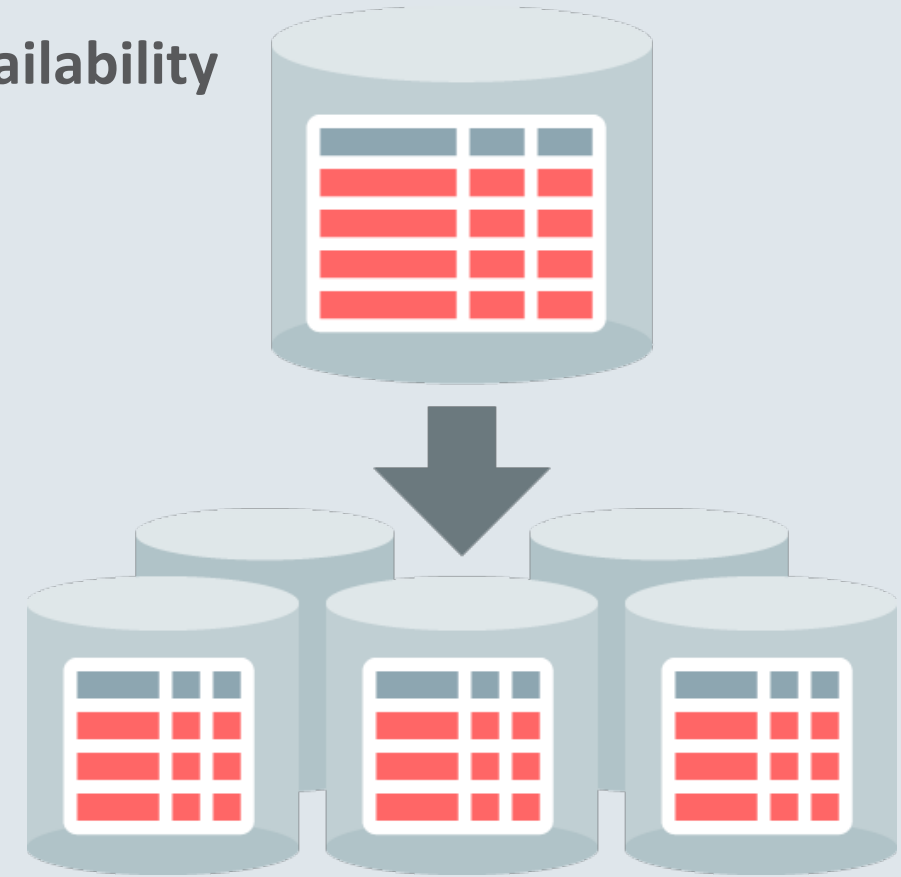


# Oracle Sharding – Session I

**Linear Scalability, Geo-distribution and Extreme Data Availability  
for Web-scale Applications**

Nagesh Battula  @NageshBattula  
Oracle Sharding Product Management

Mark Dilman, Director  
Oracle Sharding Product Development



# 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 (Session 1)

- 1 ➤ Sharding Overview
- 2 ➤ Sharded Database and Schema Creation
- 3 ➤ Data Dependent Routing
- 4 ➤ MAA Benchmark Results - Sharding on Bare Metal Cloud
- 5 ➤ Customers' Perspective
- 6 ➤ Competitive

# Program Agenda (Session 2)

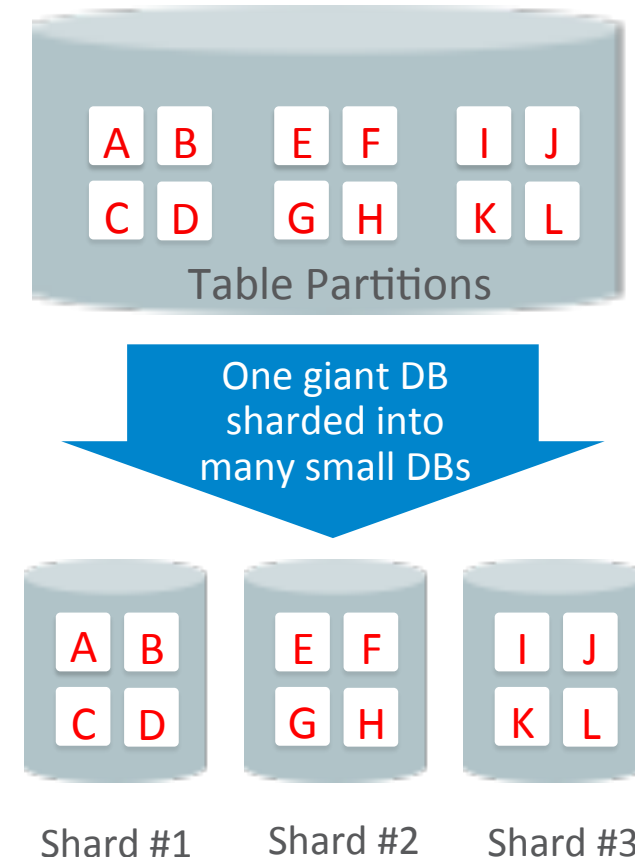
- 1 ➤ China Telecom – Sharding for WeChat IoT Application
- 2 ➤ Oracle Dyn's experience with Oracle Sharding
- 3 ➤ Summary

# Program Agenda (Session 1)

- 1 Sharding Overview
- 2 Sharded Database and Schema Creation
- 3 Data Dependent Routing
- 4 MAA Benchmark Results - Sharding on Bare Metal Cloud
- 5 Customers' Perspective
- 6 Competitive

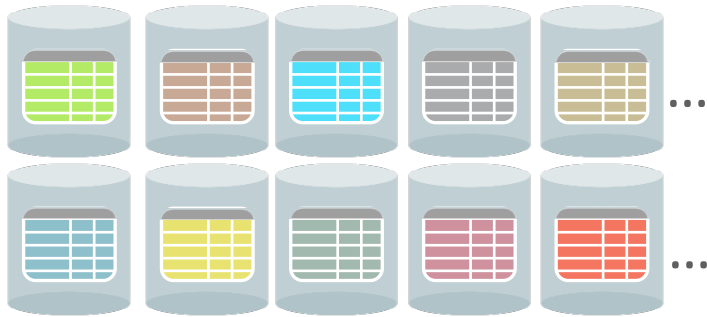
# Oracle Sharding – An Elastic Relational Database

- Horizontal partitioning of data across up to 1000 independent Oracle Databases (shards)
- Shared-nothing hardware architecture
  - Each shard runs on a separate server
  - No shared storage
  - No clusterware
- Data is partitioned using a sharding key (i.e. `account_id`)



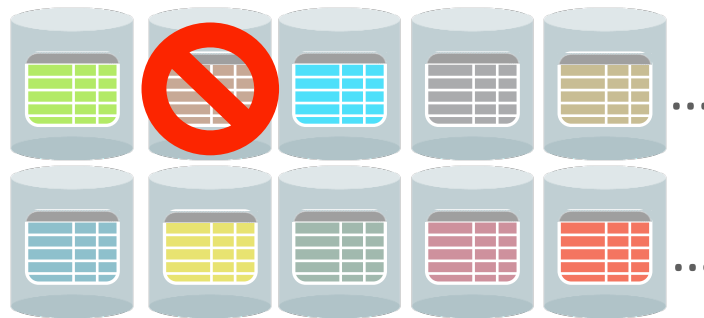
# Oracle Database Sharding – Benefits

## Linear Scalability



Add shards online to increase database size and throughput.  
Online split and rebalance.

## Extreme Availability



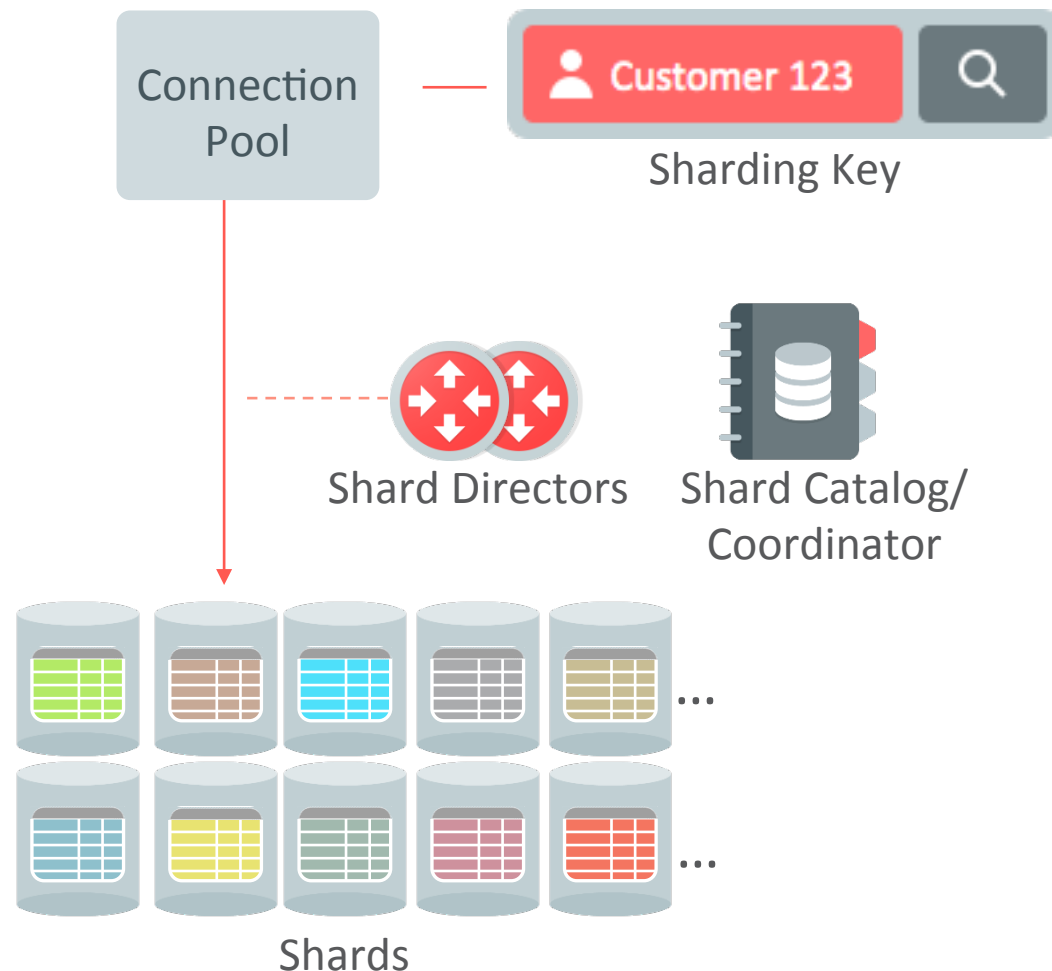
Shared-nothing hardware architecture. Fault of one shard has no impact on others.

