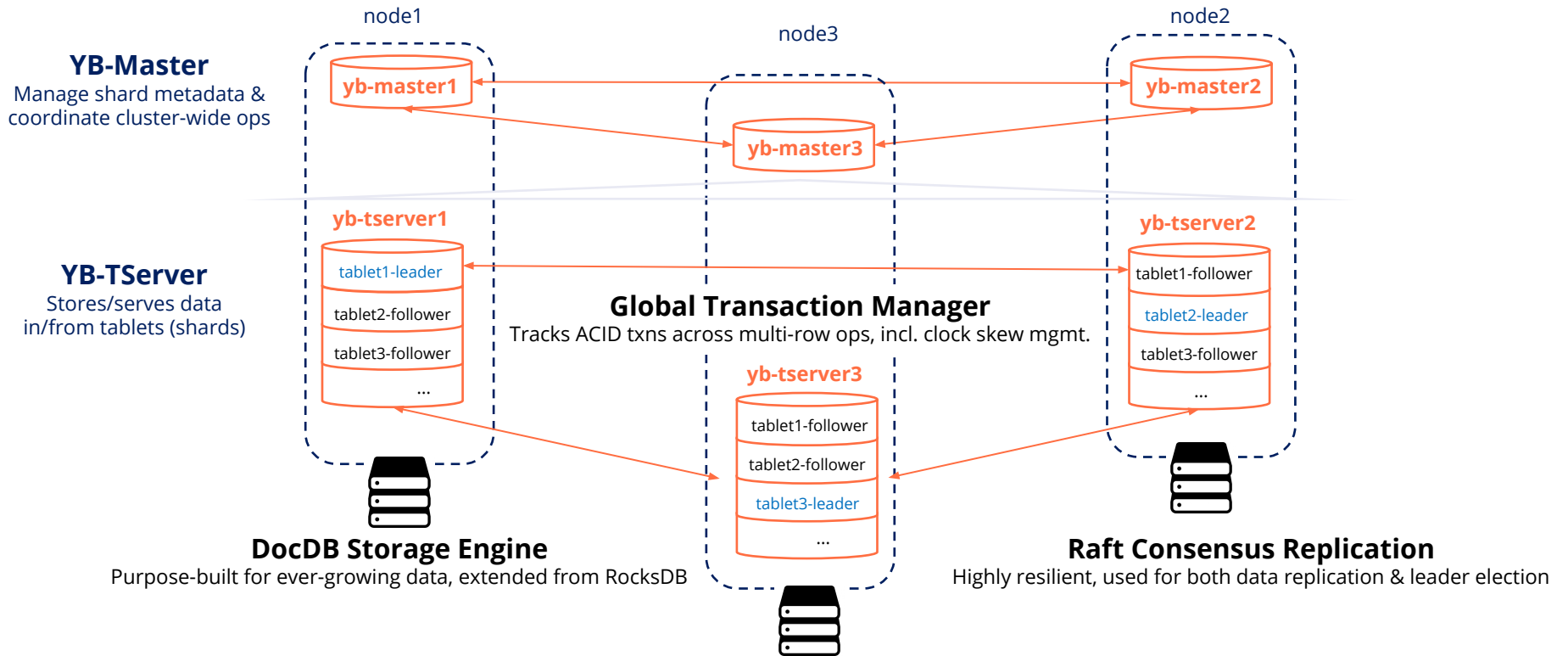




# YugabyteDB Deployed as StatefulSets

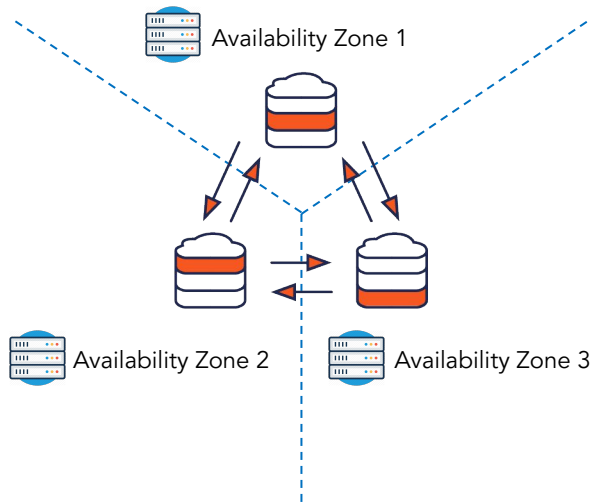


# Under the Hood – 3 Node Cluster



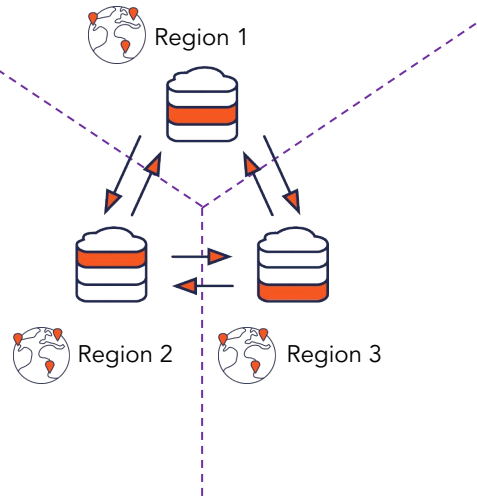
# Deployment Topologies

## 1. Single Region, Multi-Zone



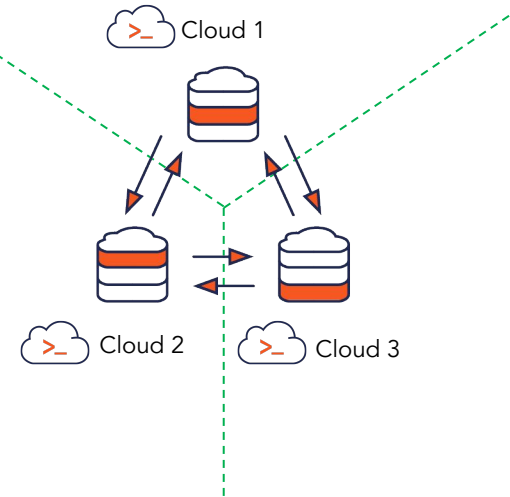
**Consistent Across Zones**  
No WAN Latency But No  
Region-Level Failover/Repair

