

ORACLE®

Why MySQL High Availability Matters

Matt Lord
MySQL Product Manager
@mattalord

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Program Agenda

- 1 ➤ High Availability Explained
- 2 ➤ Native MySQL High Availability
- 3 ➤ Highly Available MySQL Reference Architectures
- 4 ➤ What's Coming Next for MySQL HA and Scale?

Program Agenda

- 1 High Availability Explained
- 2 Native MySQL High Availability
- 3 Highly Available MySQL Reference Architectures
- 4 What's Coming Next for MySQL HA and Scale?

High Availability: Terms and Concepts

- Availability
 - Outage and downtime
 - Mean Time Between Failures (MTBF)
 - Mean Time To Recover (MTTR)
 - Service Level Agreement (SLA)
 - The Nine's Scale : 5 nines = 5 mins downtime per year
- Capacity / Performance
- Redundancy
- Fault Tolerance
- Disaster Tolerance/Recovery



High Availability: Considerations

- SLA requirements
 - What is needed to support your business objectives
- Operational capabilities
 - Will you have the workforce to implement and maintain it
- Service agility
 - Can it grow and change as your company does
- Time to market
 - Can you go to market quickly
- Budgetary constraints
 - Cost of downtime versus cost of HA implementation



High Availability: Factors

- Environment
 - Redundant servers in different datacenters and geographical areas will protect you against regional issues—power grid failures, hurricanes, earthquakes, etc.
- Hardware
 - Each part of your hardware stack—networking, storage, servers—should be redundant
- Software
 - Every layer of the software stack needs to be duplicated and distributed across separate hardware and environments
- Data
 - Data loss and inconsistency/corruption must be prevented by having multiple copies of each piece of data, with consistency checks and guarantees for each change

High Availability: Design Factors



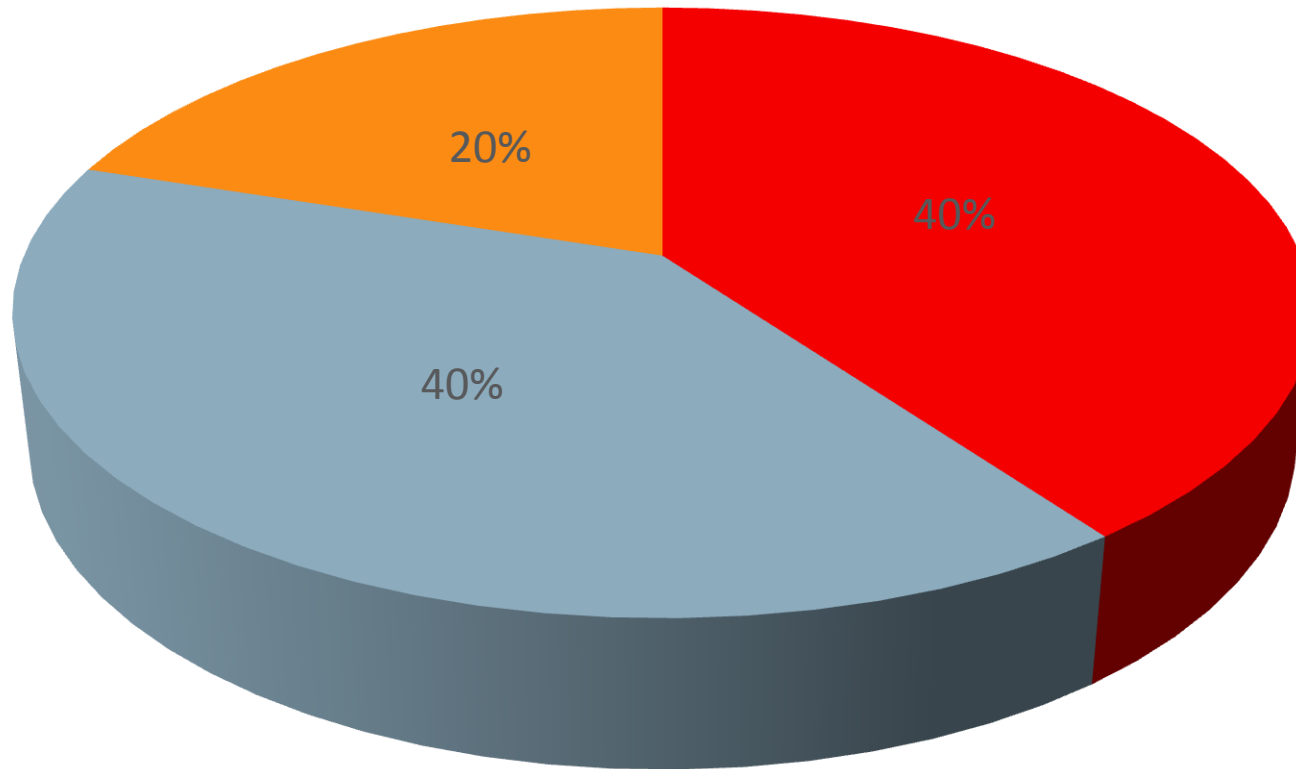
- Reliability, Availability, and Serviceability (RAS)
 - Reliability
 - Quality hardware and software components that you can manage and trust
 - Availability
 - Multiple copies of each datum replicated to separate hardware
 - Multiple copies of each piece of the hardware and software stack
 - Multiple physical locations
 - Separate power grids
 - Enough physical distance to survive disasters such as floods, earthquakes, and power grid outages
 - Resiliency
 - Automation to transparently deal with any potential failure in the infrastructure, hardware, software
 - Serviceability
 - Ensure that you can monitor, diagnose, maintain, and repair it all

High Availability: **Acceptable Performance**

- Performance outside of acceptable bounds is unavailability
 - Determine your acceptable application response times
 - Ensure that you can measure impact from each layer of stack
 - Determine failure versus load
 - Scale up/out or failover
- Build elasticity into each layer
 - Quickly and easily scale up and down to handle changing load and traffic patterns



High Availability: The Causes of Downtime



■ Software/Application ■ Human Error ■ Hardware

* Source: Gartner Group 1998 survey

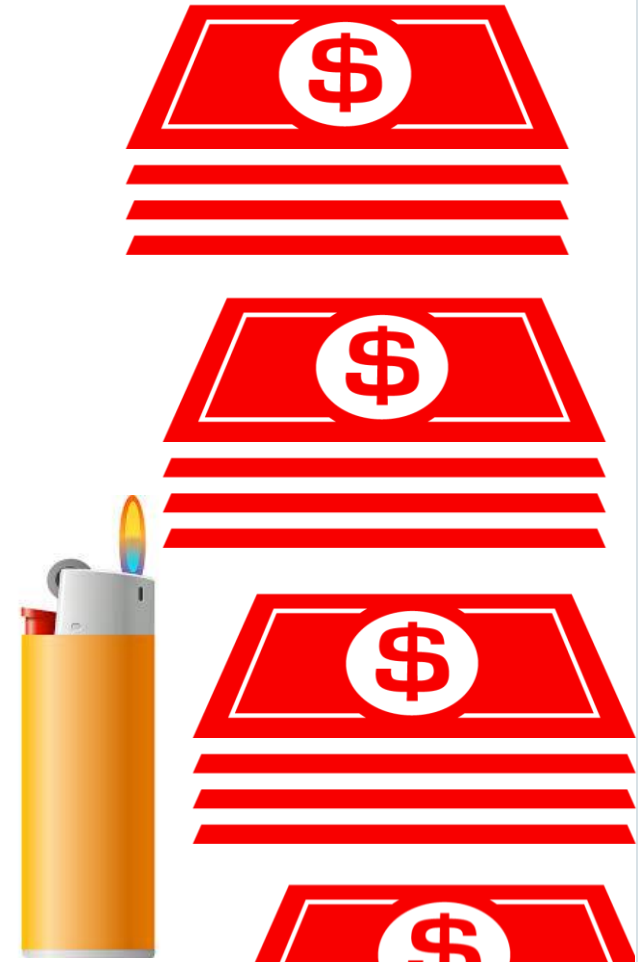
A study by the Gartner Group projected that through 2015, 80% of downtime will be due to people and process issues

High Availability: The Business Cost of Downtime

- Calculate a cost per minute of downtime
 - Average revenue generated per-minute over a year
 - Cost of not meeting any customer SLAs
 - Factor in costs that are harder to quantify

1. Revenue
2. Reputation
3. Customer sentiment
4. Stock price
5. Service's success
6. Company's very existence

THIS is why
HA matters!