## Geographic Distribution



User defined data placement for performance, availability, DR or to meet regulatory requirements.

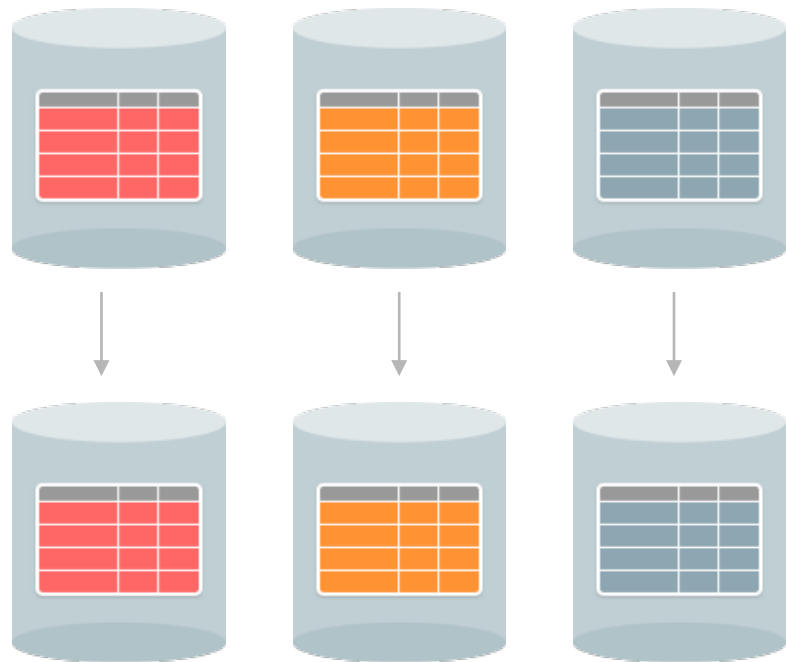
# Architecture & Key Features of Oracle Sharding



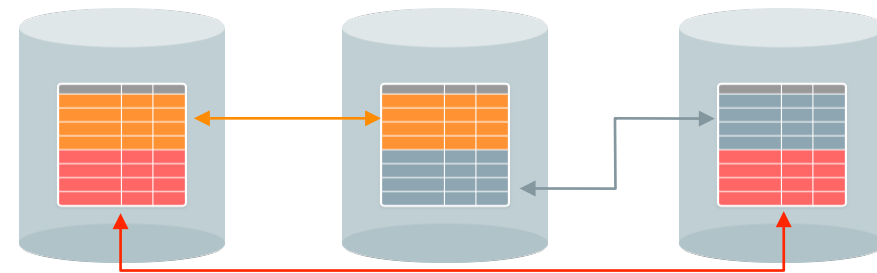
- Auto deployment of up to 1000 shards with replication
  - Declarative specification of topology
  - Supports Active Data Guard and Oracle GoldenGate
- Multiple sharding methods
  - Consistent Hash, **List**, **Range** or Composite
- Centralized schema maintenance
  - Native SQL for sharded and duplicated tables
- Direct routing and Proxy routing
- Online scale-out w/auto resharding or scale-in



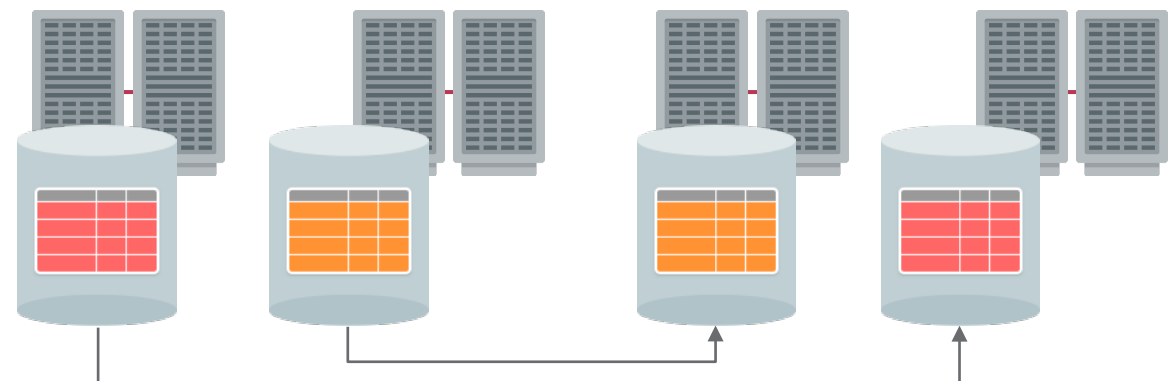
# Automated Data Availability Configuration



Active Data Guard with Fast-Start Failover



GoldenGate 'chunk-level' active-active replication  
with automatic conflict detection/resolution (OGG 12.3)

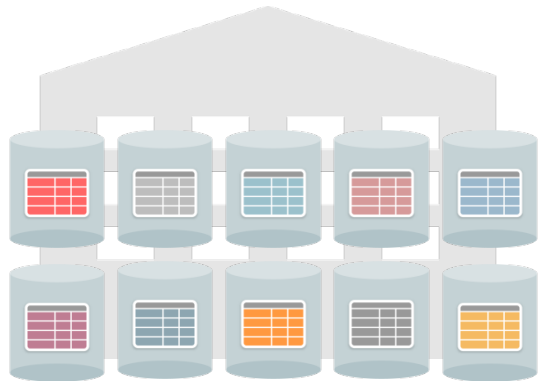


Optionally – complement replication with Oracle RAC for server HA

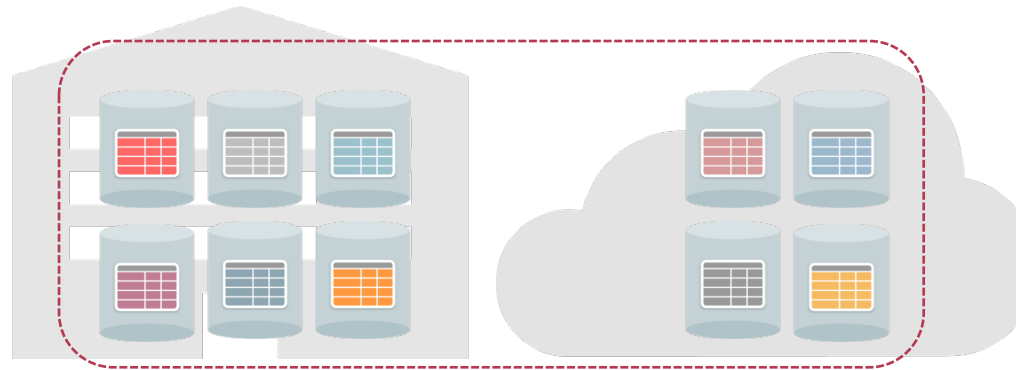
# Sharding – a Superior Way to Deploy Database