## 2. Single Cloud, Multi-Region



**Consistent Across Regions**  
Cross-Region WAN Latency with  
Auto Region-Level Failover/Repair

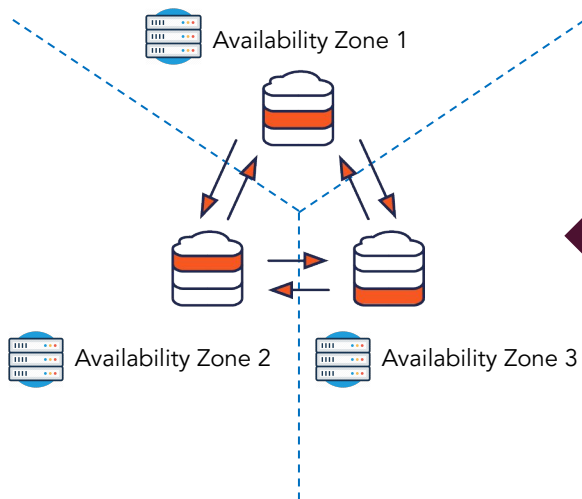
## 3. Multi-Cloud, Multi-Region



**Consistent Across Clouds**  
Cross-Cloud WAN Latency with Auto  
Cloud-Level Failover/Repair

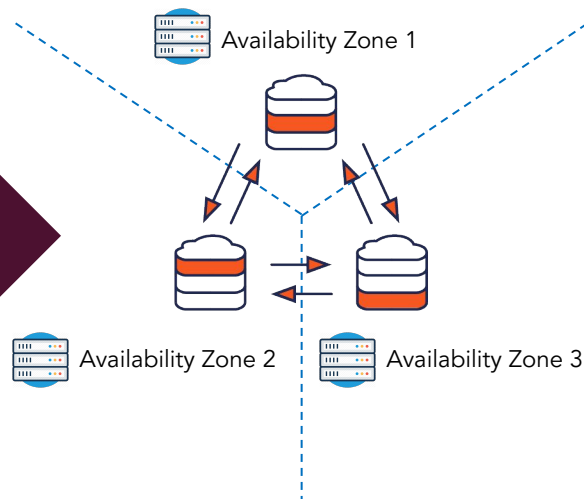
# Multi-Master Deployments w/ xCluster Replication

