# yugabyted
# Jumpstart with YugabyteDB

**Yogi Rampuria**
Principal Solution Engineer, YugabyteDB
@yogendra | github.com/yogendra

yugabyteDB

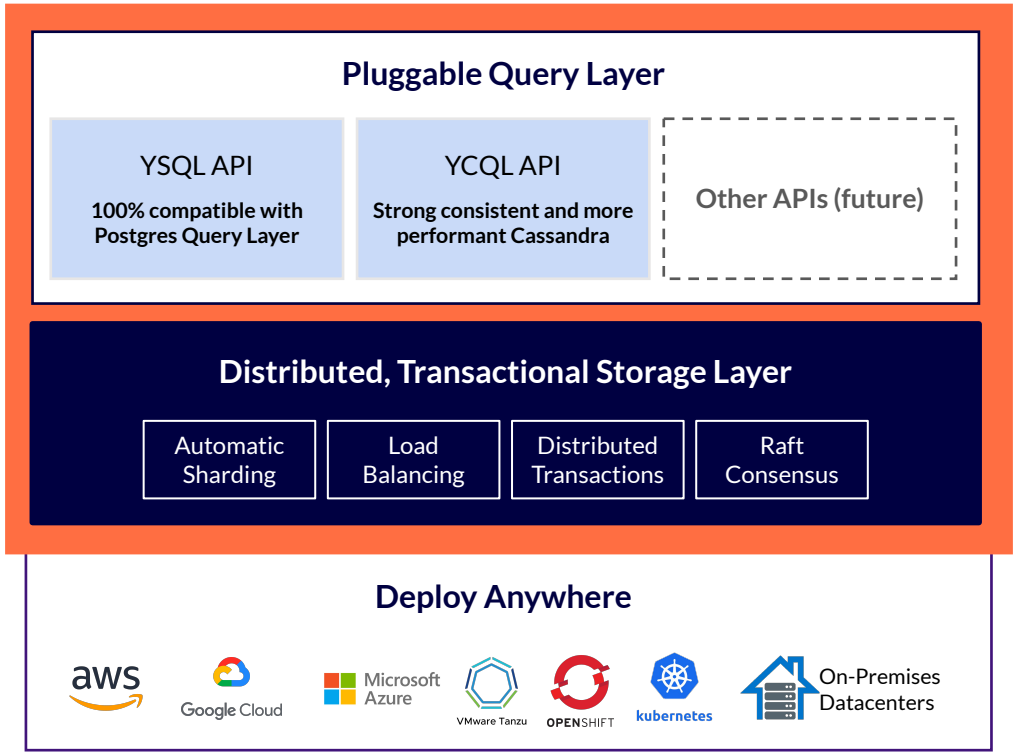
# Agenda

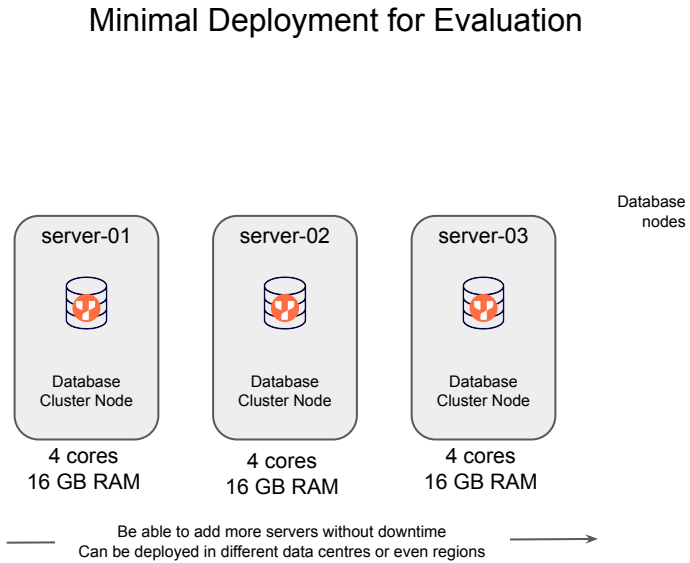
Basics of YugabyteDB
yugabyted - Core tools for running YugabyteDB
yugabyted-ui - New YugabyteDB UI
Other tools


Distributed PostgreSQL for business-critical apps delivered as a flexible service