High Availability: Database Needs

- Management infrastructure
 - Monitoring of status, health, performance
 - Facilities for service changes, service transitions
- Failure detection and handling
 - Identify and handle failures
- Elasticity
 - Scale up to ensure acceptable performance is always maintained
- Consistency guarantees
 - Conflict detection and handling ; data loss protection/prevention
- Online maintenance



High Availability: Does The Application Require It?

- Recovery Time Objective
 - Maximum length of downtime before there is break in “business continuity”
- Recovery Point Objective
 - Point in time to which data must be recovered when service is re-established

Tier 1

- Mission-critical services

Tier 2

- Business-critical services

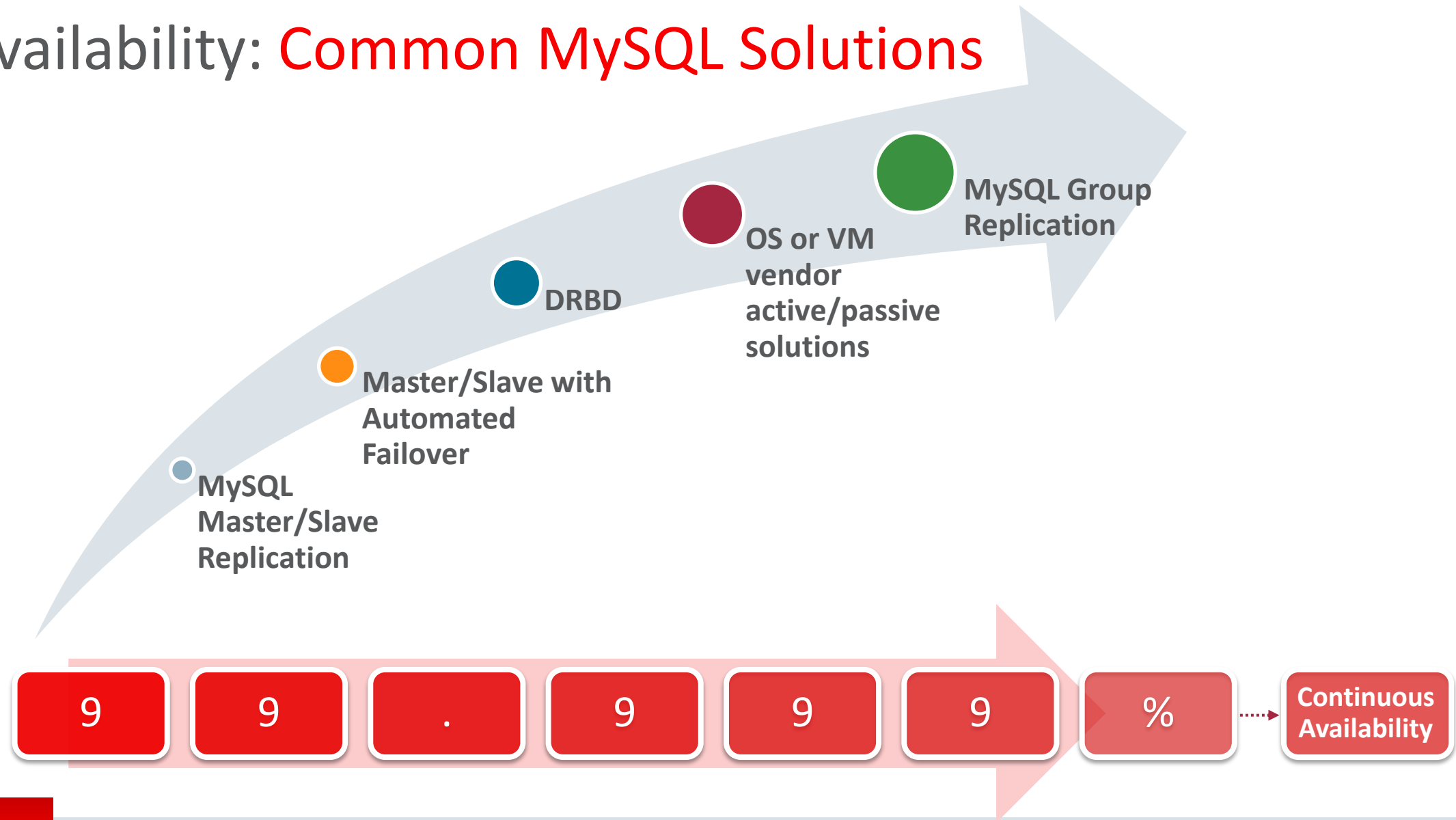
Tier 3

- Task-critical services

Tier 4

- Non-critical services

High Availability: Common MySQL Solutions

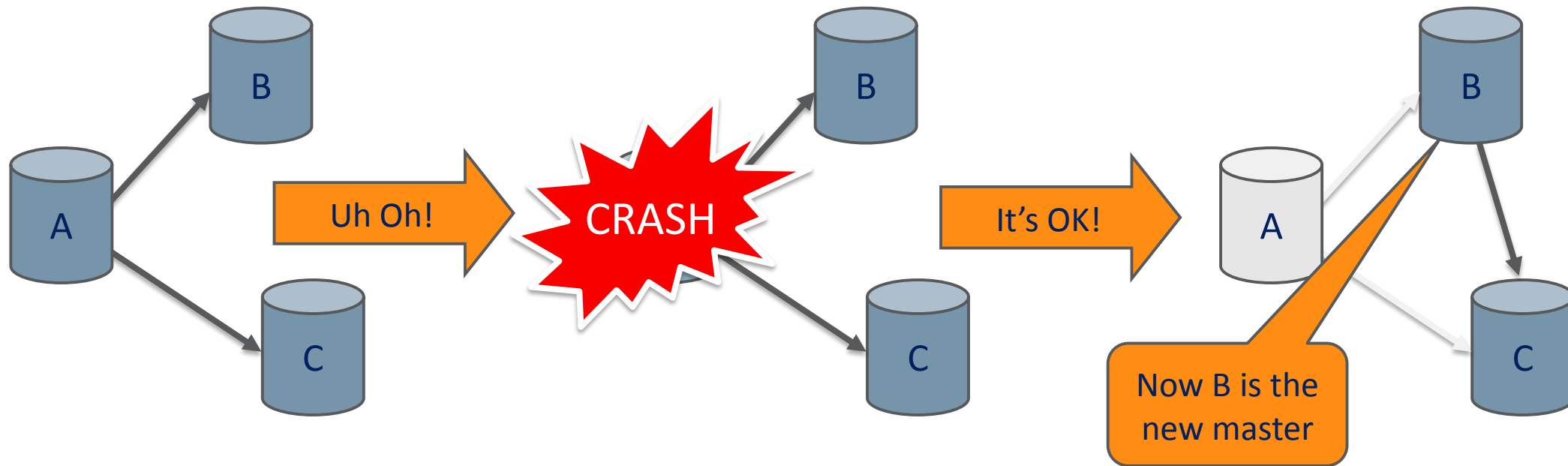


Program Agenda

- 1 High Availability Explained
- 2 Native MySQL High Availability**
- 3 Highly Available MySQL Reference Architectures
- 4 What's Coming Next for MySQL HA and Scale?