Master Cluster 1 in Region 1



Consistent Across Zones  
No Cross-Region Latency for Both Writes & Reads  
App Connects to Master Cluster in Region 2 on Failure

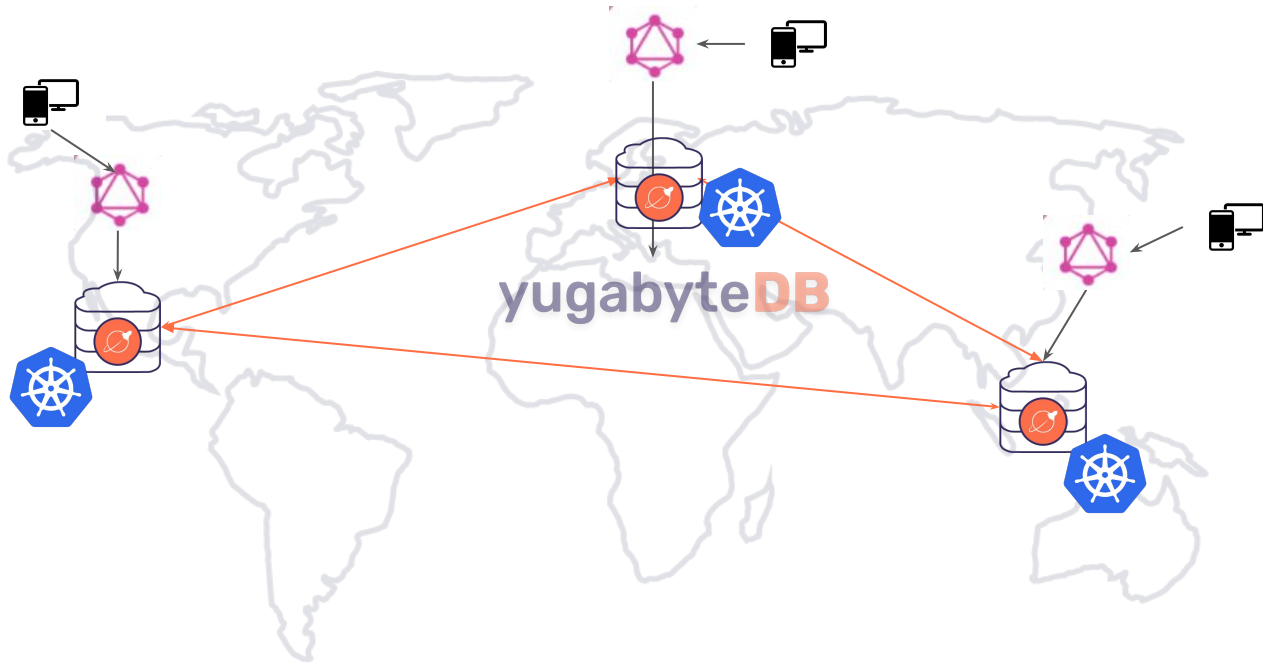
Master Cluster 2 in Region 2



Consistent Across Zones  
No Cross-Region Latency for Both Writes & Reads  
App Connects to Master Cluster in Region 1 on Failure



# Deploying Yugabyte on Multi-Region K8s



- Scalable and highly available data tier
- Business continuity
- Geo-partitioning and data compliance