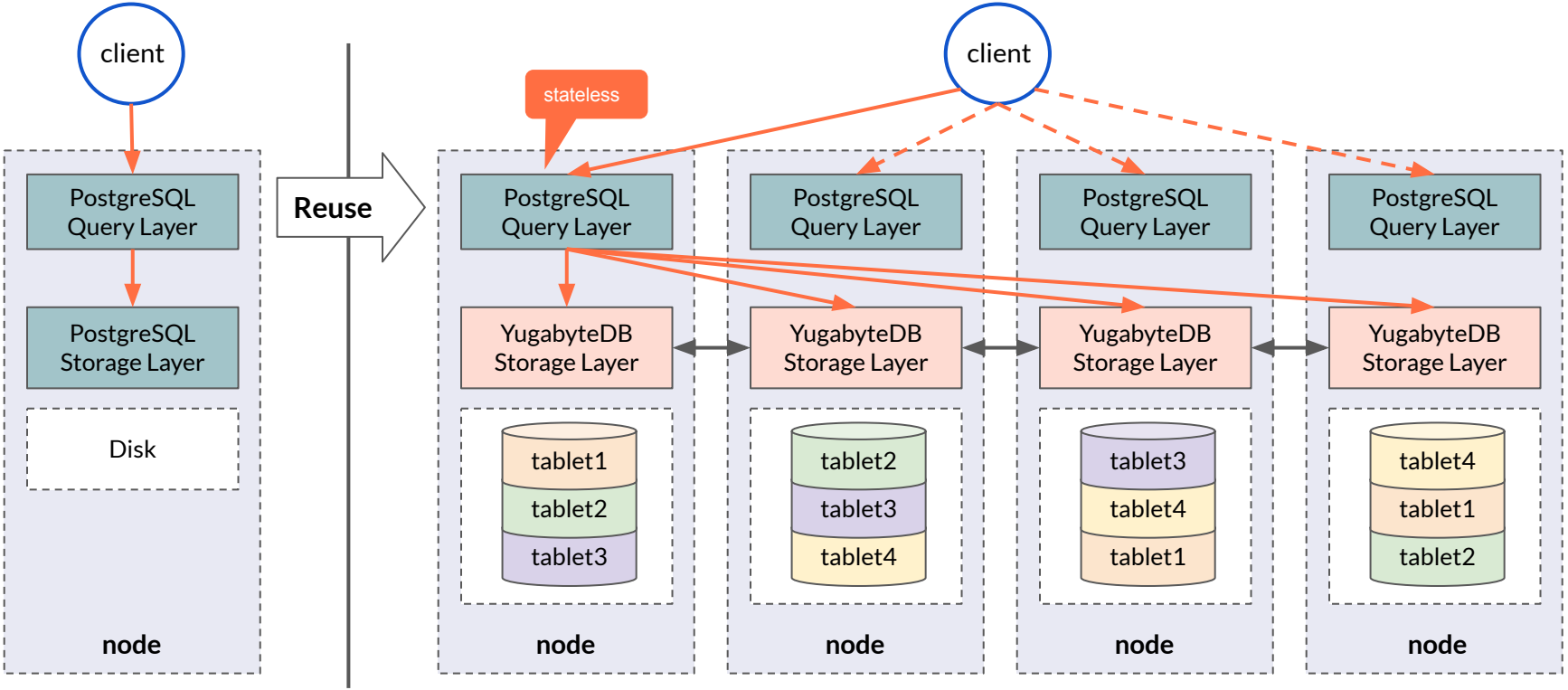
# Basics of YugabyteDB

# Innovative architecture combines best of databases

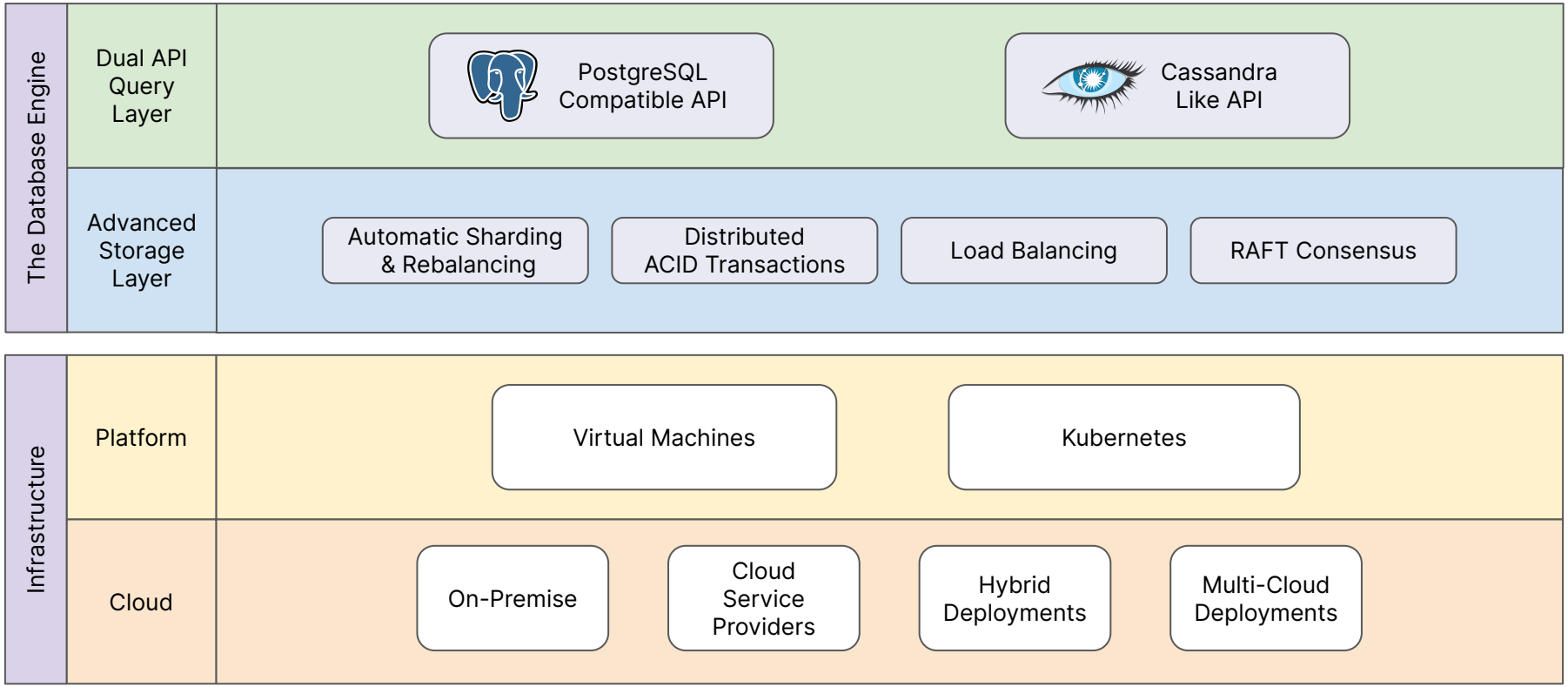


# How Does YSQL Work



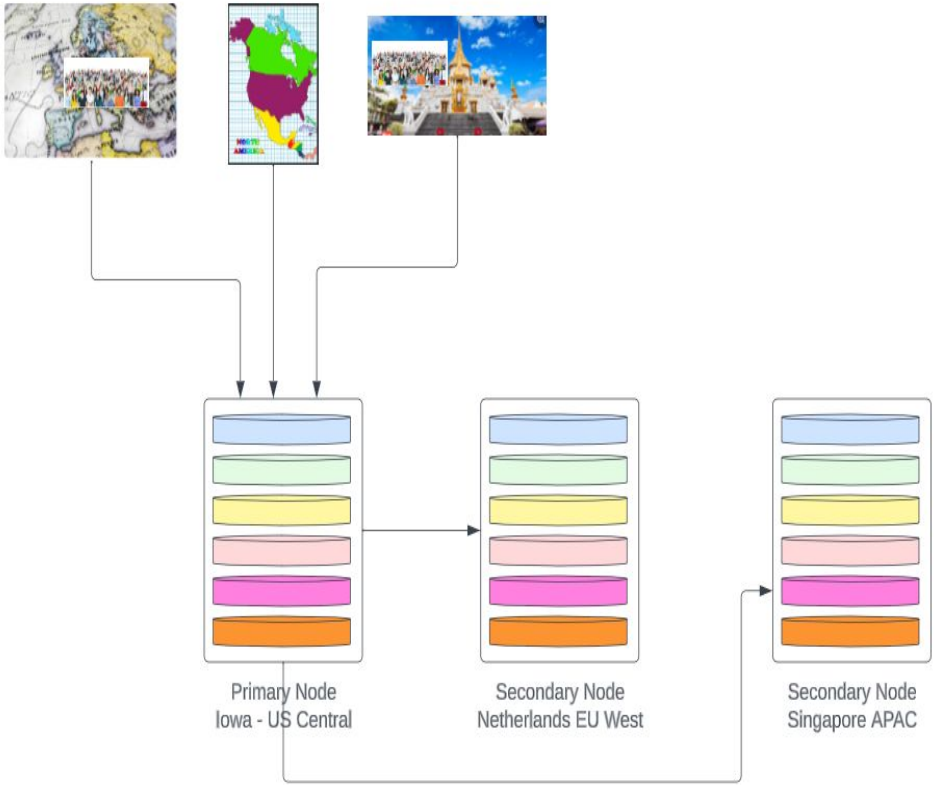
# Logical Deployment of yugabyteDB



# YugabyteDB Key Advantages over Legacy RDBMS

# Classic RDBMS Architecture with PostgreSQL/Oracle

- 3 nodes distributed across zones/regions.
- Illustrates a table with 6 partitions.
- Only one primary at a time: all writes go to that primary irrespective of origination or data partition.
- Secondary nodes are essentially for failover purposes.
- Scaling out horizontally simply increases secondary nodes.