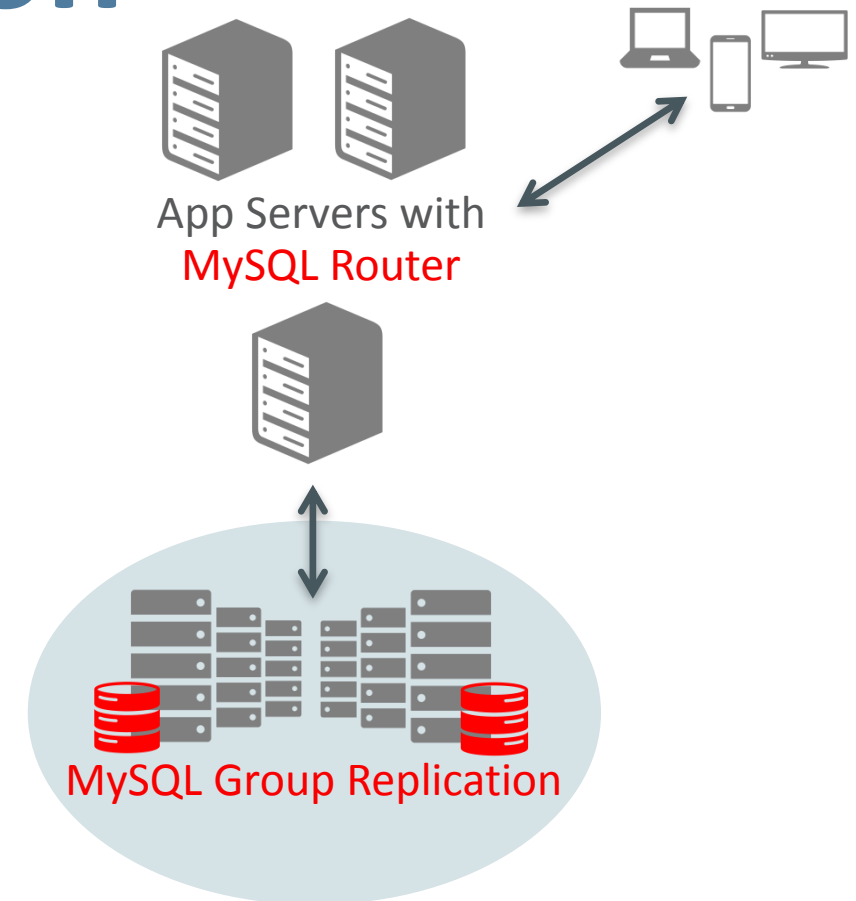
What is Replication Used For?

A major building block for **high availability**



MySQL Group Replication

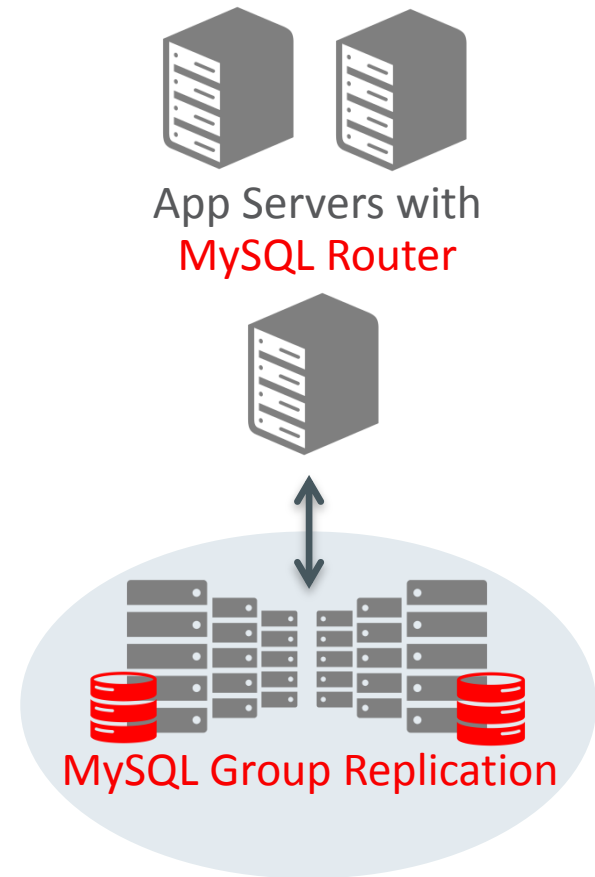
“High Availability becomes a core first class feature of MySQL!”



MySQL Group Replication: **What Is It?**

- Group Replication library
 - Implementation of Replicated Database State Machine theory
 - MySQL GCS is based on Paxos (variant of Mencius)
 - Provides *virtually* synchronous replication for MySQL 5.7+
 - Supported on ***all MySQL platforms***
 - Linux, Windows, Solaris, OSX, FreeBSD

*“Multi-master **update anywhere** replication plugin for MySQL with built-in **conflict detection and resolution**, **automatic distributed recovery**, and **group membership**.”*



MySQL Group Replication: What Does It Provide?

- A highly available distributed MySQL database service
 - Clustering eliminates single points of failure (SPOF)
 - Allows for online maintenance
 - Removes the need for manually handling server fail-over
 - Provides distributed fault tolerance and self-healing
 - Enables Active/Active update anywhere setups
 - Automates reconfiguration (adding/removing nodes, crashes, failures)
 - Makes it easy to scale up/down based on demand
 - Automatically ensures data consistency
 - Detects and handles conflicts
 - Prevents data loss
 - Prevents data corruption



MySQL Group Replication: Use Cases

- **Elastic Replication**

- Environments that require a very fluid replication infrastructure, where the number of servers has to grow or shrink dynamically and with little pain as possible.

- **Highly Available Shards**

- Sharding is a popular approach to achieve write scale-out. Users can use MySQL Group Replication to implement highly available shards in a federated system. Each shard can map into a Replication Group.

- **Alternative to Master-Slave Replication**

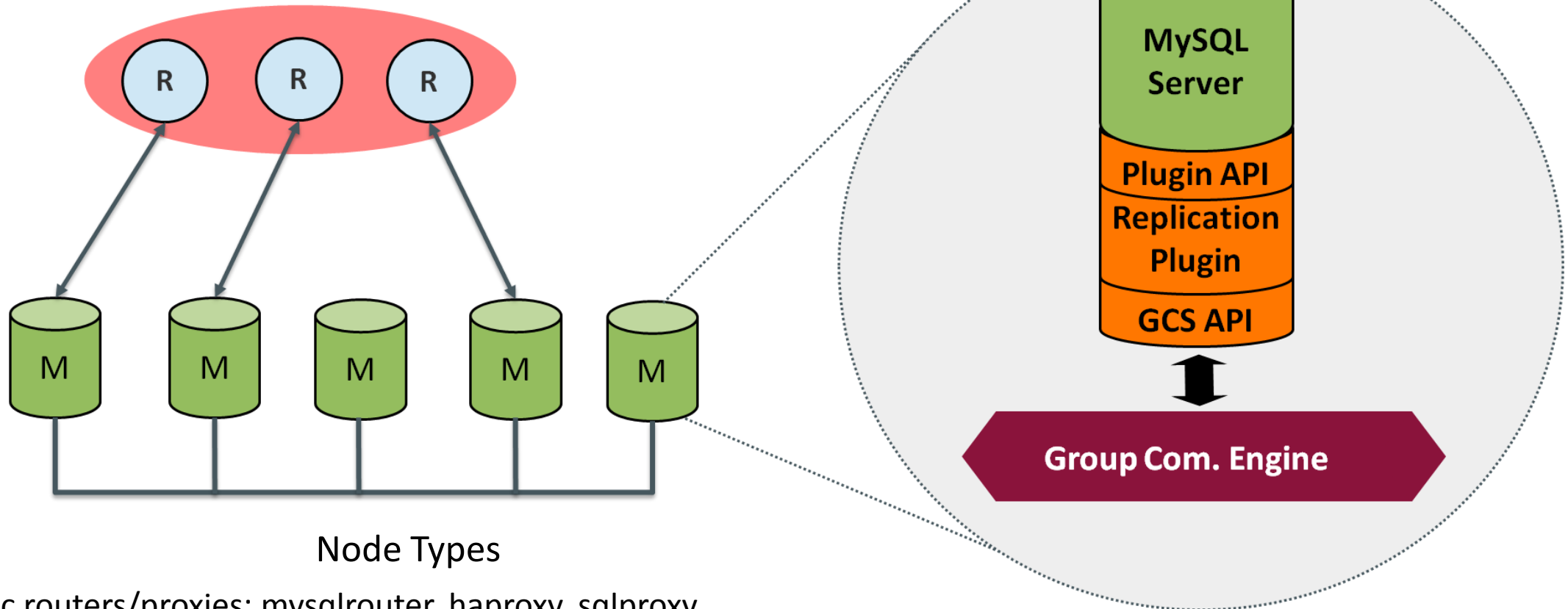
- It may be that a single master server makes it a single point of contention. Writing to an entire group may prove more scalable under certain circumstances.

MySQL Group Replication: What Sets It Apart?