# YugabyteDB on K8s Multi-Region Requirements

---

- Pod to pod communication over TCP ports using RPC calls across n K8s clusters
- Global DNS Resolution system
  - Across all the K8s clusters so that pods in one cluster can connect to pods in other clusters
- Ability to create load balancers in each region/DB
- RBAC: ClusterRole and ClusterRoleBinding
- Reference:  
Deploy YugabyteDB on multi cluster GKE  
<https://docs.yugabyte.com/latest/deploy/kubernetes/multi-cluster/gke/helm-chart/>

# YugabyteDB on K8s

Demo – Single YB Universe Deployed on 3 GKE Clusters

# YugabyteDB Universe on 3 GKE Clusters



## Deployment:

3 GKE clusters

Each with 3 x **N1 Standard 8** nodes

3 pods in each cluster using 4 cores

**Cores:** 4 cores per pod

**Memory:** 7.5 GB per pod

**Disk:** ~ 500 GB total for universe



# Yugabyte Platform

Demo

# Ensuring High Performance

## LOCAL STORAGE

Since v1.10

Lower latency, Higher throughput

Recommended for workloads that do their own replication

Pre-provision outside of K8s

Use SSDs for latency-sensitive apps

```
volumes:  
- name: datadir0  
  hostPath:  
    path: "/mnt/disks/ssd0"  
- name: datadir1  
  hostPath:  
    path: "/mnt/disks/ssd1"  
nodeSelector:  
  cloud.google.com/gke-local-ssd: "true"
```

## REMOTE STORAGE

Most used

Higher latency, Lower throughput

Recommended for workloads do not perform any replication on their own

Provision dynamically in K8s

Use alongside local storage for cost-efficient tiering

```
volumeClaimTemplates:  
- metadata:  
    name: datadir0  
  spec:  
    accessModes: [ "ReadWriteOnce" ]  
    resources:  
      requests:  
        storage: 10Gi
```

# Configuring Data Resilience

## POD ANTI-AFFINITY

Pods of the same type should not be scheduled on the same node

Keeps impact of node failures to absolute minimum

```
spec:
  affinity:
    podAntiAffinity:
      preferredDuringSchedulingIgnoredDuringExecution:
      - weight: 100
        podAffinityTerm:
          labelSelector:
            matchExpressions:
            - key: app
              operator: In
              values:
              - yb-tserver
          topologyKey: kubernetes.io/hostname
```

## MULTI-ZONE/REGIONAL/MULTI-REGION POD SCHEDULING

Multi-Zone – Tolerate zone failures for K8s worker nodes

Regional – Tolerate zone failures for both K8s worker and master nodes

Multi-Region / Multi-Cluster – Requires network discovery between multi cluster

# Automating Day 2 Operations



## HANDLING FAILURES

Pod failure handled by K8s automatically

Node failure has to be handled manually by adding a new slave node to K8s cluster

Local storage failure has to be handled manually by mounting new local volume to K8s



## ROLLING UPGRADES

Supports two *upgradeStrategies*:  
*onDelete* (default) and  
*rollingUpgrade*

Pick rolling upgrade strategy for DBs that support zero downtime upgrades such as YugabyteDB

New instance of the pod spawned with same network id and storage



## BACKUP & RESTORE

Backups and restores are a database level construct

YugabyteDB can perform distributed snapshot and copy to a target for a backup

Restore the backup into an existing cluster or a new cluster with a different number of TServers