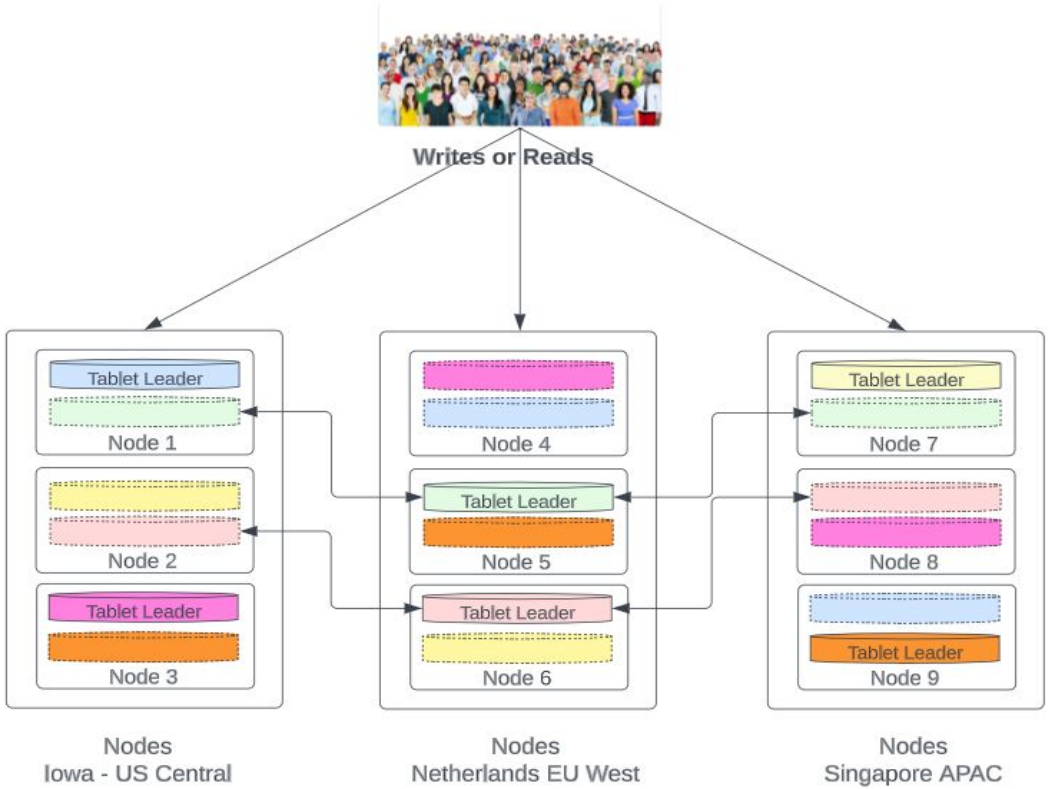


# Scaling to Nine Nodes

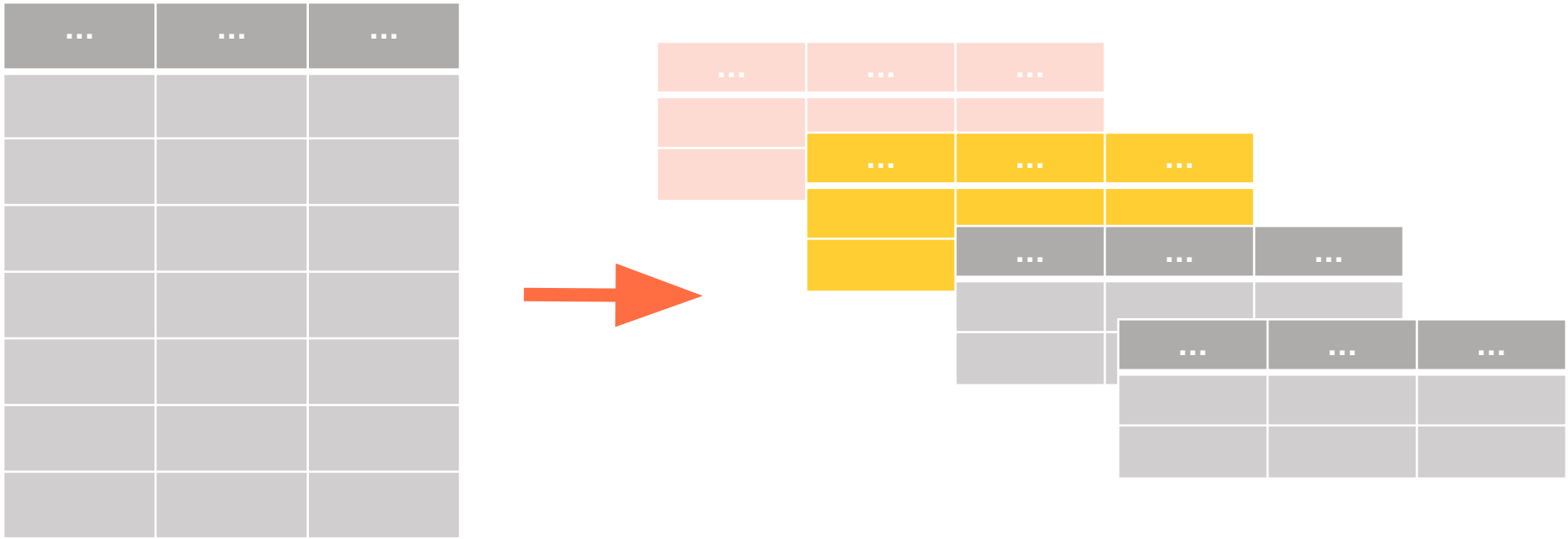
- 9 nodes distributed across regions.
- Replication factor of 3.
- 1 replica per region.
- 3 nodes uniformly distributed per region.
- Leader and Follower distributed and balanced across the region.
- Survives multiple node failures.
- Automatically rebalances nodes in case of failure.

Tablet Leader: Solid lines :

Follower Tablet: Dash lines:

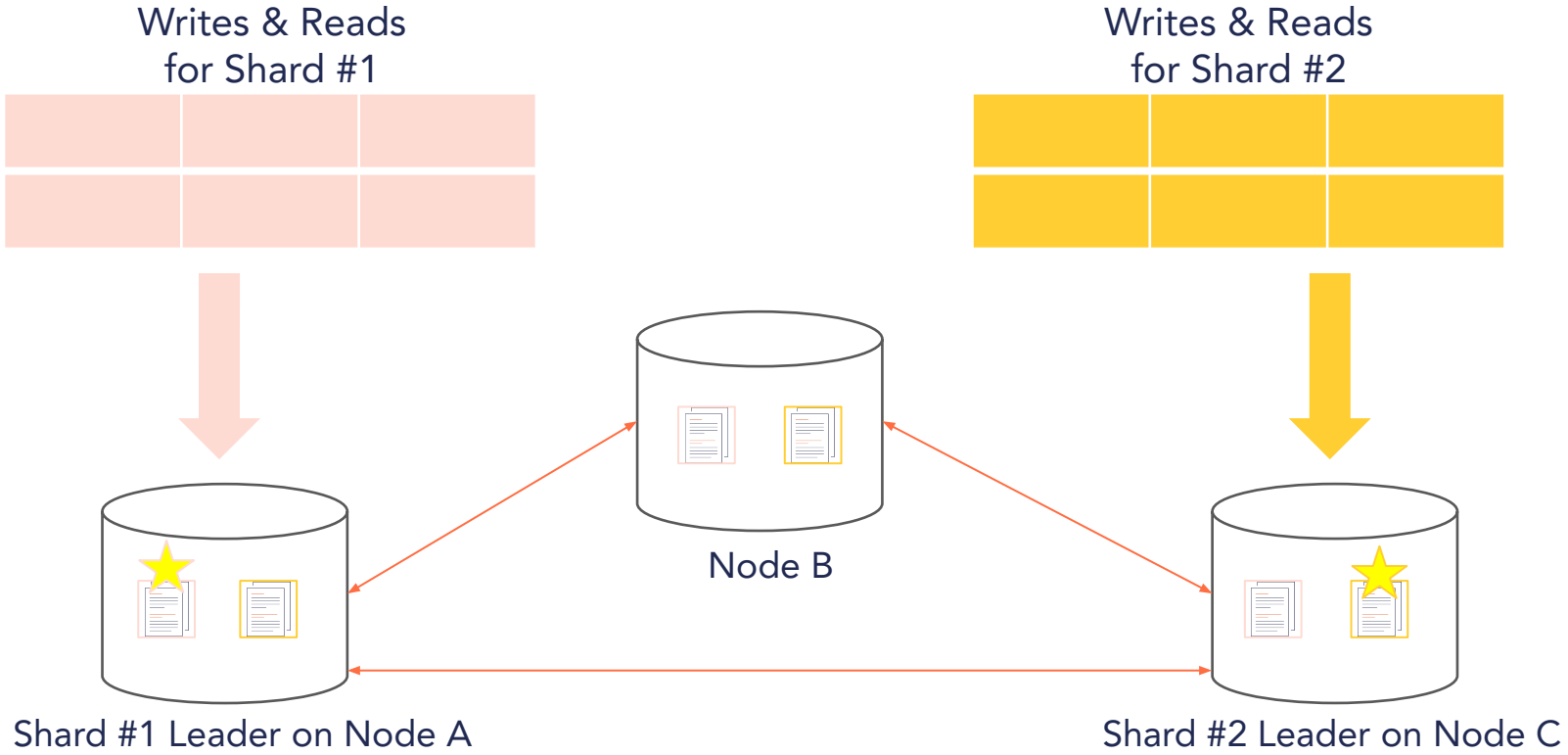


# Every Table is Automatically Sharded

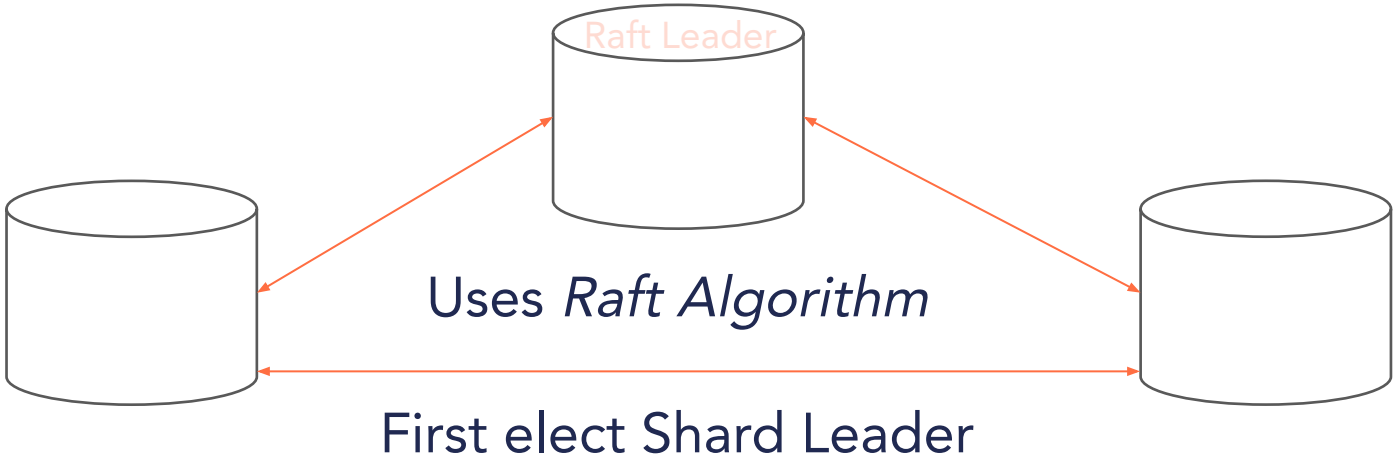


SHARDING = **AUTOMATIC PARTITIONING OF TABLES**

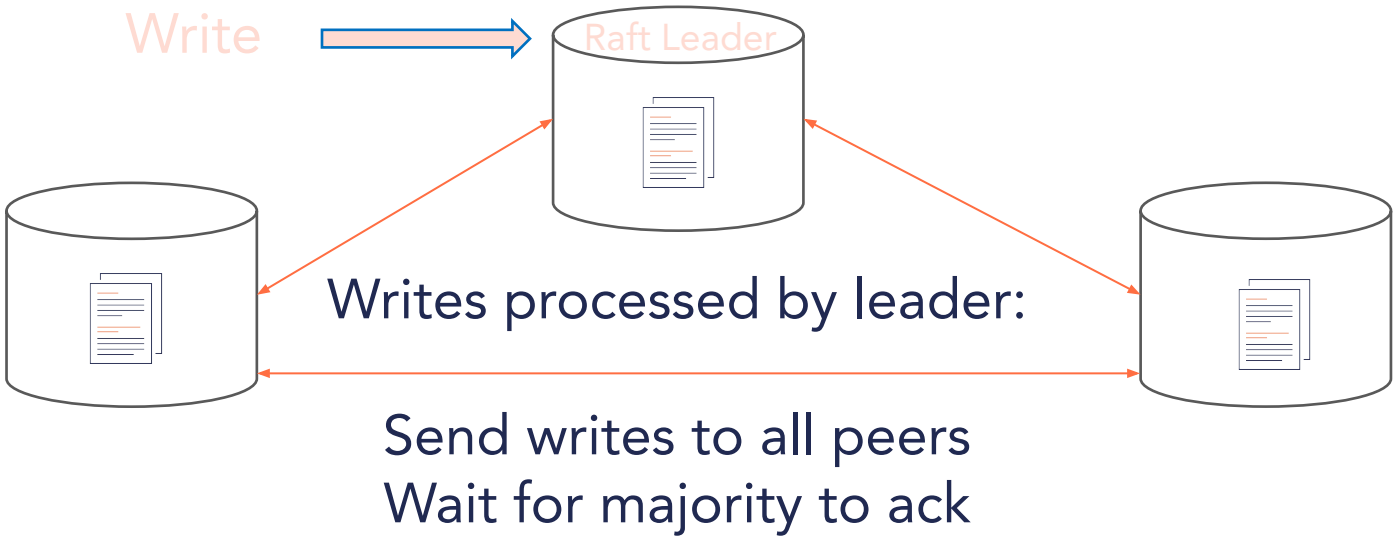
# Replication Done at Shard Level



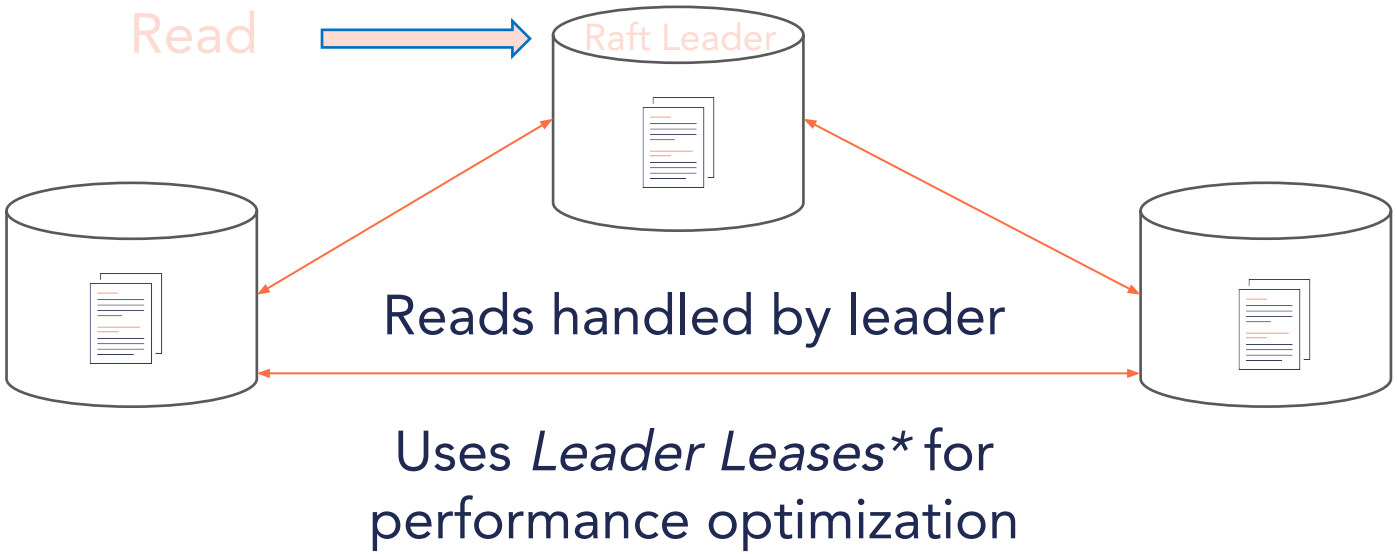
# Replication uses a Consensus algorithm