- Built by the MySQL Engineering Team
 - Natively integrated into Server: InnoDB, Replication, GTIDs, Performance Schema, SYS
 - Built-in, no need for separate downloads
 - Available on all platforms [Linux, Windows, Solaris, FreeBSD, etc]
- Better performance than similar offerings
 - MySQL GCS has [optimized network protocol](#) that reduces the impact on latency
- Easier monitoring
 - Simple Performance Schema tables for group and node status/stats
 - Native support for Group Replication coming to MySQL Enterprise Monitor
- Modern full stack MySQL HA being built around it
 - Native end-to-end easy to use sharded InnoDB clusters



MySQL Group Replication: **Architecture**

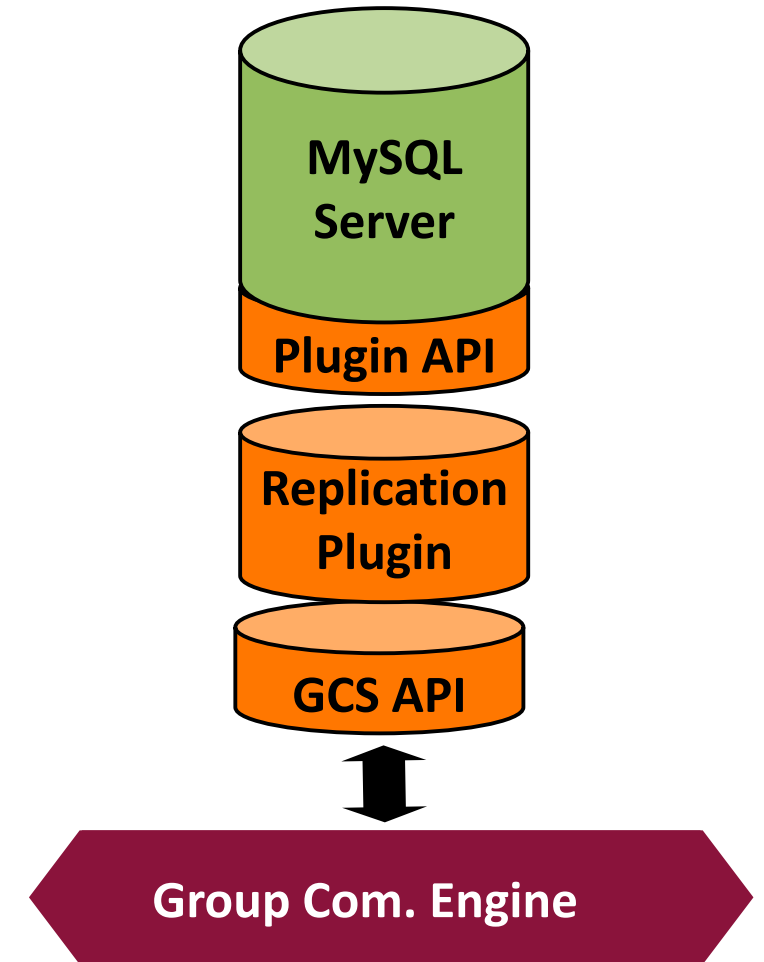


R: Traffic routers/proxies: mysqlrouter, haproxy, sqlproxy, ...

M: mysqld nodes participating in Group Replication

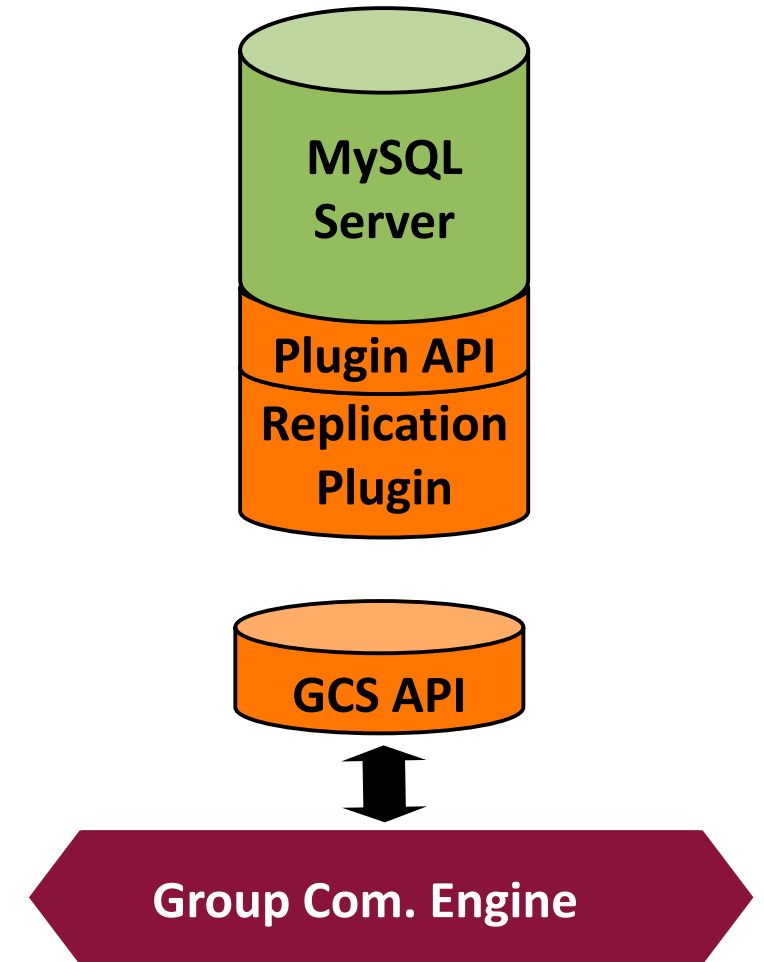
MySQL Group Replication: **Stack**

- The Group Replication plugin is responsible for
 - Maintaining distributed execution context
 - Detecting and handling conflicts
 - Handling distributed recovery
 - Detect membership changes
 - Donate state if needed
 - Collect state if needed
 - Proposing transactions to other members
 - Receiving and handling transactions from other members
 - Deciding the ultimate fate of transactions
 - commit or rollback



MySQL Group Replication: **Stack**

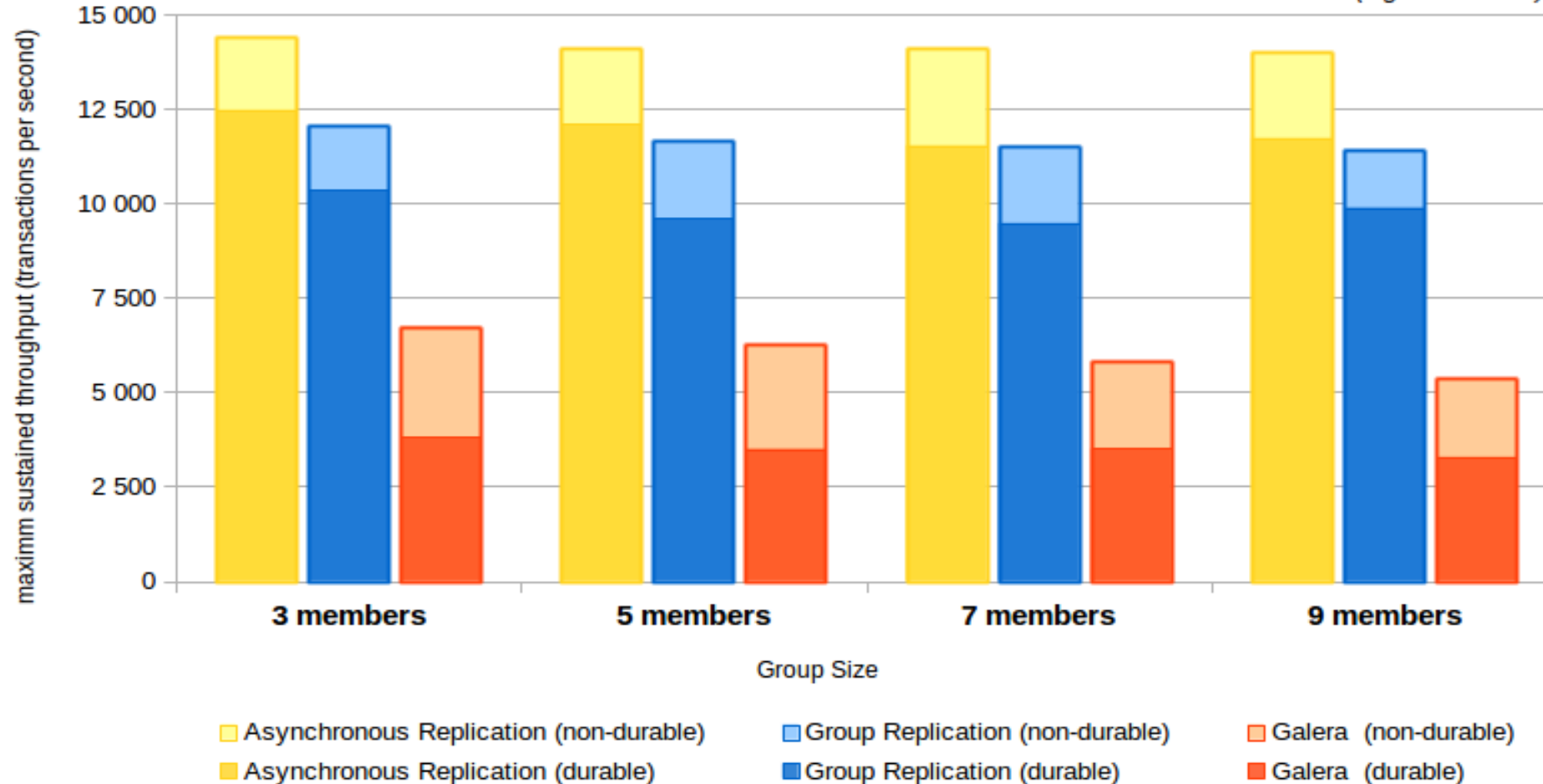
- The Group Communication System API:
 - Abstracts the underlying group communication system implementation from the plugin itself
 - Maps the interface to a specific group communication system implementation
- The Group Communication System engine:
 - Variant of Paxos developed at MySQL
 - Building block to provide distributed agreement between servers