Flexible On-Premises or Cloud Deployment

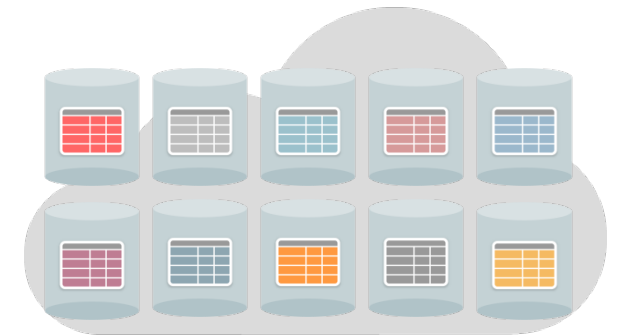
On-Premises



Hybrid



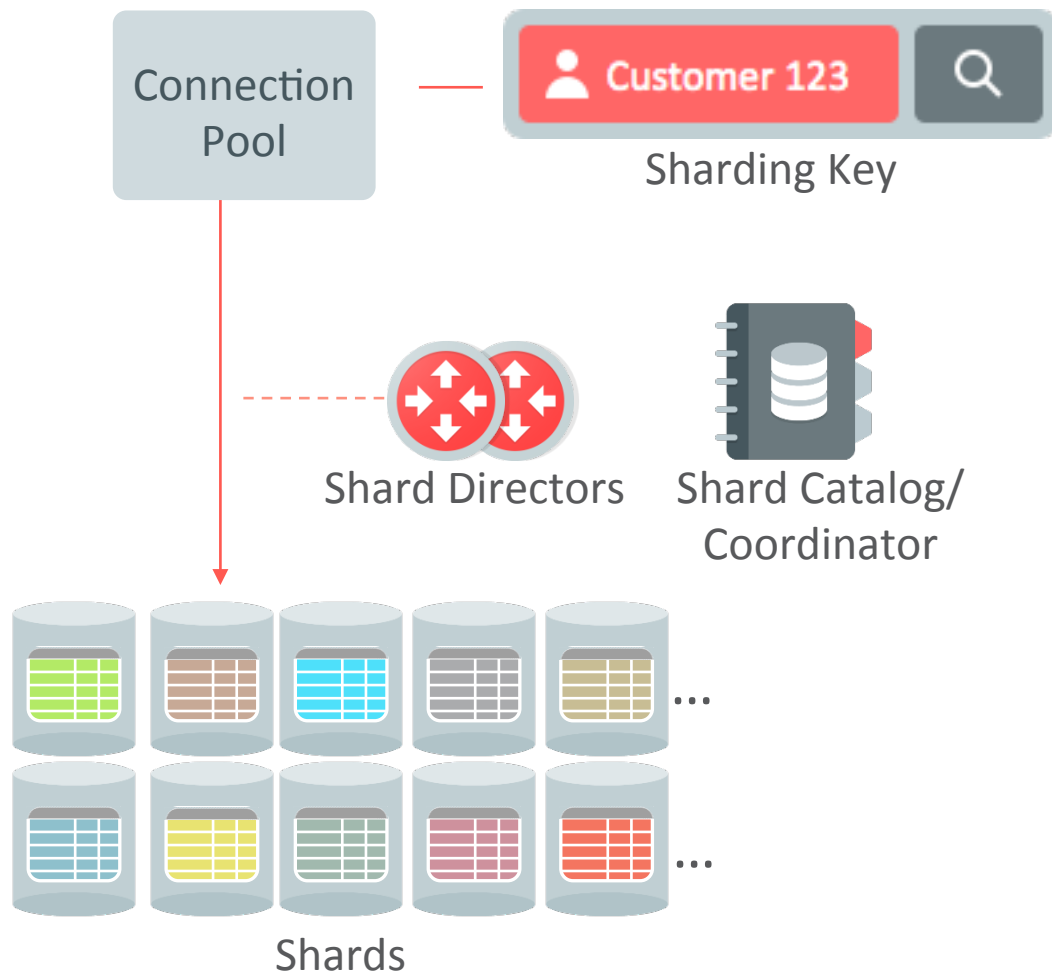
Cloud



# Program Agenda

- 1 Sharding Overview
- 2 Sharded Database and Schema Creation
- 3 Data Dependent Routing
- 4 MAA Benchmark Results - Sharding on Bare Metal Cloud
- 5 Customers' Perspective
- 6 Competitive

# Declarative Specification of SDB Configuration



```
create shardcatalog -database <shardcat host>:  
1521:shardcat -user sdb_admin/passwd_sdb_admin
```

```
add gsm -gsm sharddirector1 -listener 1571 -pwd  
passwd_gsmcatuser -catalog <shardcat host>:1521:shardcat
```

```
..  
add credential -credential oracle_cred -osaccount oracle -  
ospassword < >
```

```
add shardgroup -shardgroup shgrp1 -deploy_as primary -  
region avail_domain1
```

```
add invitednode <>
```

```
..  
create shard -shardgroup shgrp1 -destination <host1> -  
credential oracle_cred
```

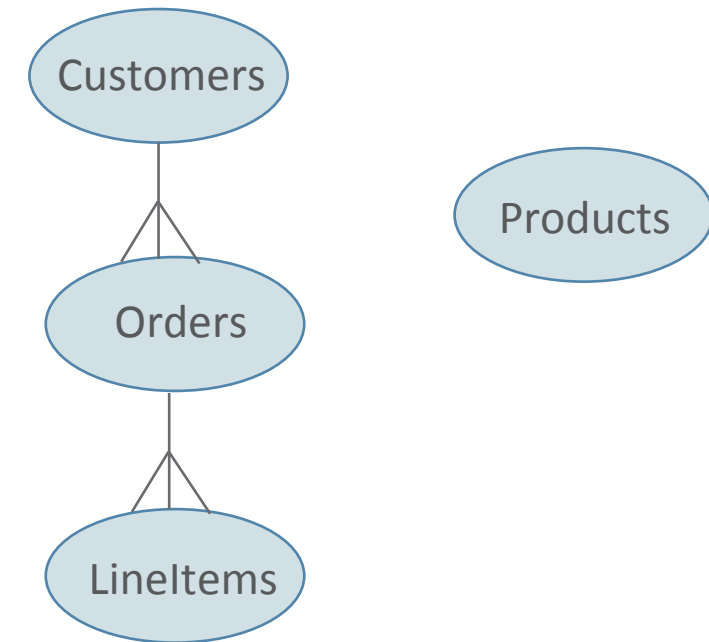
```
...
```

```
deploy
```

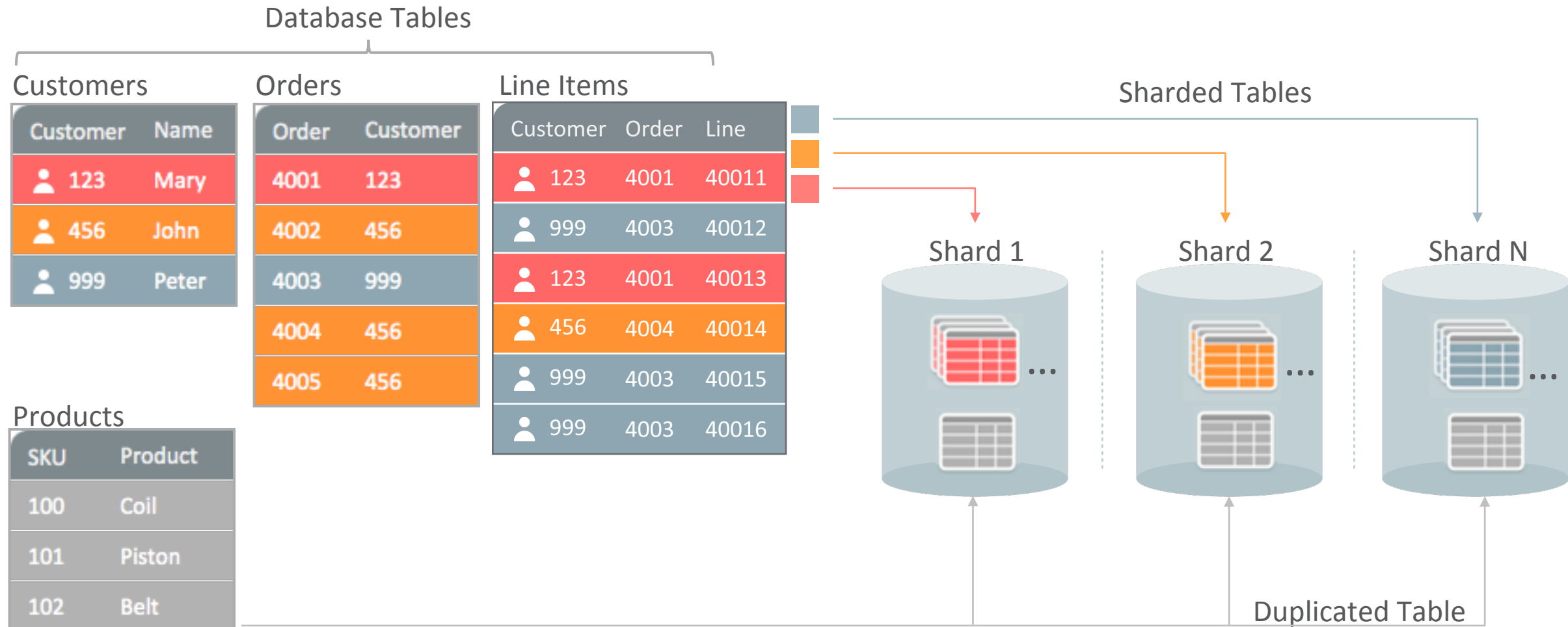
```
add service -service oltp_rw_srvc -role primary
```

# Data Modeling Considerations

- To reap sharding benefits, schema must be designed to maximize number of single-shard requests
- Sharding-amenable schema consists of:
  - **Sharded Table Family:**
    - Set of tables equi-partitioned by the sharding key
      - Related data is always stored and moved together
      - Joins & integrity constraint checks are done within a shard
    - Sharding method and key are based on App requirements
    - Sharding key must be the leading column of a primary key
  - **Duplicated Tables:**
    - Non-sharded tables are replicated to all shards
    - Usually contain common reference data
    - Can be read on each shard