# Writes in Raft Consensus



# Reads in Raft Consensus



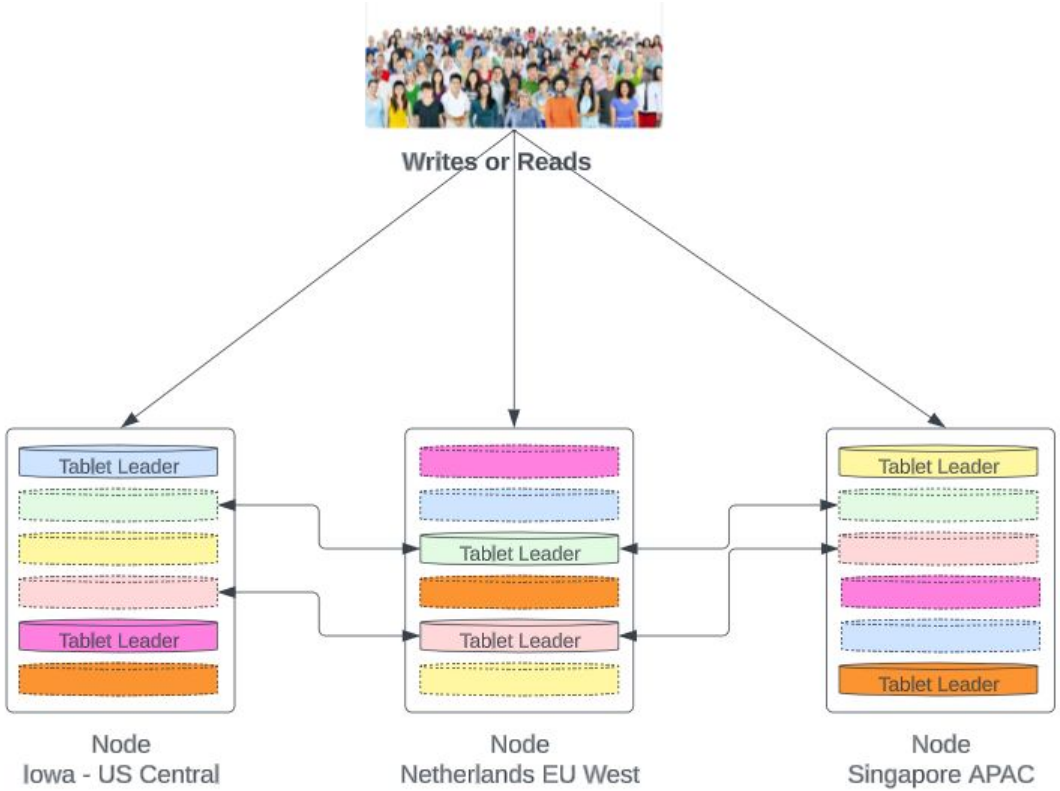

*Leader Leases: https://blog.yugabyte.com/low-latency-reads-in-geo-distributed-sql-with-raft-leader-leases/

# YugabyteDB Distributed RDBMS Architecture

- 3 nodes distributed across zones/regions with replicatic factor of 3
- Illustrates a table with 6 partitions/Tablets/Shards.
- Leader and Follower distributed and balanced across the nodes.
- Survives any 1 node failure.
- Scaling out horizontally both writes and reads.

Tablet Leader: Solid lines :

Follower Tablet: Dash lines:

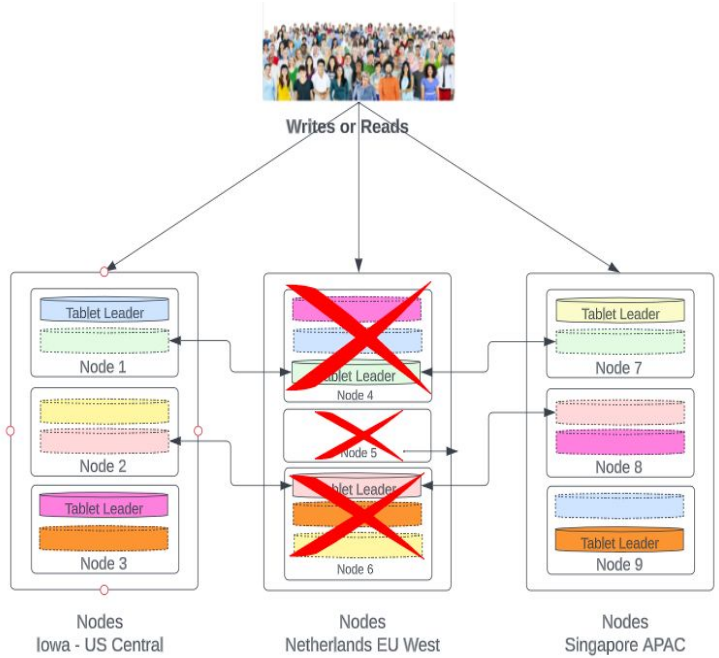


# Tolerating Node Outage with YugabyteDB

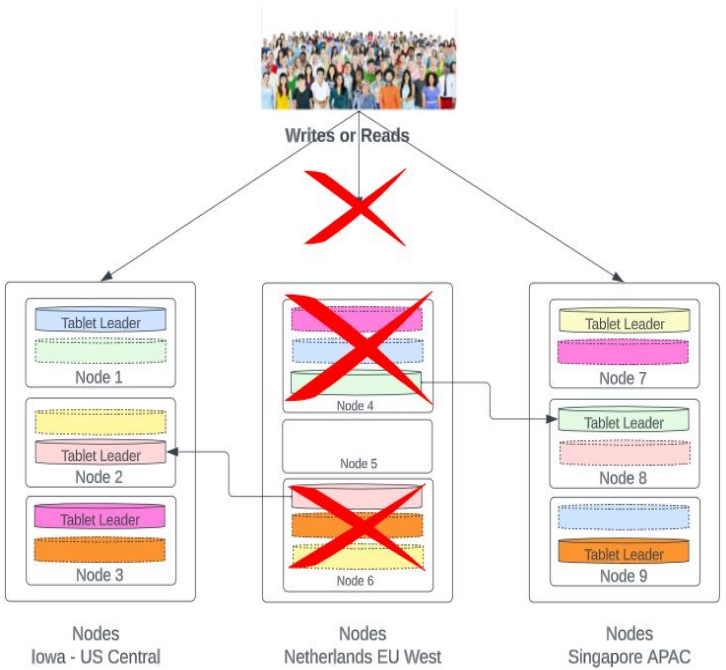

- Tablet leader on node 5 first fails over to a follower node on say node 1 or 7 in seconds.
- Leader re-election on sustaining follower tablets completes in seconds

- Tablet leader auto-rebalances to node 4 as shown within a few mins.
- Failed node recovery automatically catches up.

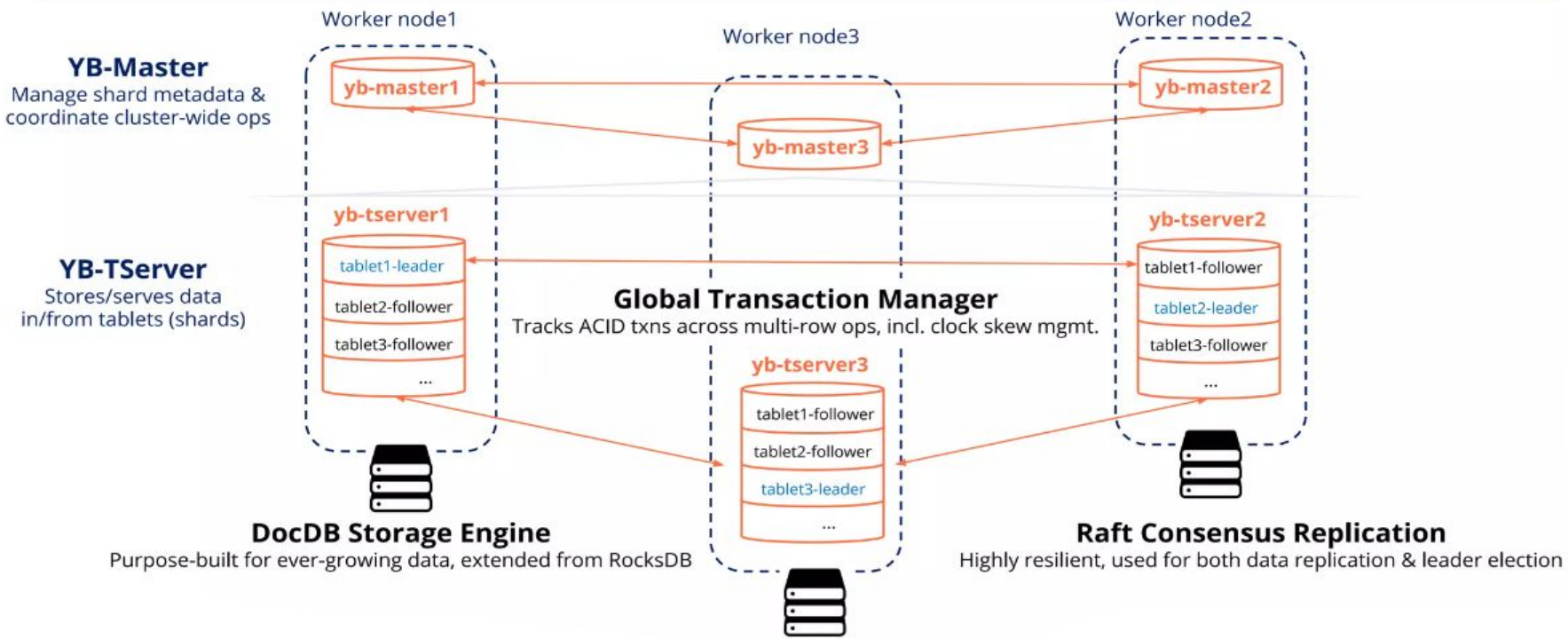
# Tolerating a Region Outage with YugabyteDB