MySQL Group Replication: Performance Comparison

Single-master Maximum Throughput: Sysbench OLTP RW

(higher is better)

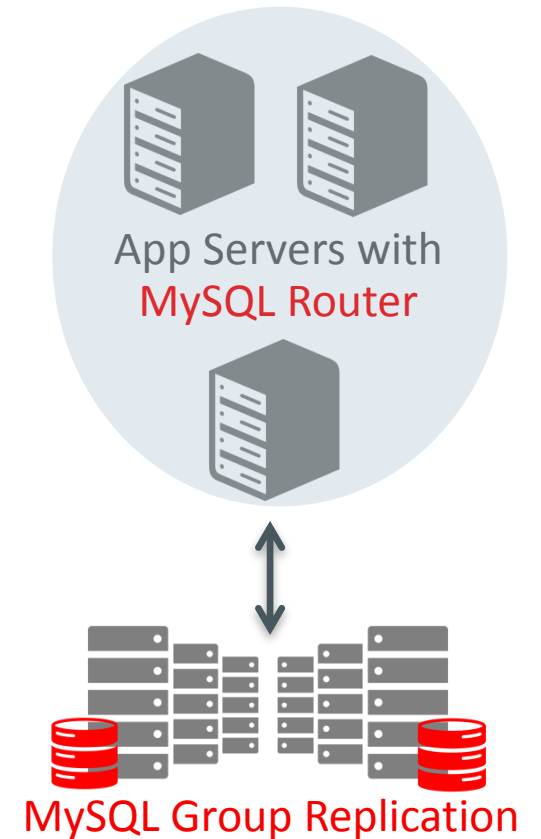


MySQL Router

Transparent Access to HA Databases for MySQL Applications

- Transparent client connection routing
 - Load balancing
 - Application connection failover
- Stateless design offers easy HA client routing
 - A local Router becomes part of the application stack

“MySQL Router allows you to easily migrate your standalone MySQL instances to natively distributed and highly available Group Replication clusters without affecting existing applications!”



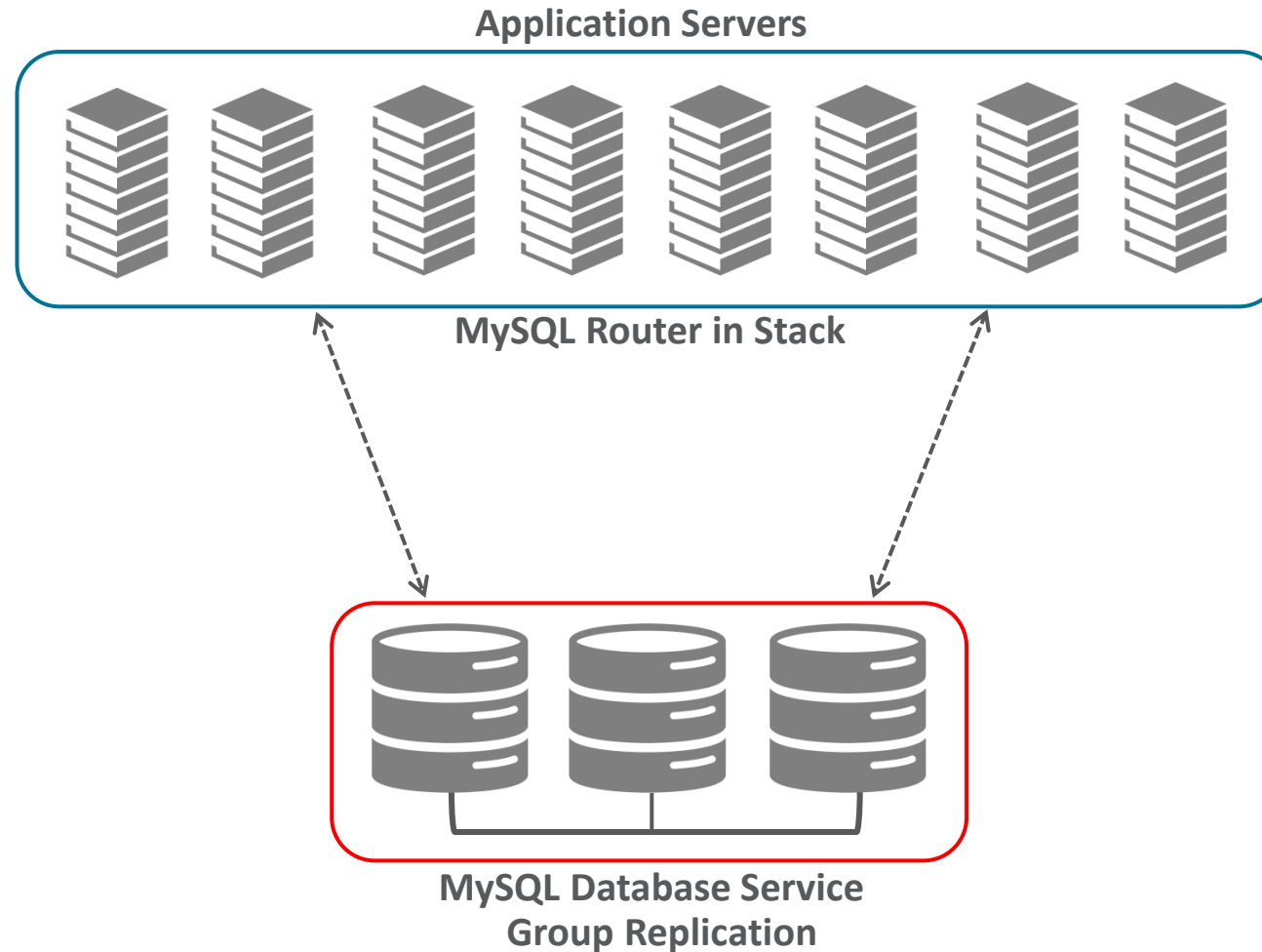
Program Agenda

- 1 High Availability Explained
- 2 Native MySQL High Availability
- 3 Highly Available MySQL Reference Architectures**
- 4 What's Coming Next for MySQL HA and Scale?