# Schema Creation – Sharded and Duplicated Tables



# Creating a Sharded Table Family with Referential Integrity

Execute DDLs on the Shard Catalog

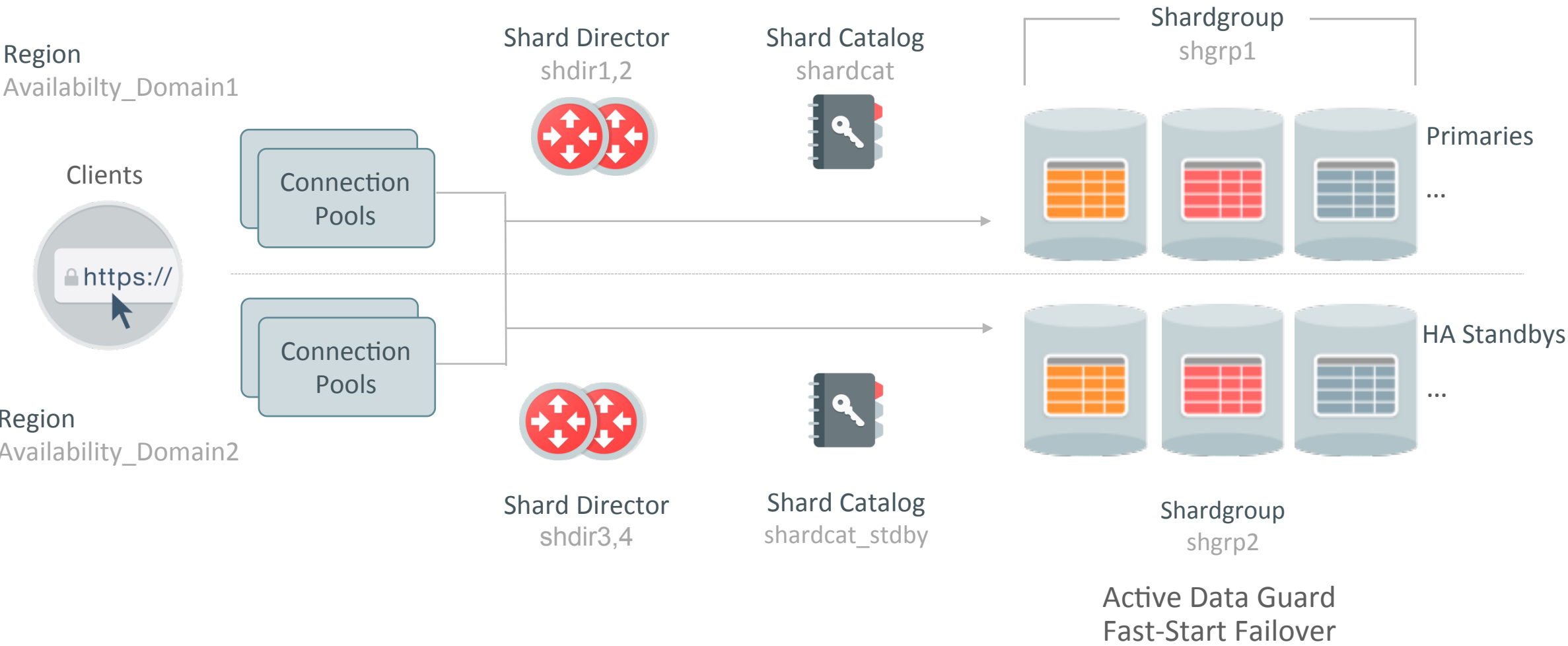
```
CREATE TABLESPACE SET tbs1 ;

CREATE SHARDED TABLE Customers
( CustId      VARCHAR2(60) NOT NULL,
  FirstName   VARCHAR2(60),
  LastName    VARCHAR2(60),
  ...
  CONSTRAINT pk_customers
    PRIMARY KEY(CustId)
)
PARTITION BY CONSISTENT HASH (CustId)
PARTITIONS AUTO
TABLESPACE SET tbs1 ;
```

```
CREATE SHARDED TABLE Orders (
  OrderId     INTEGER,
  CustId      VARCHAR2(60),
  OrderDate   TIMESTAMP,
  ...
  CONSTRAINT pk_orders
    PRIMARY KEY (CustId, OrderId),
  CONSTRAINT fk_orders_parent
    FOREIGN KEY (CustId) REFERENCES
Customers(CustId)
)
PARTITION BY REFERENCE (fk_orders_parent) ;
```

```
CREATE DUPLICATED TABLE Products (
  ProductId   INTEGER PRIMARY KEY,
  Name        VARCHAR2(128),
  LastPrice   NUMBER(19,4),
  ...
)
TABLESPACE products_tsp ;
```

# MAA w/ Oracle Sharded Database





# Program Agenda

- 1 Sharding Overview
- 2 Sharded Database and Schema Creation
- 3 Data Dependent Routing**
- 4 MAA Benchmark Results - Sharding on Bare Metal Cloud
- 5 Customers' Perspective
- 6 Competitive

# UCP: Connection Request with a Shard Key

## An Example of New Sharding APIs

- Pool creation: no Java application code change
- Shard-aware application gets a connection, using UCP Sharding APIs

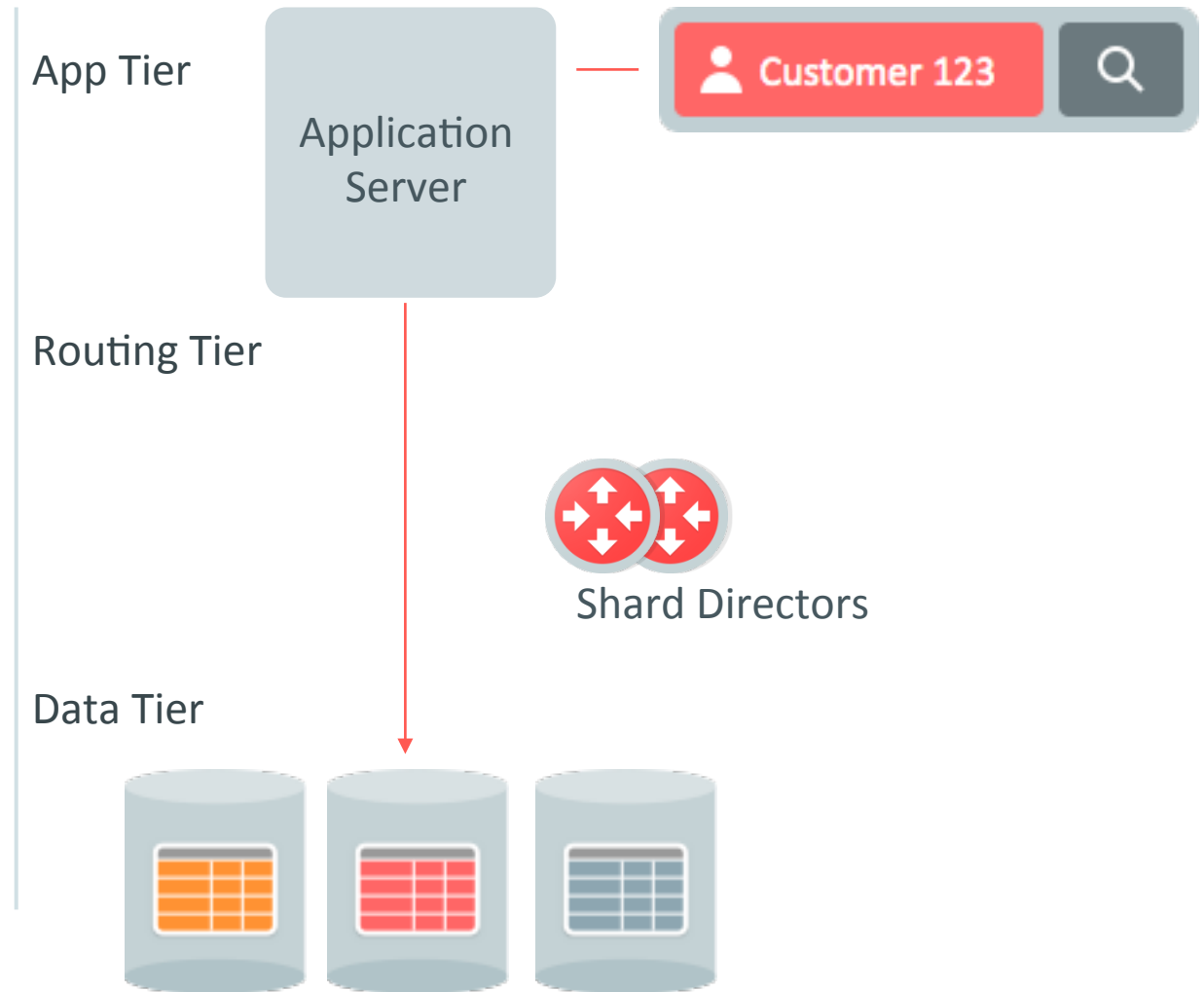
```
OracleShardingKey keyMaryEmail =  
    pds.createShardingKeyBuilder()  
        .subkey("mary.smith@xyz.com", OracleType.VARCHAR2)  
        .build();
```

```
Connection connection =  
    pds.createConnectionBuilder()  
        .shardingKey(keyMaryEmail)  
        .build();
```