# Extending StatefulSets with Operators

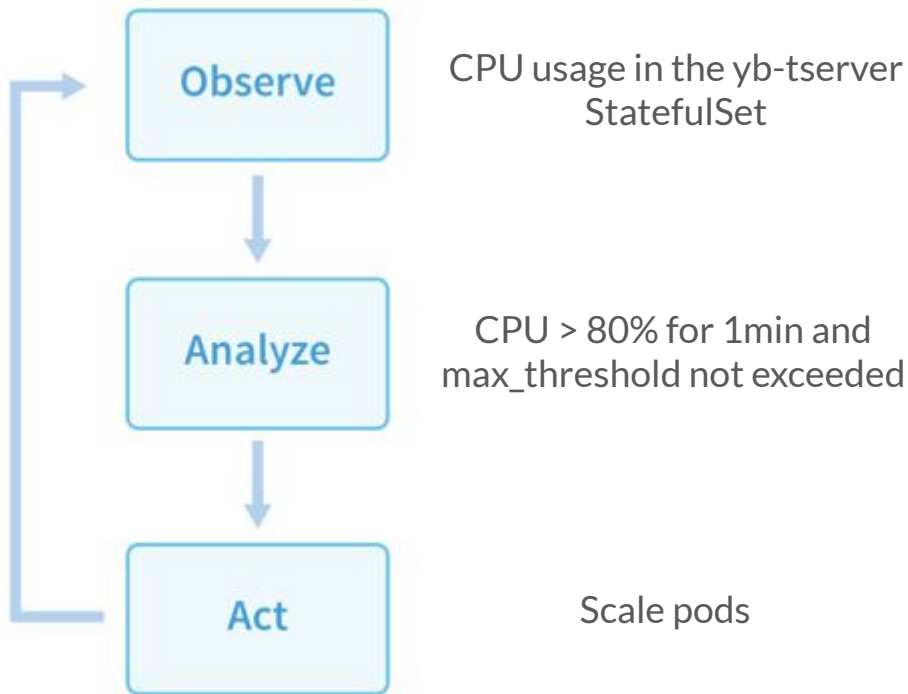


An Operator represents human operational knowledge in software to reliably manage an application.

Based on Custom Controllers that have direct access to lower level K8S API

Excellent fit for stateful apps requiring human operational knowledge to correctly scale, reconfigure and upgrade while simultaneously ensuring high performance and data resilience

Complementary to Helm for packaging

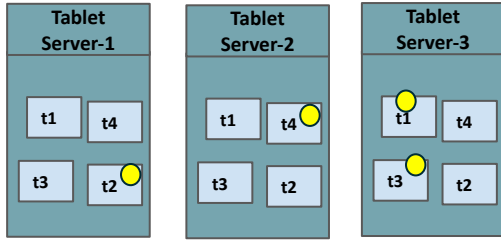


<https://github.com/yugabyte/yugabyte-platform-operator>

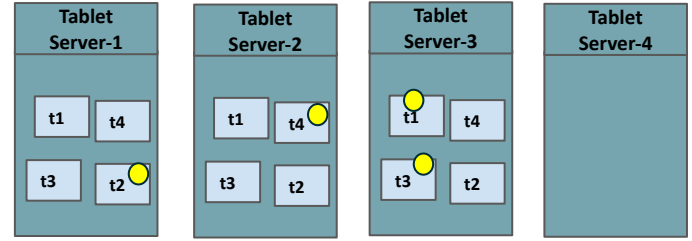
# Yugabyte Platform

Demo

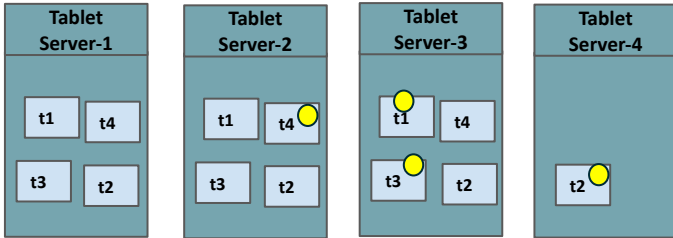
# Live Demo: Cluster Scale Up



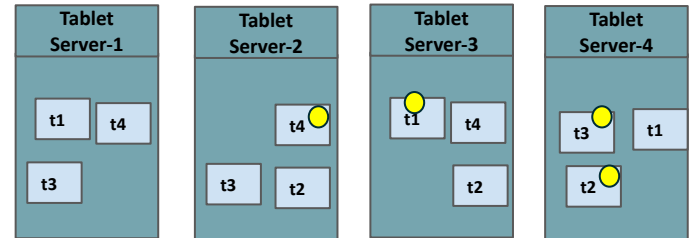
1 Before expansion, all 3 nodes taking traffic.



2 New node just added. No traffic to new node yet.



3 New node received t2 from current Leader, which is guaranteed to have consistent copy. A simple file transfer and with just that one Tablet, it is ready to take traffic. Expansion is still in progress.