- New Tablet leaders automatically evenly rebalanced.
- Automatic self-healing and re-replication up on failed node recovery.

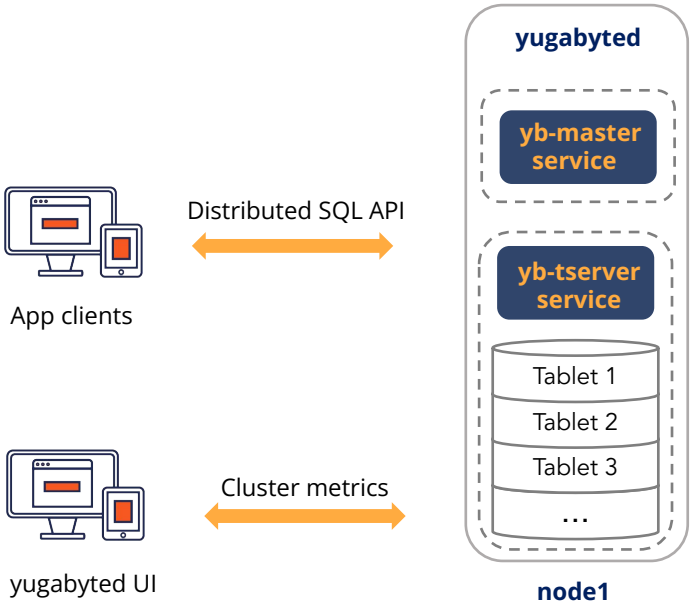


# High-Level Architecture: Under the Hood of a 3-Node Cluster



# yugabyted Overview

# yugabyted overview




... Scale to as many nodes as needed

- **yugabyted** is a daemon service for starting and managing the YugabyteDB cluster
- simple and opinionated approach for launching different deployment topologies
- consistent getting started experience across cloud and on-prem environments
- pre-requisite checks run based on the OS
- out of the box UI for managing the cluster
- currently yugabyted service is in **preview**. GA soon.

**yb-tserver service** Store & serve app data in/from tablets (aka shards)

**yb-master service** Manage tablet/shrd metadata & coordinate config changes

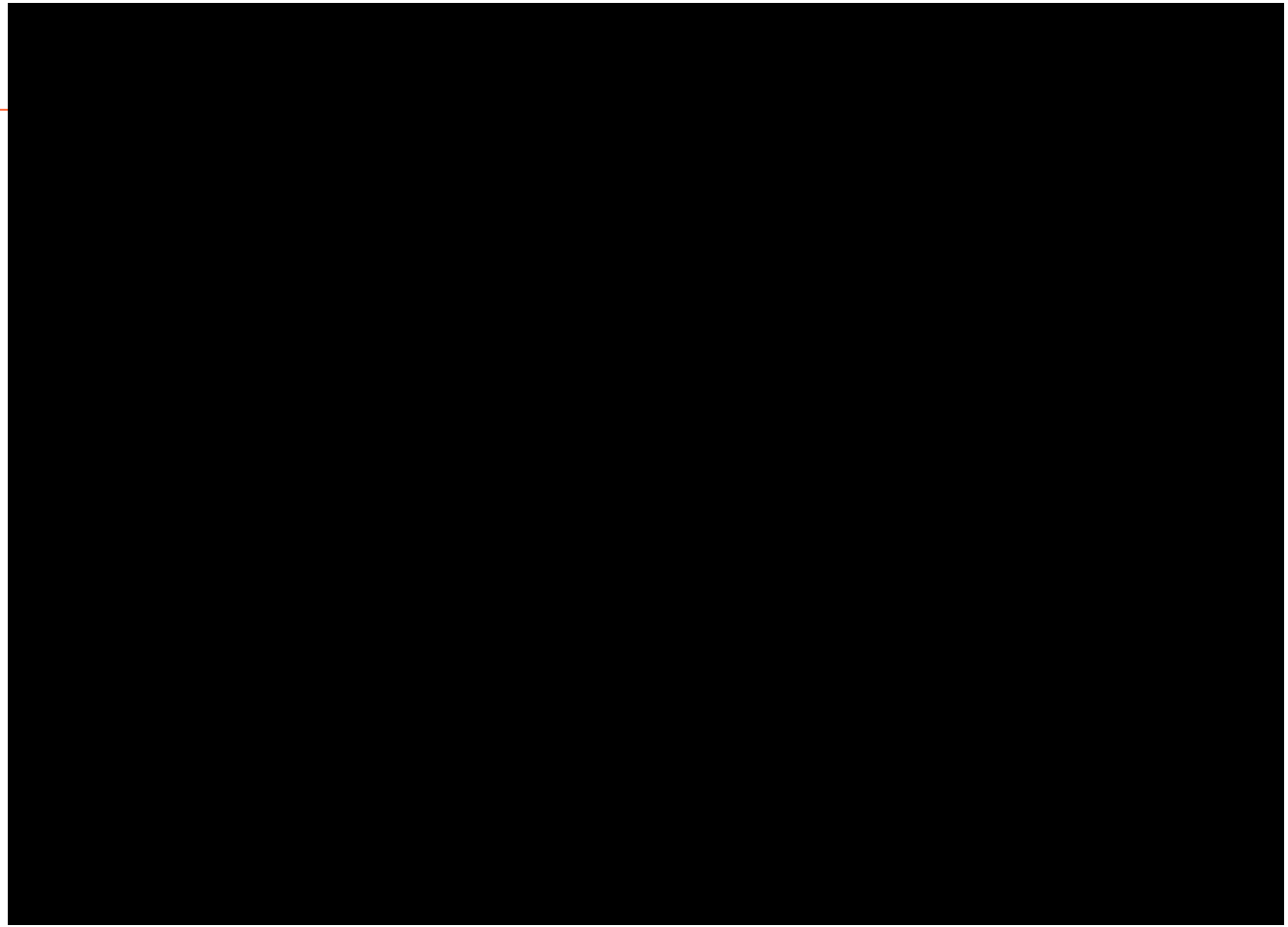
# Starting your YugabyteDB local instance

```
$ ./bin/yugabyted start
```