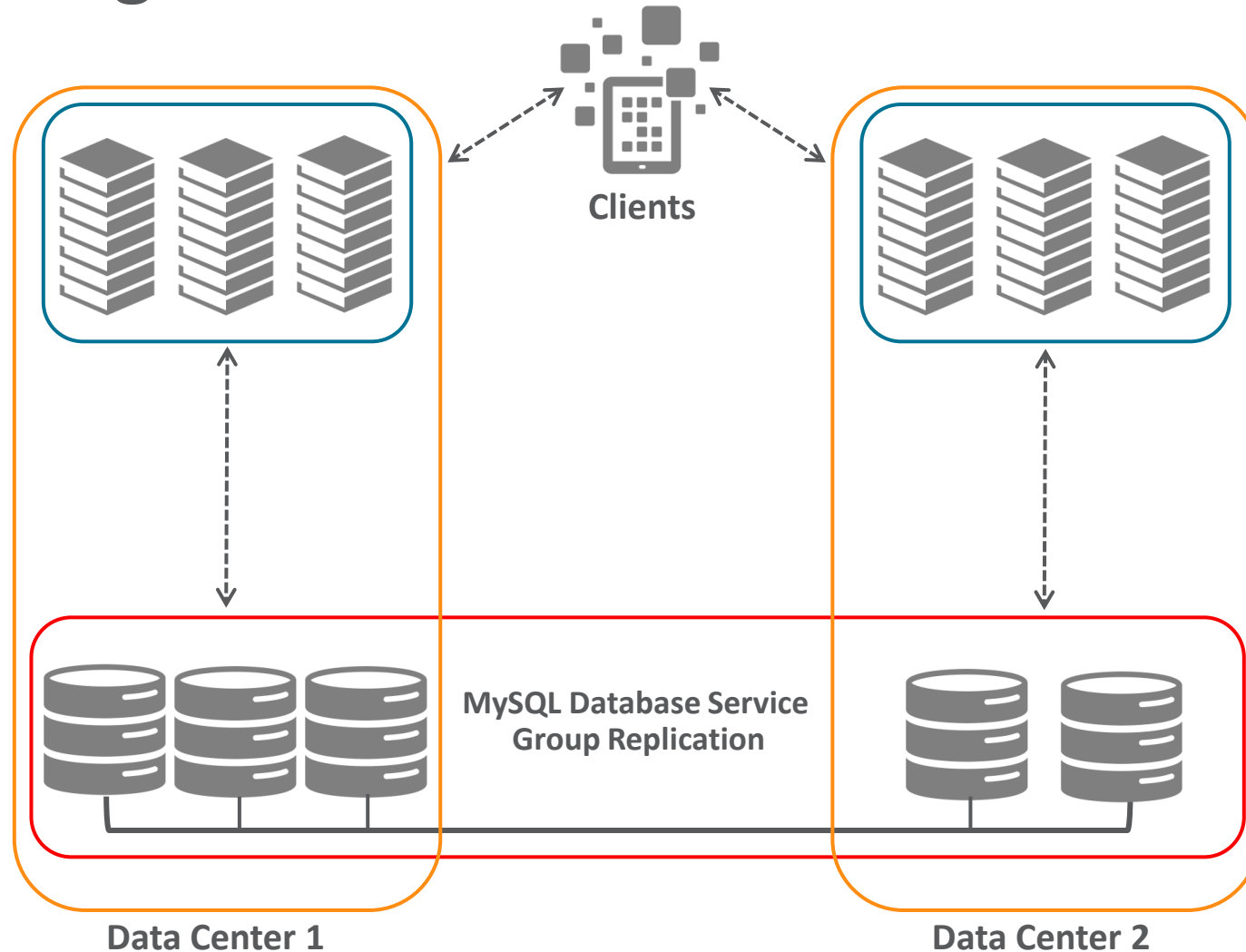
Hardware and Infrastructure Notes

- 3, 5, or 7 machines per group
 - Isolate machine resources as much as possible
 - Limit virtualization layers
 - Machines configured for dedicated database server role
 - Recommended configuration
 - 32-64 vCPUs with fast CPU clock (2.5GHz+)
 - SSDs (for data and replication logs)
 - High quality network connection between each machine
 - Low latency, high throughput, reliable
 - Limit routers and hubs as much as possible
 - Isolated and dedicated network when possible

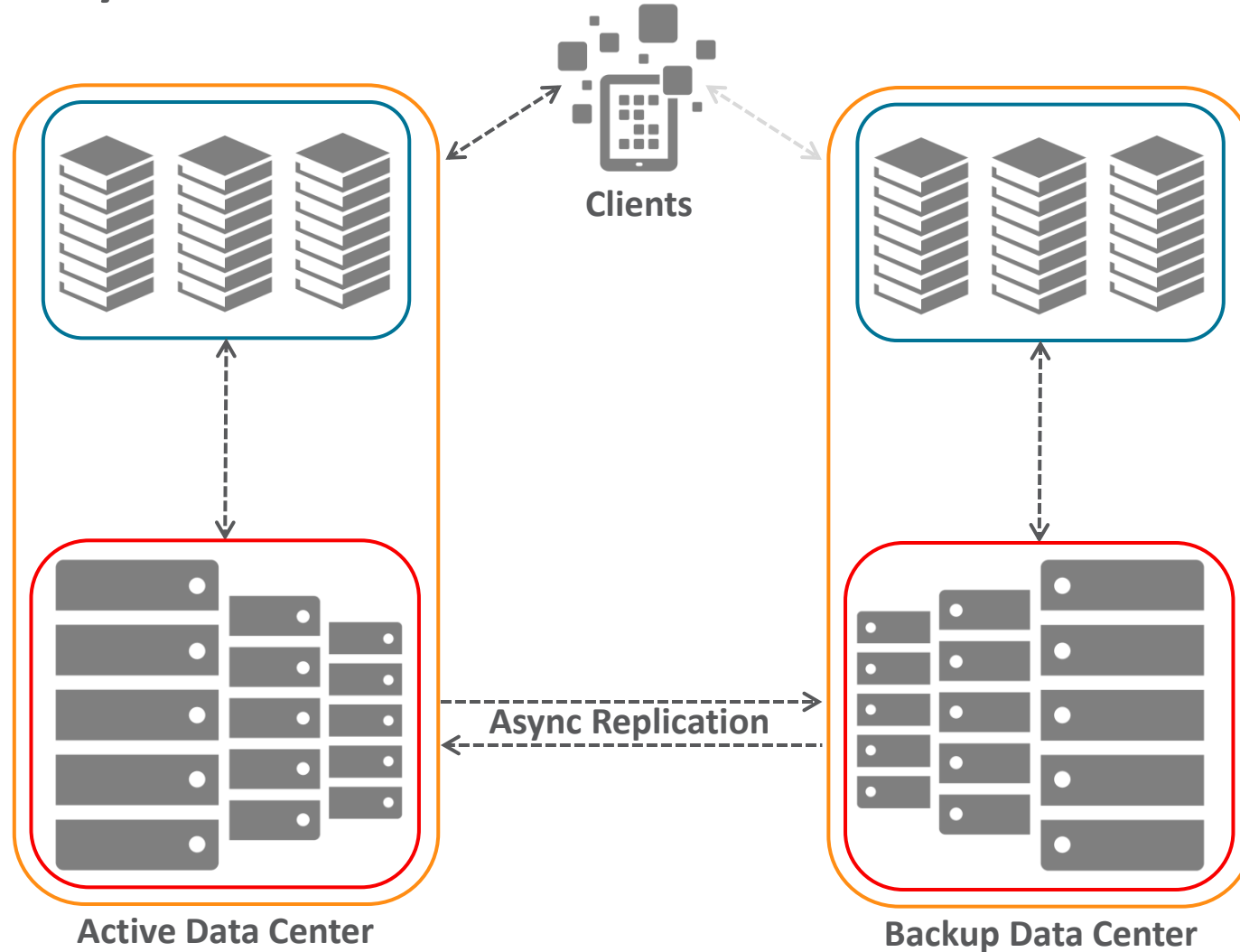
Shared Nothing Cluster – Single Data Center