# Direct Routing via Sharding Key

## Fast Path for key-based access

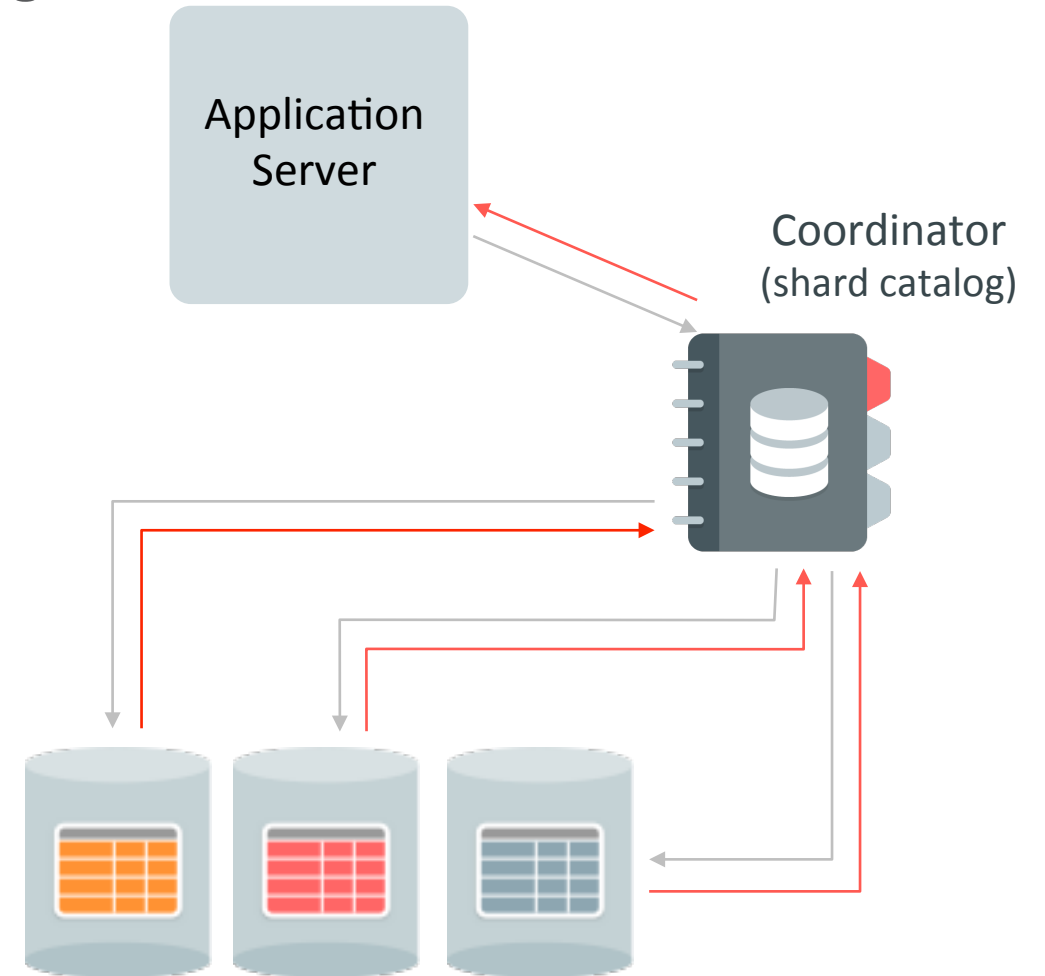
- Connection pool maintains the shard topology cache
  - Upon first connection to a shard
    - Connection pool retrieves all sharding key ranges in the shard
    - Connection pool caches the key range mappings
- DB request for a key that is in any of the cached key ranges goes directly to the shard (i.e., bypasses shard director)



# Proxy Routing for Multi-shard Queries

## For Workloads that cannot pass sharding key during connection check-out

- Applications connect to Query coordinator/Shard Catalog
  - E.g. `SELECT GEO, CLASS, COUNT(*) FROM CUSTOMERS GROUP BY GEO, ROLLUP(CLASS);`
- Coordinator rewrites the query to do most processing on the shards
- Supports shard pruning and scatter-gather
- Final aggregation performed on the coordinator



# Lifecycle Management of SDB

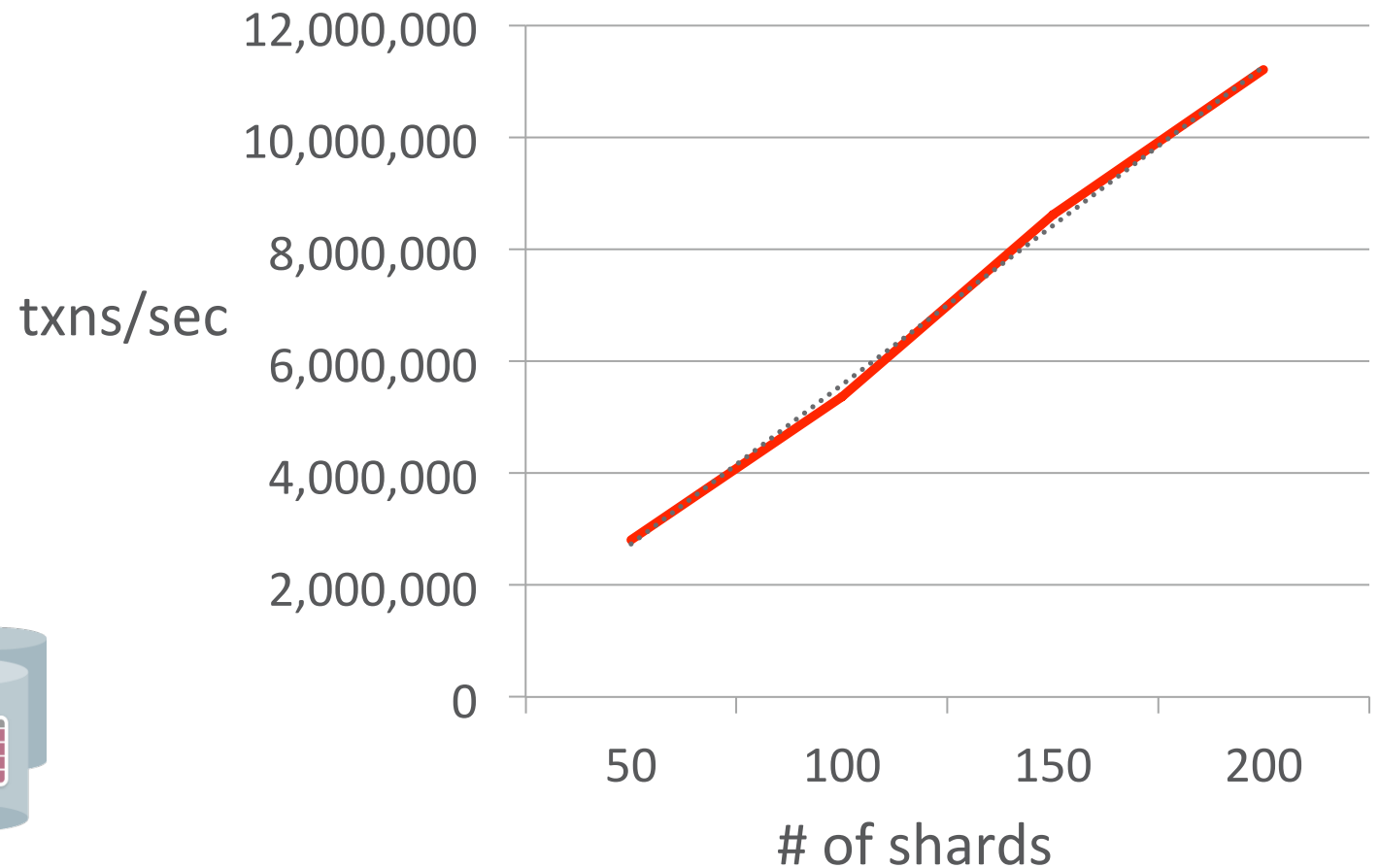
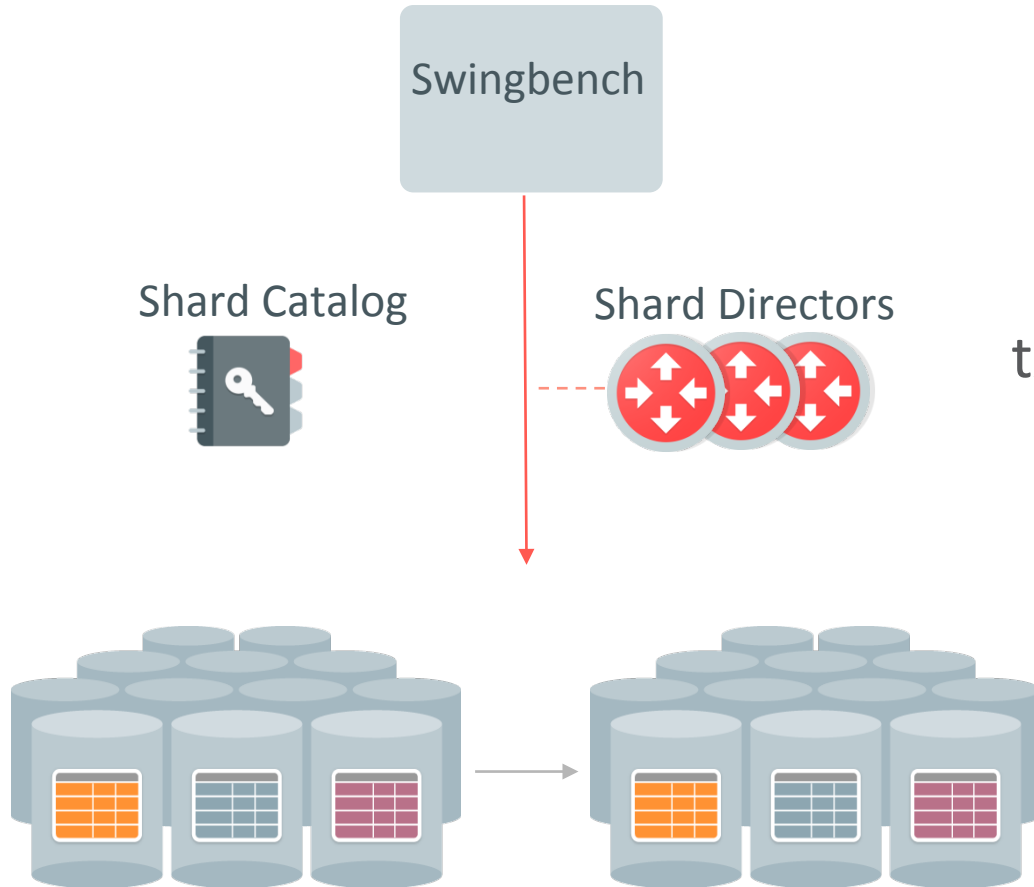
- DBA can manually **move or split** a chunk from one shard to another
- When a **new shard is added**, chunks are automatically rebalanced
  - Uses RMAN Incremental Backup & Transportable Tablespace
- Connection pools get notified (via ONS) about split, move, add/remove shards, auto-resharding
  - Application can either reconnect or access read-only
- All MAA practices apply for **backup and recovery of shards and shard catalog**
- Can patch all shards with one command via **opatchauto**
- **EM** supports monitoring & management of SDB
- Oracle SQL Developer integration