4 Zero-downtime cluster expansion and much faster because of Raft and strong consistency.

# Live Demo: Terminate a YB Pod

---

→ `~ kubectl delete pods yb-tserver-1 -n yb-dev-yb-webinar-us-west1-c`

→ `~ kubectl get pods -n yb-dev-yb-webinar-us-west1-c`



# A Classic Enterprise App Scenario

# Yugastore – E-Commerce App : A Real-World Demo

---



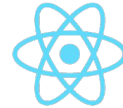
yugabyte**DB**



Istio



APACHE  
**Spark**<sup>TM</sup>



React

Deployed on



<https://github.com/yugabyte/yugastore-java>

# Yugastore – Kronos Marketplace

 KronosMarketplace



Books



Music



Beauty

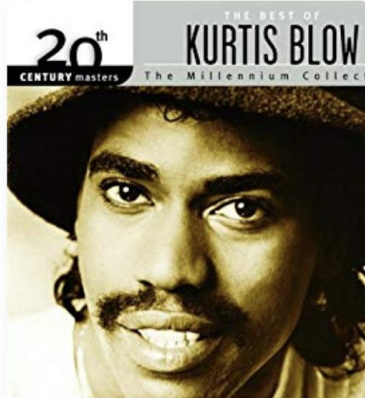


Electronics