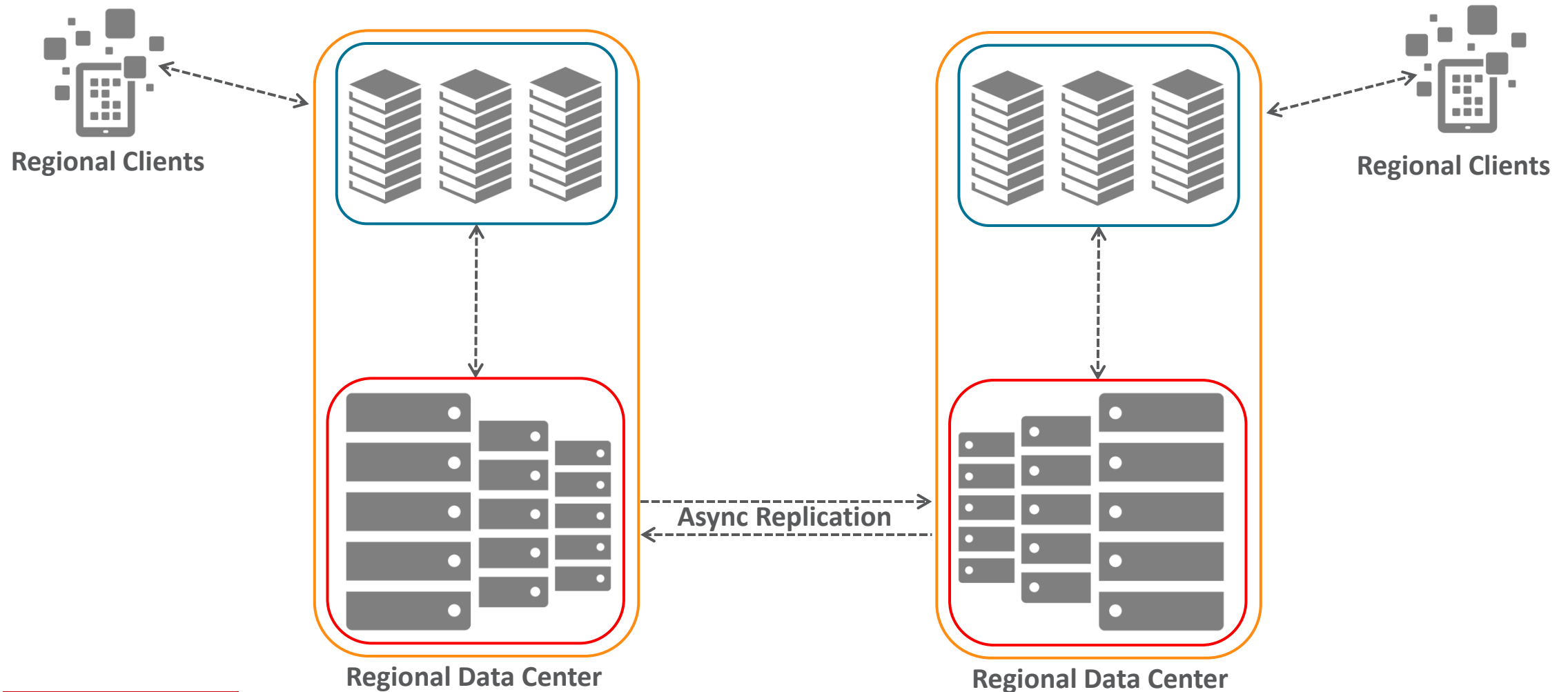
Shared Nothing Cluster – Cross Data Center



Geographically Redundant Cluster

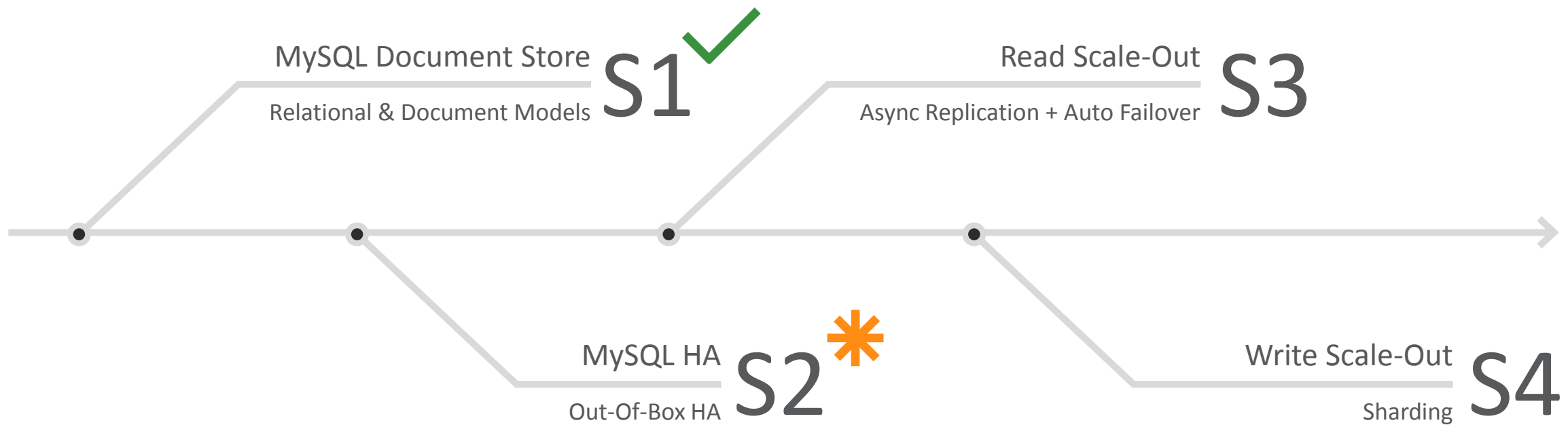


Active/Active Multi-Data Center Setup



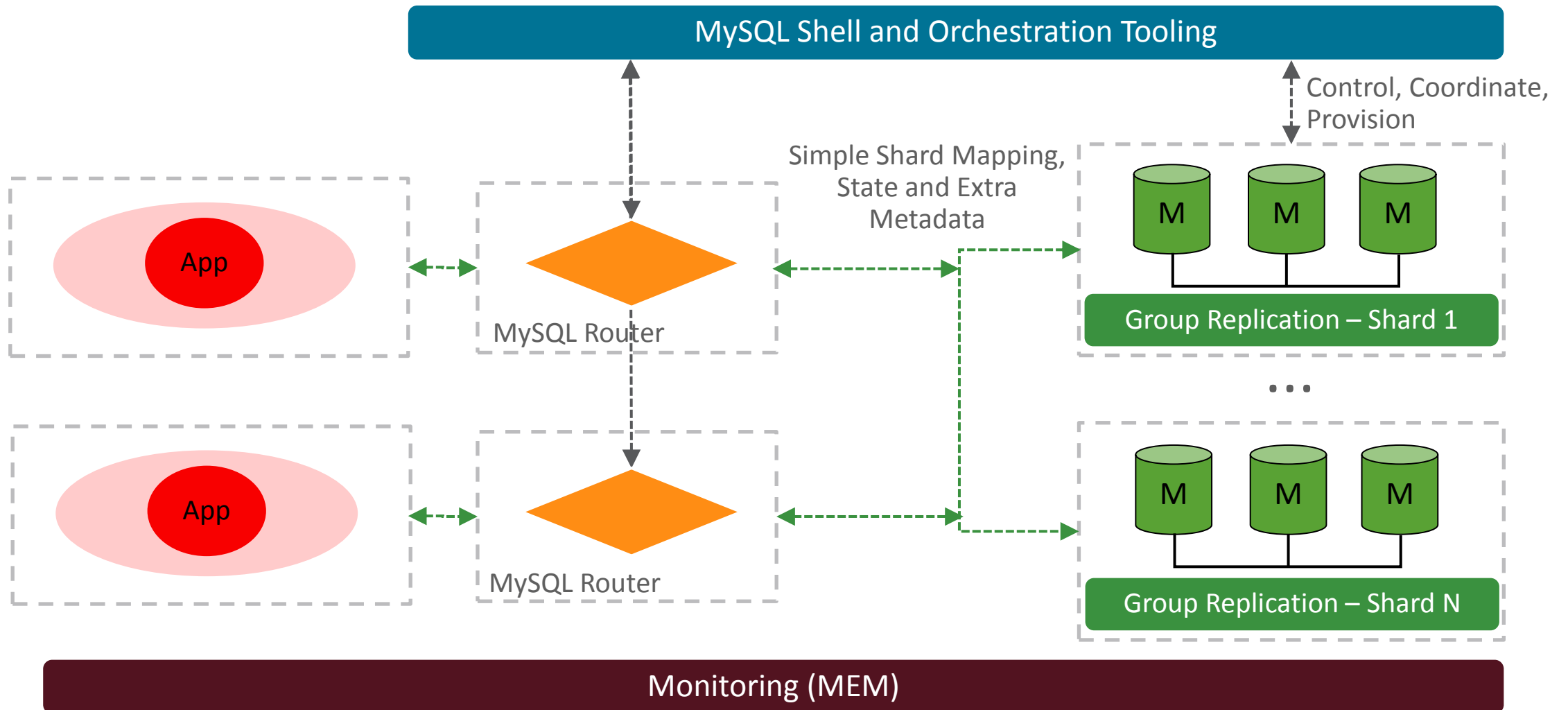
Program Agenda

- 1 ➤ High Availability Explained
- 2 ➤ Native MySQL High Availability
- 3 ➤ Highly Available MySQL Reference Architectures
- 4 ➤ What's Coming Next for MySQL HA and Scale?