# Program Agenda

- 1 Sharding Overview
- 2 Sharded Database and Schema Creation
- 3 Data Dependent Routing
- 4 MAA Benchmark Results - Sharding on Bare Metal Cloud**
- 5 Customers' Perspective
- 6 Competitive

# Demonstrated Linear Scalability

## Oracle Sharded Database on Oracle Cloud Infrastructure



# Program Agenda

- 1 Sharding Overview
- 2 Sharded Database and Schema Creation
- 3 Data Dependent Routing
- 4 MAA Benchmark Results - Sharding on Bare Metal Cloud
- 5 Customers' Perspective**
- 6 Competitive





# Oracle Database Native Sharding: a Customer Perspective

John Kanagaraj, Sr. Member of Technical Staff, PayPal Core Data Platform

# Sharding Principles

- Sharding is mostly a Scalability play
- "Key is Key" – Choose one STRONG key and align. E.g. Account/User ID
- Data model changes (drastically!) – Normalization is relaxed
- Cross-key transaction boundaries are broken
- Non-shard key access can be expensive; forces "scatter-gather" pattern
- CAP Theorem: 2 of 3 for (C)onsistency, (A)vailability and Network (P)artition
- "Lookup" requirement for Common data elements
- Scheme for mapping logical to physical is critical for future scale-out
- Joins and ACID principles usually not available in Sharded systems (e.g. NoSQL)
- *Oracle Sharding supports the sharding principles while providing consistency, transactions, and relational capabilities such as Joins*

# Oracle Sharding – Feedback from Beta software

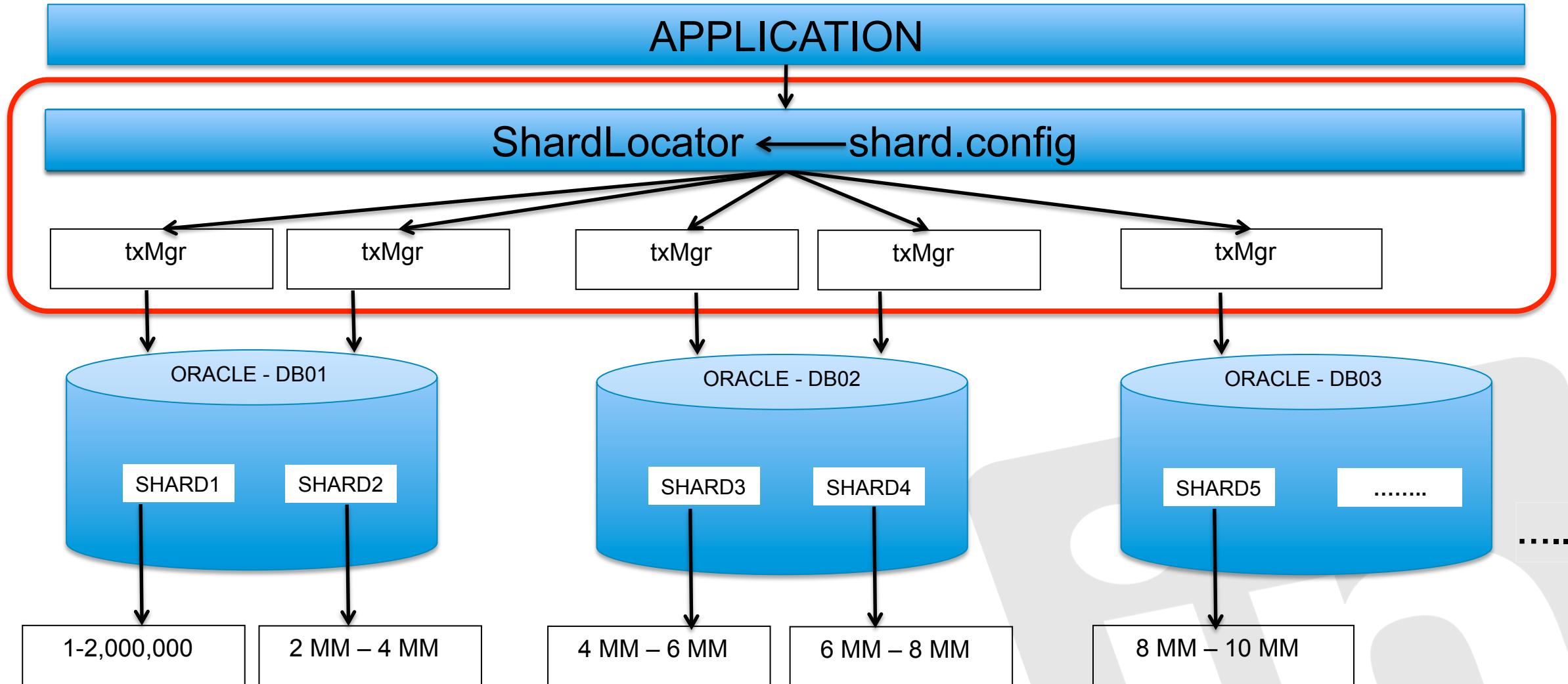
- What we liked
  - Auto-configuration (Create and Manage Shards and Routing)
  - Out of box support for Shard Catalog, Routing layer, Duplicated Tables
  - Cross-shard query support (Not available with custom sharding)
  - Familiar SQL for sharding management
  - Two level sharding (not available in other technologies such as NoSQL)
  - Consistent hashing to prevent large data movement as new shards are added
  - Dynamically changing the number of shards
- Things to be careful of
  - Sharding requires major rework in Data Model and Data Design
  - Duplicated Table – Make sure not to over-use; Not for write-heavy use cases
  - Enable connection pool limits per Shard