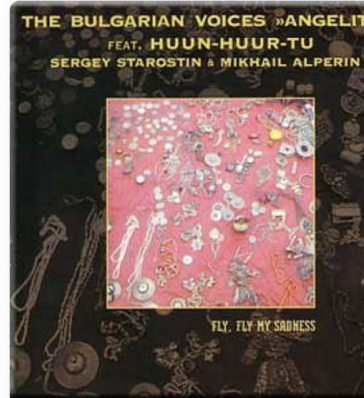
Cart

## Music



462 stars from 106 reviews

The Best of Kurtis Blow



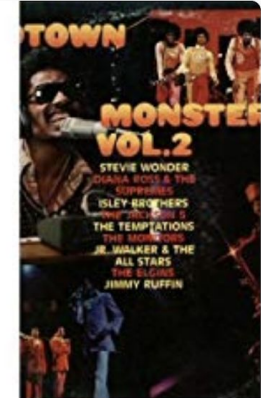
1888 stars from 625 reviews

Fly, Fly My Sadness -  
Angelite feat. Huun-Huur-Tu



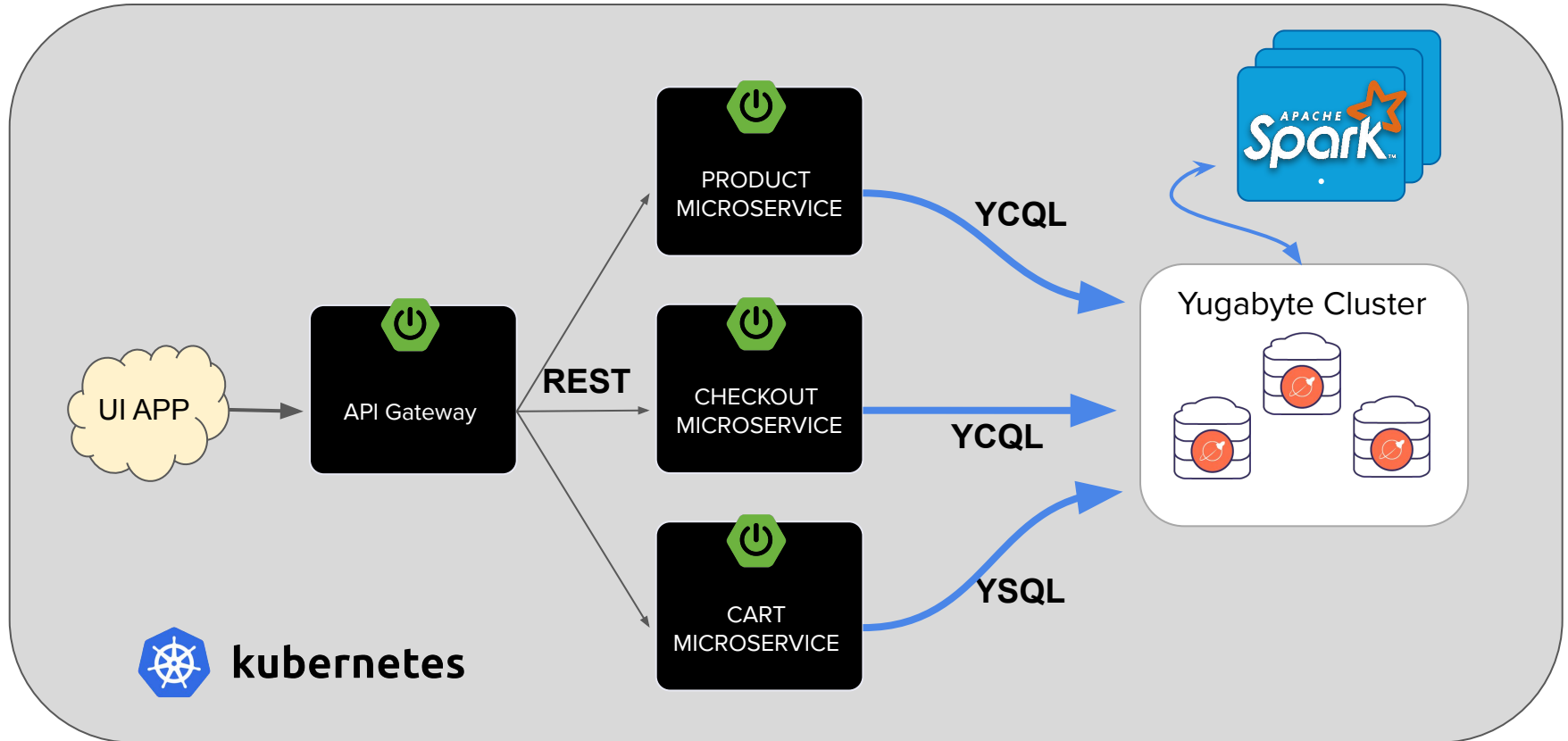
744 stars from 152 reviews

Symphony of the Night

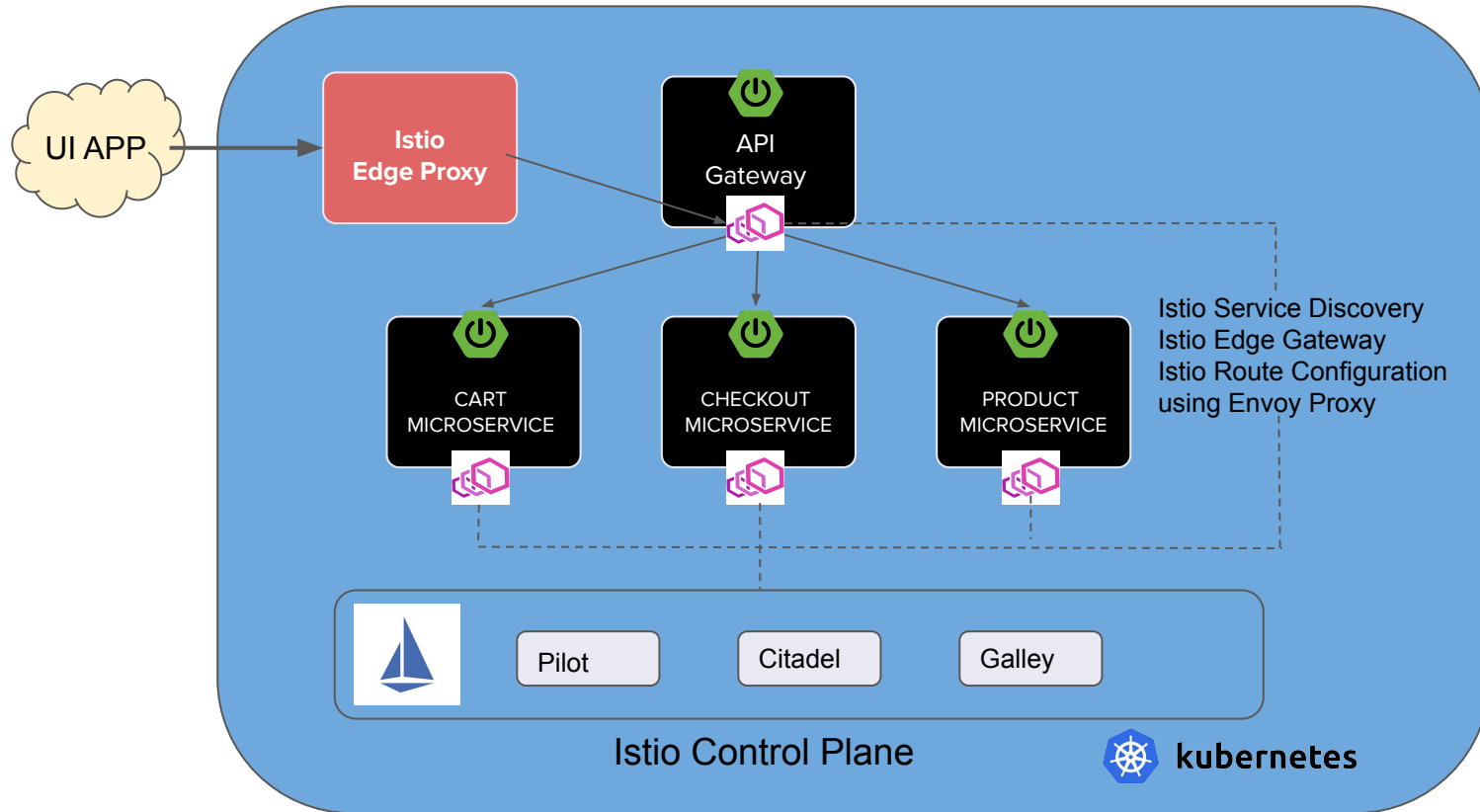


2738 stars from 622 reviews

# Classic Enterprise Microservices Architecture

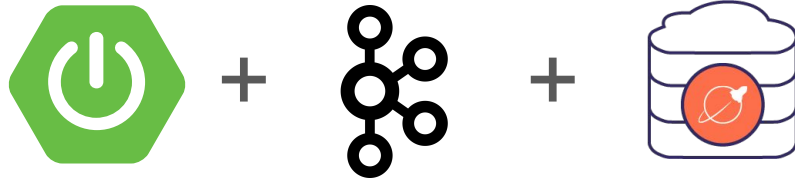


# Istio Traffic Management for Microservices



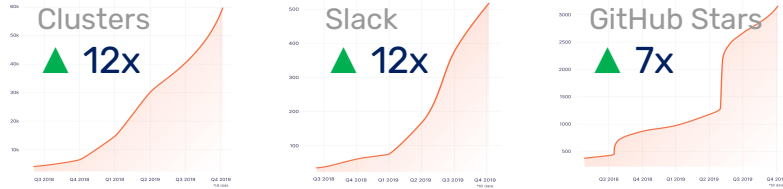
# A Platform Built for a New Way of Thinking

---



- Event + Microservice first design
- Team autonomy with platform efficiency
- 100% Cloud Native operating model on K8s
- Turnkey multi-cloud
- Full Spring Data support

# We're a Fast Growing Project



Growth in 1 Year

Powers business-critical apps at scale.



**27B+**

Ops/Day

**xignite**

**10B+**

Ops/Day



**3B+**

Ops/Day



**1B+**

Ops/Day

We  stars! Give us one:  
[github.com/YugaByte/yugabyte-db](https://github.com/YugaByte/yugabyte-db)

Join our community:  
[yugabyte.com/slack](https://yugabyte.com/slack)

# Thank You

Join us on Slack: [yugabyte.com/slack](https://yugabyte.com/slack)

Star us on GitHub: [github.com/yugabyte/yugabyte-db](https://github.com/yugabyte/yugabyte-db)