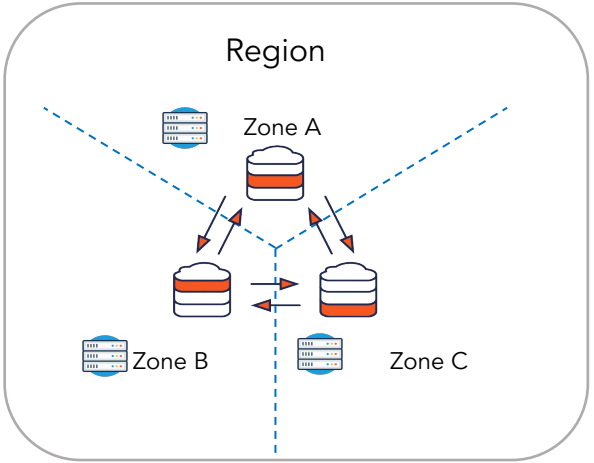
```
Starting yugabyted...
✅ System checks
✅ YugabyteDB Started
⚠ WARNING: Cluster started in an insecure mode without authentication and encryption enabled. For non-production
 use only, not to be used without firewalls blocking the internet traffic.

+-------------------------------------------------------------------------------------------------------------+
|                                              yugabyted                                                      |
+-------------------------------------------------------------------------------------------------------------+
| Status              : Running.                                                                              |
| Replication Factor  : 1                                                                                     |
| Web console         : http://10.150.0.157:7000                                                              |
| JDBC                : jdbc:postgresql://10.150.0.157:5433/yugabyte?user=yugabyte&password=yugabyte          |
| YSQL                : bin/ysqlsh -h 10.150.0.157  -U yugabyte -d yugabyte                                   |
| YCQL                : bin/ycqlsh 10.150.0.157 9042 -u cassandra                                             |
| Data Dir            : /home/gargsans/var/data                                                               |
| Log Dir             : /home/gargsans/var/logs                                                               |
| Universe UUID       : 040684bf-ba6c-45a7-ae5a-aa9c1fe67cc2                                                  |
+-------------------------------------------------------------------------------------------------------------+
🚀 YugabyteDB started successfully! To load a sample dataset, try 'yugabyted demo'.
🎉 Join us on Slack at https://www.yugabyte.com/slack
👕 Claim your free t-shirt at https://www.yugabyte.com/community-rewards/
```

# yugabyted Demo

1. Ease of getting started with a local install (single node)
2. Ease of progressively building a multi-AZ 6-node cluster



# Multi-zone topology using yugabyted




```
$ ./bin/yugabyted start
--cloud_location=cloud.region.zone-a
--fault_tolerance=zone
```

**Step 1**

Start the yugabyted with the `--cloud_location` flag

- cloud location will be `cloud.region.zone` details corresponding to VM/container
- For example: Deploying to AWS EC2 instance in ap-southeast-1a, cloud location will be `aws.ap-southeast-1.ap-southeast-1a`

- `--fault_tolerance` flag will be set to **zone**.

  YugabyteDB will be configured to survive zone failure.

# Joining the YugabyteDB cluster

```
$ ./bin/yugabyted start
--cloud_location=cloud.region.zone-b
--fault_tolerance=zone
--join=<ip-first-node>
```

**Step 2**

- After the first node is started, all the other nodes join the first node using `--join` flag
- Repeat this step on all the nodes joining the YugabyteDB cluster

# Configure multi-zone configuration using yugabyted

```
$ ./bin/yugabyted configure --fault_tolerance=zone
```

```
+-----------------------------------------------------------------------------------------------+
|                                          yugabyted                                            |
+-----------------------------------------------------------------------------------------------+
| Status              : Configuration successful. Primary data placement is geo-redundant       |
| Fault Tolerance     : Universe can survive at least 1 availability zone failure               |
+-----------------------------------------------------------------------------------------------+
```

**Step 3**

- `configure` command automatically applies the data placement constraint and replication factor based on the available zones
- data placement constraint can be overridden using the `--data_placement_constraint` flag. Replication factor can be overridden using the `--rf` flag.

# Commands supported by yugabyted

```
$ ./bin/yugabyted -h
```

```
+-----------------------------------------------------------------------------------------------+
|                            Yugabyted CLI: YugabyteDB command line                             |
+-----------------------------------------------------------------------------------------------+
YugabyteDB command-line interface for creating and configuring YugabyteDB cluster.

Usage: yugabyted [command] [flags]

To start YugabyteDB cluster, run 'yugabyted start'.

Find more information at: https://docs.yugabyte.com/preview/reference/configuration/yugabyted/

Commands:
  start         Start YugabyteDB cluster.
  stop          Stop running YugabyteDB cluster.
  destroy       Destroy YugabyteDB cluster and remove data.
  status        Print status of YugabyteDB cluster.
  version       Release version of YugabyteDB cluster.
  collect_logs  Collect and package logs for troubleshooting.
  configure     Configure multi-zone/multiregion cluster
  connect       Connect to YugabyteDB cluster through the CLI.
  demo          Load and interact with preset demo data.

Flags:
  -h, --help    show this help message and exit

Run 'yugabyted [command] -h' for help with specific commands.
```

# Deploying secure YugabyteDB Cluster using yugabyted

# Starting a secure YugabyteDB instance

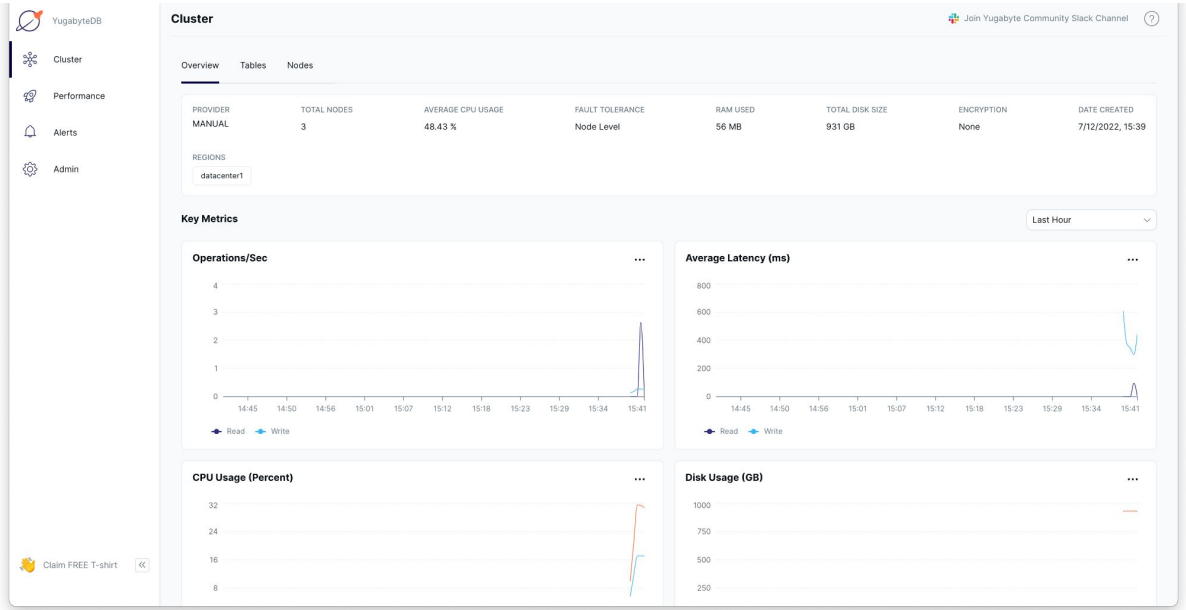
```
$ ./bin/yugabyted start --secure
```

```
Starting yugabyted...
✅ System checks
✅ YugabyteDB Started
✅ SSL Authentication enabled.
✅ User Authentication enabled.

+-----------------------------------------------------------------------------------------------------+
|                                             yugabyted                                               |
+-----------------------------------------------------------------------------------------------------+
| Status              : Running.                                                                      |
| Replication Factor  : 1                                                                             |
| Web console         : http://10.150.0.157:7000                                                      |
| JDBC                : jdbc:postgresql://10.150.0.157:5433/yugabyte?user=yugabyte&password=b4dBRMvCQSmx |
| YSQL                : bin/ysqlsh -h 10.150.0.157  -U yugabyte -d yugabyte                           |
| YCQL                : bin/ycqlsh 10.150.0.157 9042 -u cassandra --ssl                               |
| Data Dir            : /home/gargsans/var/data                                                       |
| Log Dir             : /home/gargsans/var/logs                                                       |
| Universe UUID       : 45d85a05-f54b-43a6-82b5-f154fef78885                                          |
+-----------------------------------------------------------------------------------------------------+
🚀 YugabyteDB started successfully! To load a sample dataset, try 'yugabyted demo'.
🎉 Join us on Slack at https://www.yugabyte.com/slack
👕 Claim your free t-shirt at https://www.yugabyte.com/community-rewards/

Credentials File is stored at /home/gargsans/var/data/yugabyted_credentials.txt
```