# Sharding Use Case – LinkedIn

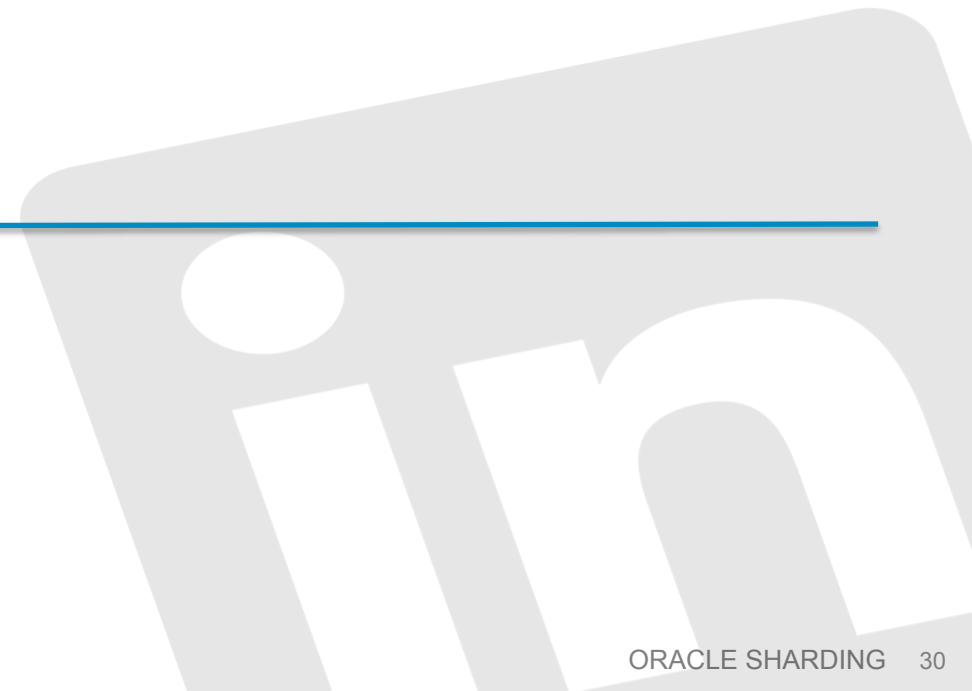
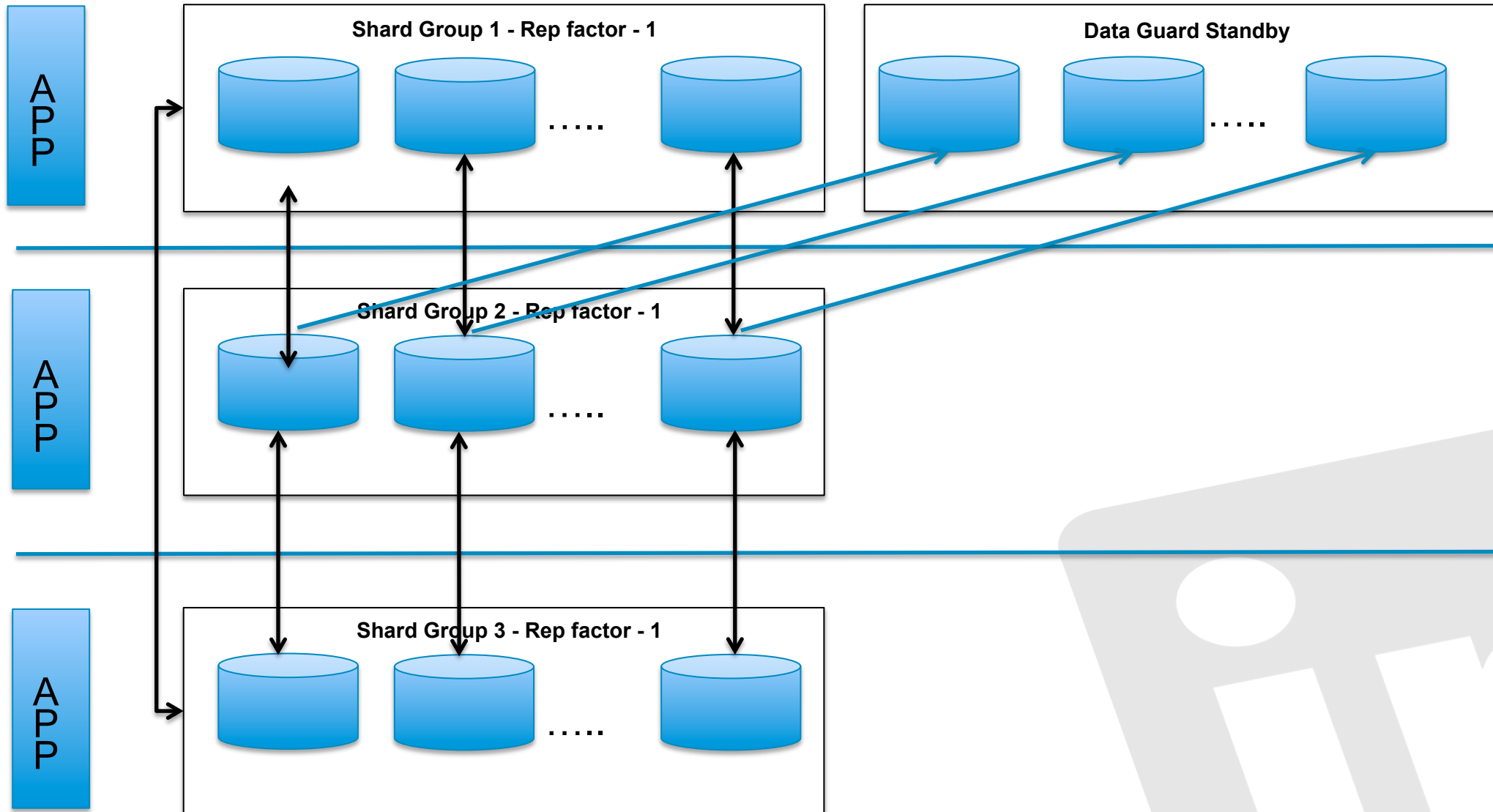
## Business-oriented Social Networking with 380 Million Members in Over 200 Countries

- Target application
  - LinkedIn uses home-grown sharding with Oracle Database to scale customer facing applications
- Challenges faced/key business driver
  - Manual re-configuration during rebalancing and new table additions
  - No multi-shard queries or composite sharding capabilities
- How does Oracle Sharding addresses LinkedIn's requirement?
  - Online addition of shards and automatic rebalance with no impact to application
  - Architectural simplicity
  - Composite sharding and multi-shard queries
  - Efficient use of CAPEX

# Home Grown Sharding Architecture: LinkedIn



# Where LinkedIn Wants To Be - Oracle 12c Sharding

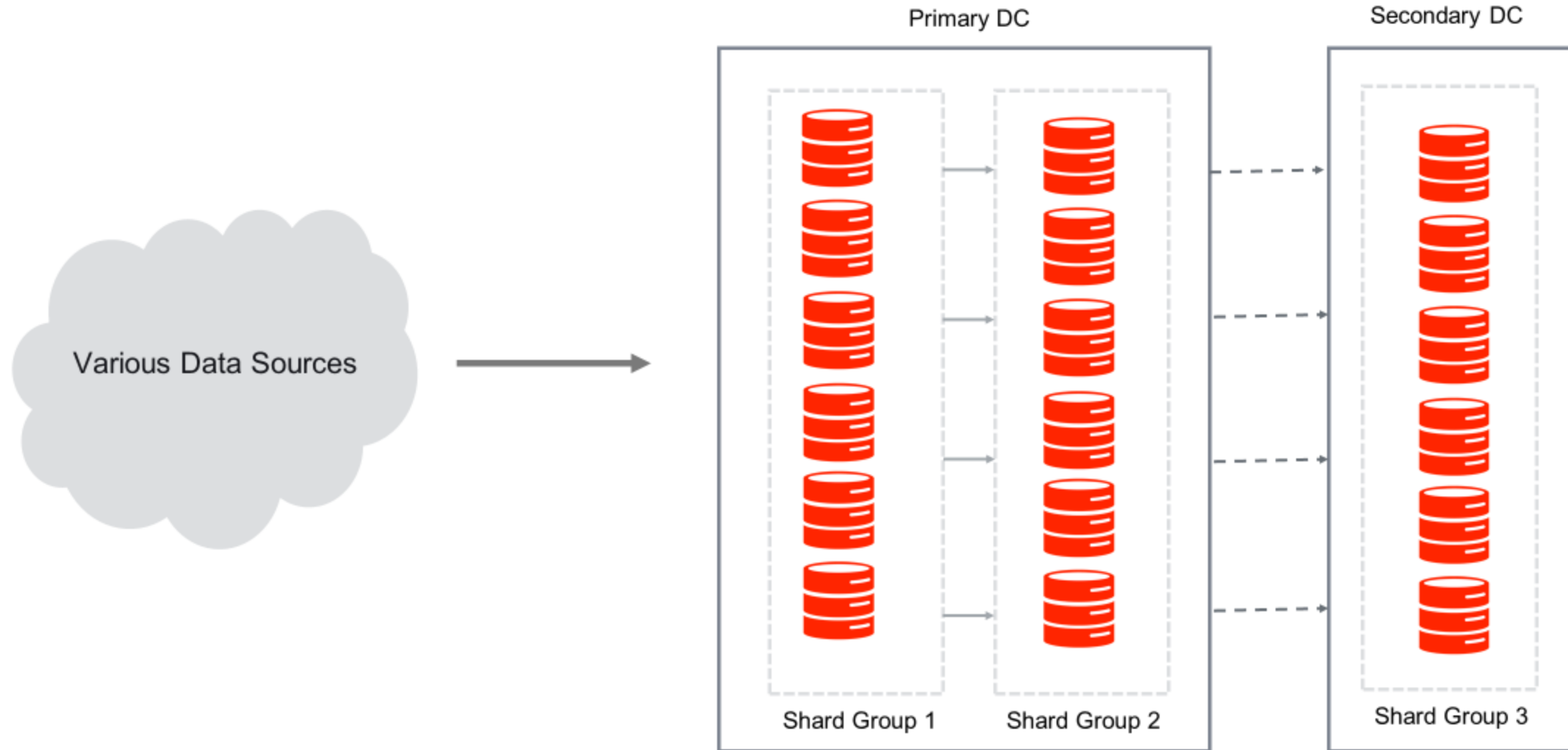


# Oracle Database Native Sharding

Sudhi Vijayakumar

Database Systems Architect, Facebook Inc

# Target Sharded Architecture





# Benefits of Oracle Sharding

- Full power of Oracle RDBMS
- Deployments can be made automatic
- Open Compute Platform can be used for deployments
- Horizontal scaling based on demand/growth
- Reduce SPOF
- Reduce tech debt (apply patches in rolling manner, etc.)
- Uses familiar technology (Data Guard, GoldenGate)

# Key Metrics for Oracle Sharding POC

- Support Facebook's existing and growing ingestion rate
- Response times for queries
- Ease of Deployment
- Ease of Maintenance
- Iterative design changes

# Program Agenda

- 1 Sharding Overview
- 2 Sharded Database and Schema Creation
- 3 Data Dependent Routing
- 4 MAA Benchmark Results - Sharding on Bare Metal Cloud
- 5 Customers' Perspective
- 6 Competitive

# Oracle Sharding = Best of NoSQL + Best of Oracle

- Oracle Sharding Key features
  - Automatic data distribution across a set of commodity servers (using consistent hash)
  - Built-in and customizable replication (Active Data Guard and GoldenGate)
  - Linear scalability (tested up to 1,000 nodes)
  - **Plus: Composite and user-defined sharding, duplicated tables, centralized management, ...**
- With the strength of Oracle RDBMS Engine
  - Familiar and powerful relational data model and query language, including full native support for modern datatypes like JSON, Spatial, Graph, ...
  - Transactions, recovery, in-memory ...
  - Industrial-strength and scalable storage engine
  - Data security and access controls – Database Vault, Label Security, ...
  - Familiar to current staff, and supported by Oracle