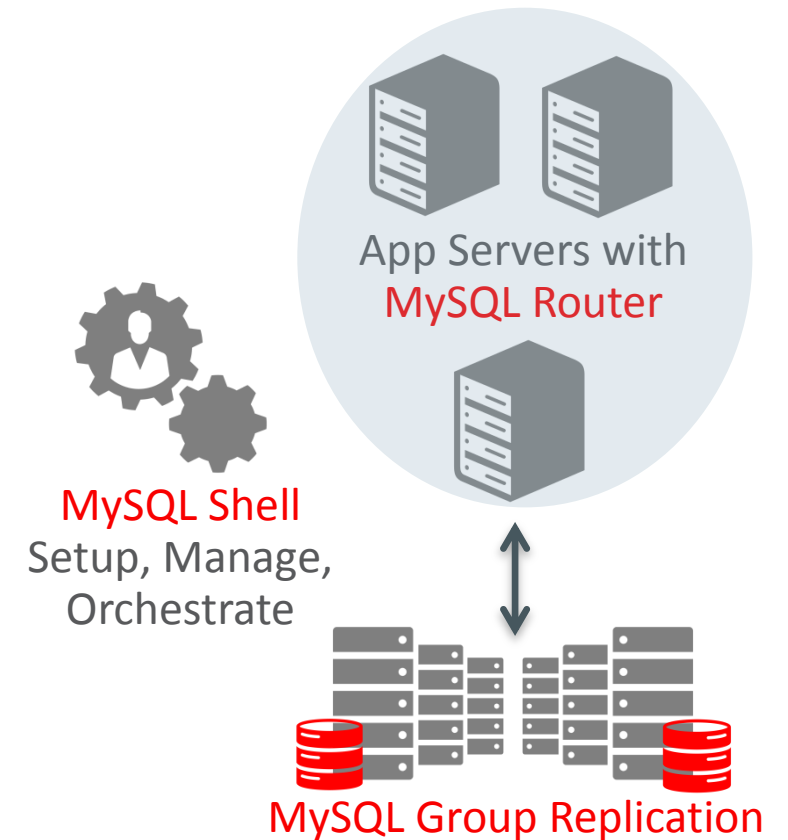
* Download InnoDB Cluster Preview Release from labs.mysql.com

MySQL InnoDB Cluster: The Big Picture



MySQL Router: 2.1

- Native support for InnoDB clusters
 - Understands Group Replication (GR) topology
 - Utilizes metadata schema stored on each member
 - Bootstraps itself and sets up client routing for the GR cluster
 - Allows for intelligent client routing into the GR cluster
 - Supports multi-master and single primary modes
- Core improvements
 - Logging
 - Monitoring
 - Performance
 - Security

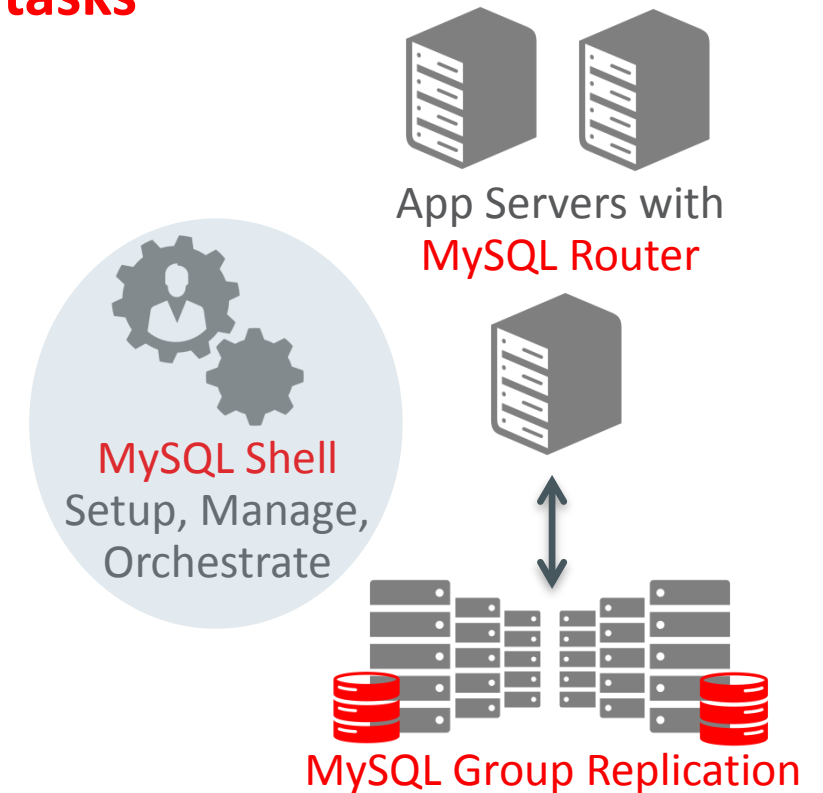


MySQL Shell

A single unified client for all administrative and operations tasks

- Multi-Language: JavaScript, Python, and SQL
 - Naturally scriptable
- Supports both Document and Relational models
- Exposes full Development and Admin API

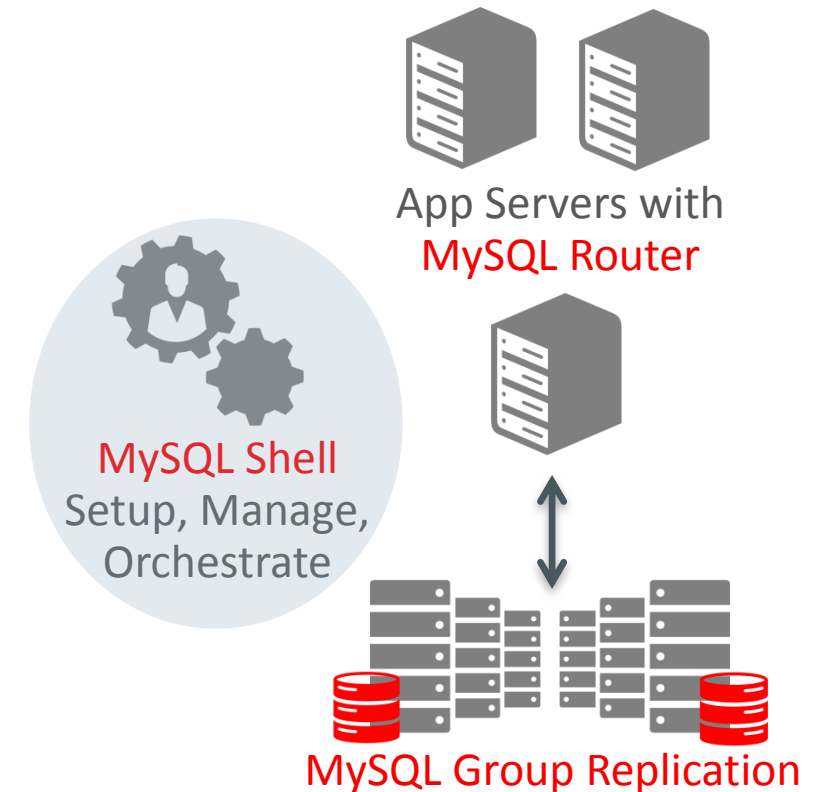
"MySQL Shell provides the developer and DBA with a single intuitive, flexible, and powerfull interface for all MySQL related tasks!"



MySQL Shell: Admin API

Database Administration Interface

- `mysql-js> dba.help()`
- The global variable 'dba' is used to access the MySQL AdminAPI
- Perform DBA operations
 - Manage MySQL InnoDB clusters
 - Create clusters
 - Get cluster info
 - Start/Stop MySQL Instances
 - Validate MySQL instances
 - ...



MySQL Enterprise Monitor

- Native holistic support for Group Replication clusters
 - Intelligent monitoring and alerting
 - Topology views
 - Detailed metrics and graphs
 - Best Practice advice
- Monitoring of MySQL Router instances

Sharded InnoDB Clusters

- Group Replication
 - Each shard is a highly available replica set
- MySQL Router
 - Manages shard mappings and related metadata
 - Manages client routing
 - Provides cross shard execution framework
- MySQL Shell
 - Exposes management and orchestration features

Oracle Cloud: **MySQL Cloud Service**

- MySQL Enterprise Edition
- Web based console to manage your MySQL Cloud instances
- Self-Service Provisioning
- Elastic Scalability
- Multi-Layered Security
- Unified Cloud Management
- Oracle PaaS and IaaS Integration
- Premier Technical Support included



**Integrated HA and DR
service options – utilizing
Group Replication –
coming soon!**

Upcoming Webinar

Always On, Without Fail: MySQL Group Replication

Luis Soares, MySQL Replication Team Lead, Oracle

Tuesday, January 17, 2017 at 09:00 AM Pacific Standard Time

[Register now!](#)

ORACLE®