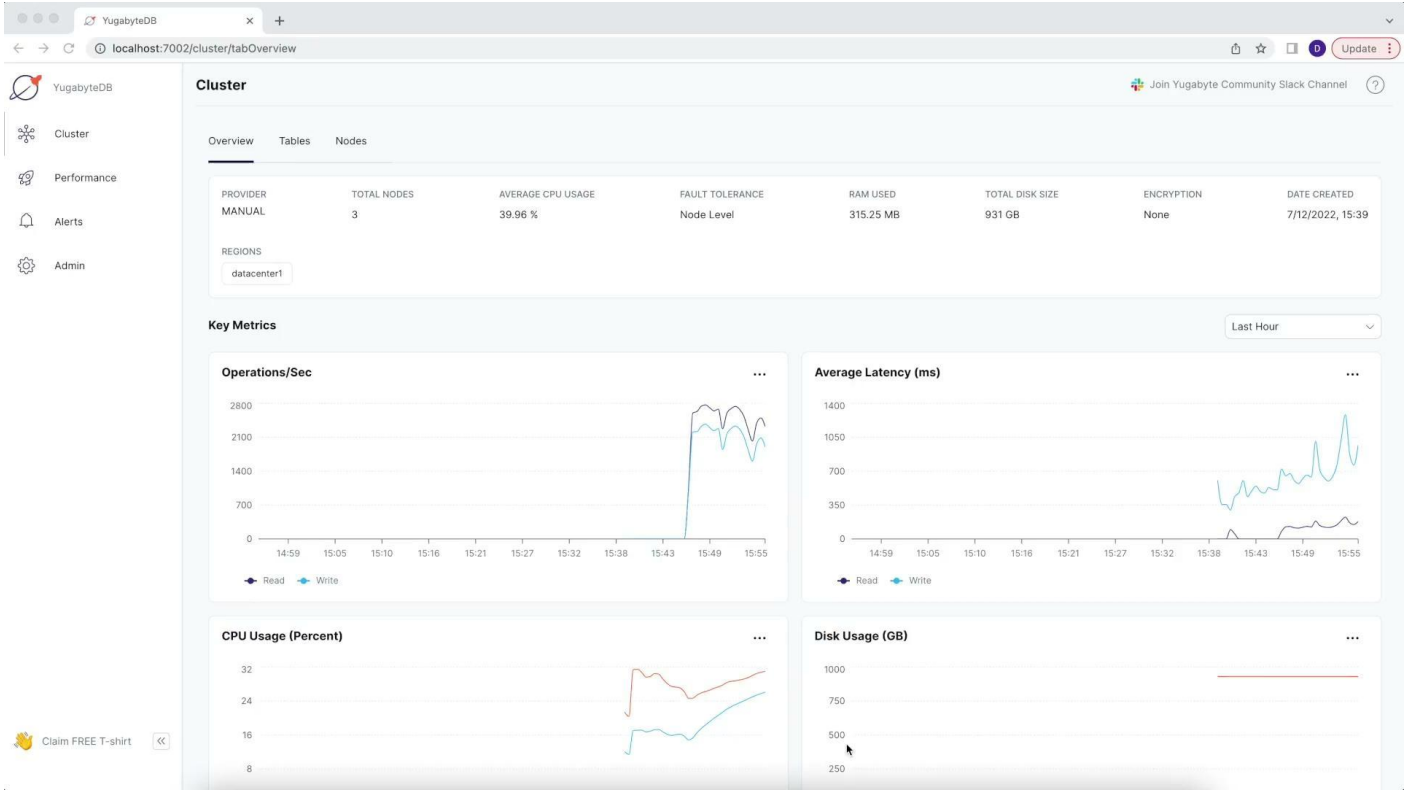
# yugabyted ui

# Managing YugabyteDB using yugabyted ui

```
$ ./yugabyted start --ui=true
```



# Demo - yugabyted ui



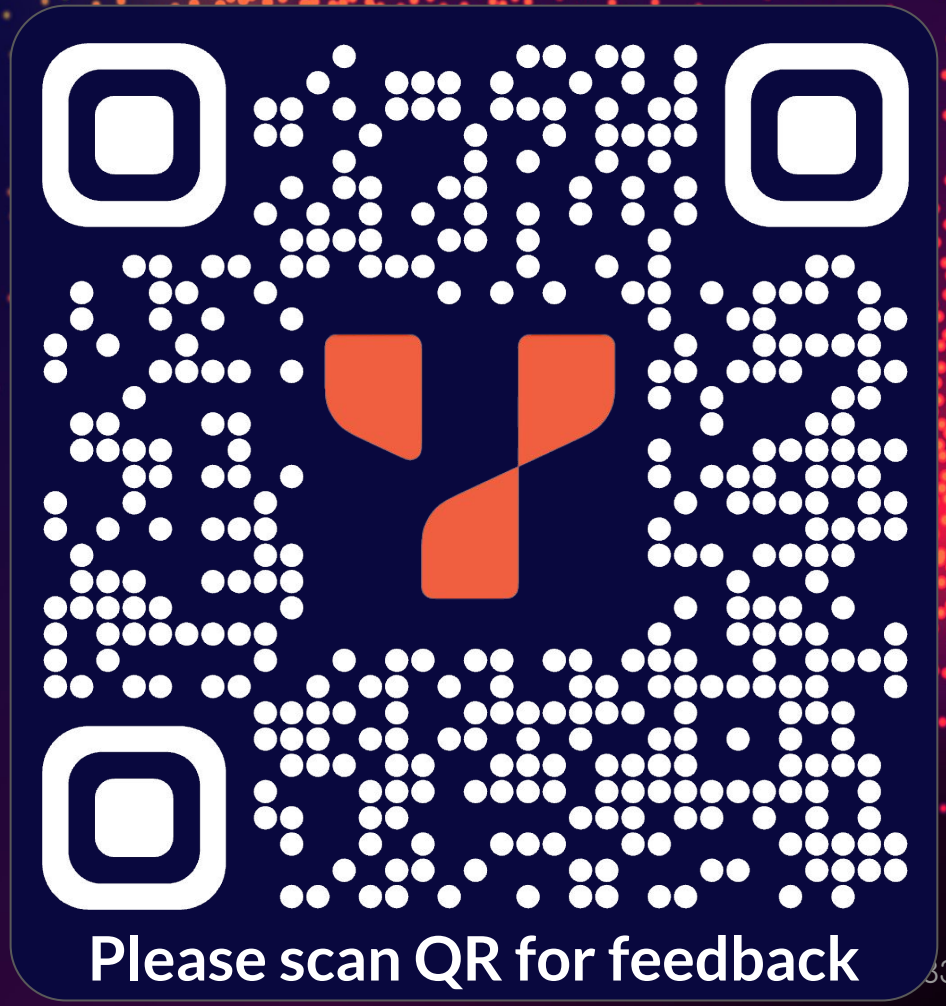
# Other Tools

## yb-admin, yb-ts-cli, ....

# Thank You

**Join us on Slack:**
www.yugabyte.com/slack

**Star us on GitHub:**
github.com/yugabyte/yugabyte-db




yugabyteDB