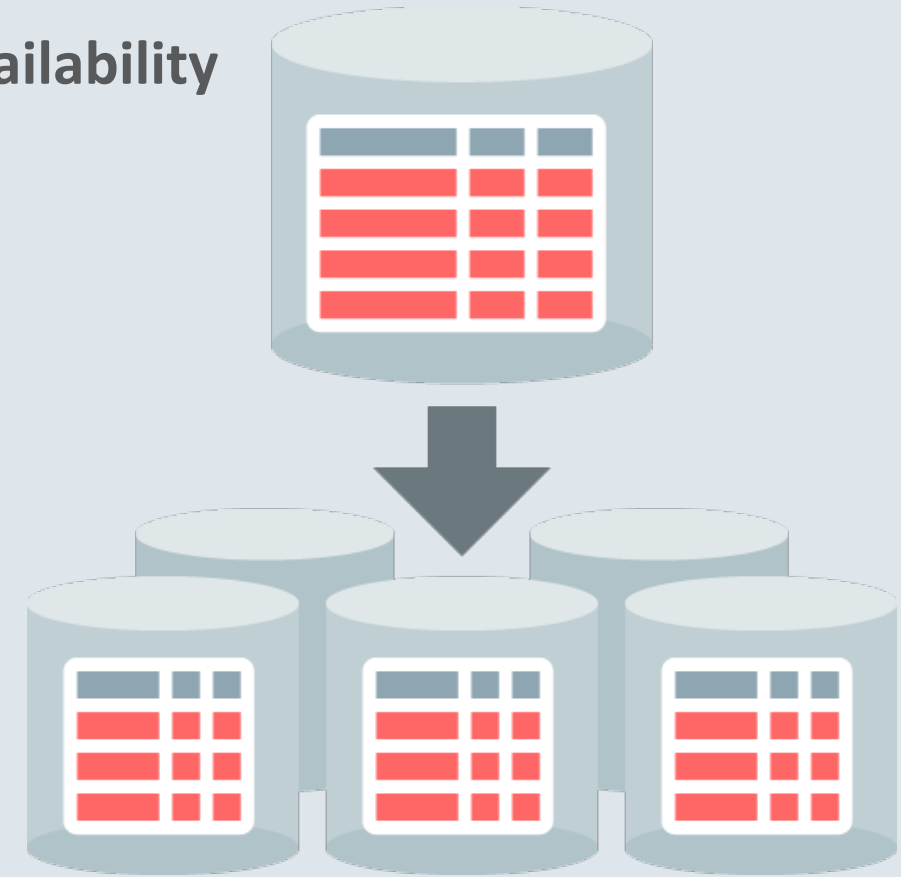
# Summary | 3 Takeaways

- Complete **platform for sharding** an Oracle database
  - *Sharding=data partitioning & distribution + colocation of related data + data replication + data dependent routing+ lifecycle management*
- Ideal for applications whose **data access pattern is primarily via a sharding key** and that require one or more of these characteristics:
  - Linear scalability for workload, data size and network throughput
  - Geographic data distribution
  - Fault Isolation
- **Licensing**
  - Oracle Cloud: Extreme Performance
  - On-premises: Enterprise Edition + either Active Data Guard or GoldenGate or RAC

# Oracle Sharding – Session II

**Linear Scalability, Geo-distribution and Extreme Data Availability  
for Web-scale Applications**

Mark Dilman, Director  
Oracle Sharding Product Development



# 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 ➤ China Telecom – Sharding for WeChat IoT Application
- 2 ➤ Oracle Dyn's experience with Oracle Sharding
- 3 ➤ Summary



# Program Agenda

- 1 China Telecom – Sharding for WeChat IoT Application
- 2 Oracle Dyn's experience with Oracle Sharding
- 3 Summary



# Oracle Sharding for China Telecom's WeChat IoT Application

---



# About China Telecom



- Large-scale and leading integrated information service operator in the world
- Provides
  - Landline & mobile telecommunications services
  - Internet access services
  - Information services and other value-added telecommunication services
- As of 2016, mobile subscribers of about 215 million
- China Telecom is building Business Support System(CRM/Billing etc.) for Narrow-Band IoT (NB-IoT) network
  - Provides packaged offers, self service on account balance, call billing, number status, etc., process of problem repairs, repair progress inquiry, etc.
- WeChat Service (part of BSS ecosystem) provides customer service by WeChat
  - System will also manage 2/3/4G network users
- NB-IoT Network user base is projected to grow exponentially

# Motivation for Oracle Sharding



- We took sharding into consideration for the projected large number of NB-IoT Network users
- Main benefit is scalability
  - *“We do not need to worry about what to do when we need to scale to larger number of users”*
- Evaluated Oracle Sharding, MySQL, MongoDB and Maria DB
  - *Determined that migration cost is too high if we go to other data stores*
  - *“DBAs and Developers are familiar with Oracle database. Since Oracle has sharding, why don’t we use Oracle?”*
- Migrated the application from Oracle 11g RAC to Oracle Sharding
- Plan to migrate other applications of NB-IoT BSS to Oracle sharding (based on scalability, performance and availability in production)

# Grand Release – Fast Verification



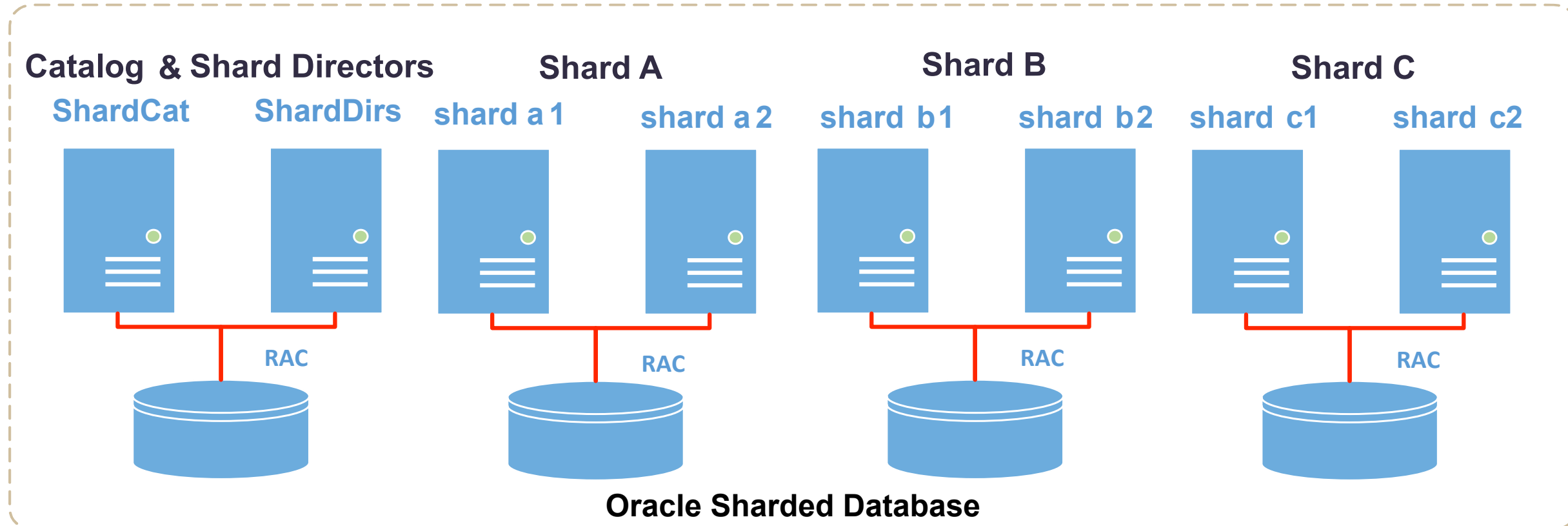
- Oracle 12c Sharding formally announced in March 2017
  - Sharding Beta1 released in 2015
    - Generated lot of interest in China
  - In 2016 Zhejiang Mobile selected its customer center as test model
    - HuZhou as testing ground for pre-study
  
- On June 29, 2017, China Telecom IOT WeChat Customer Service went into production with Oracle 12c sharding
  - Application and database are both running stable after going online

# Oracle Sharding Deployment

Current Oracle 12.2 sharded environment has 8 database servers

Created 4 independent databases: **Shard Catalog and Shards** across 8 nodes

Used 2-node RAC at shard-level



# Future Plans



- With current Oracle sharding environment, **continue to deploy new applications, such as: China Telecom 10000 WeChat mini-program**
- Keep in touch with Oracle experts, obtain Oracle Sharding new developments and information, upgrade to major and minor versions in time
- Increase communication with like-minded colleagues from other provincial companies, collaborate, and more widely adopt Oracle Sharding across other applications

# Program Agenda

- 1 China Telecom – Sharding for WeChat IoT Application
- 2 Oracle Dyn's experience with Oracle Sharding**
- 3 Summary



A person is shown from the chest up, wearing a VR headset. They are holding the sides of the headset with both hands. The person has a joyful expression, with their mouth open in a smile. The background is blurred, suggesting an indoor setting with other people. The entire image is overlaid with a semi-transparent red geometric pattern consisting of large triangles.

ORACLE  
OPEN  
WORLD

October 1–5, 2017  
SAN FRANCISCO, CA

# Customer Driven Database Design

**2 Trillion Rows of DNS Query Volume**

Charlie Baker  
Sr. Director Product Management  
Oracle Dyn GBU  
October 4, 2017

[charles.baker@oracle.com](mailto:charles.baker@oracle.com)

ORACLE®

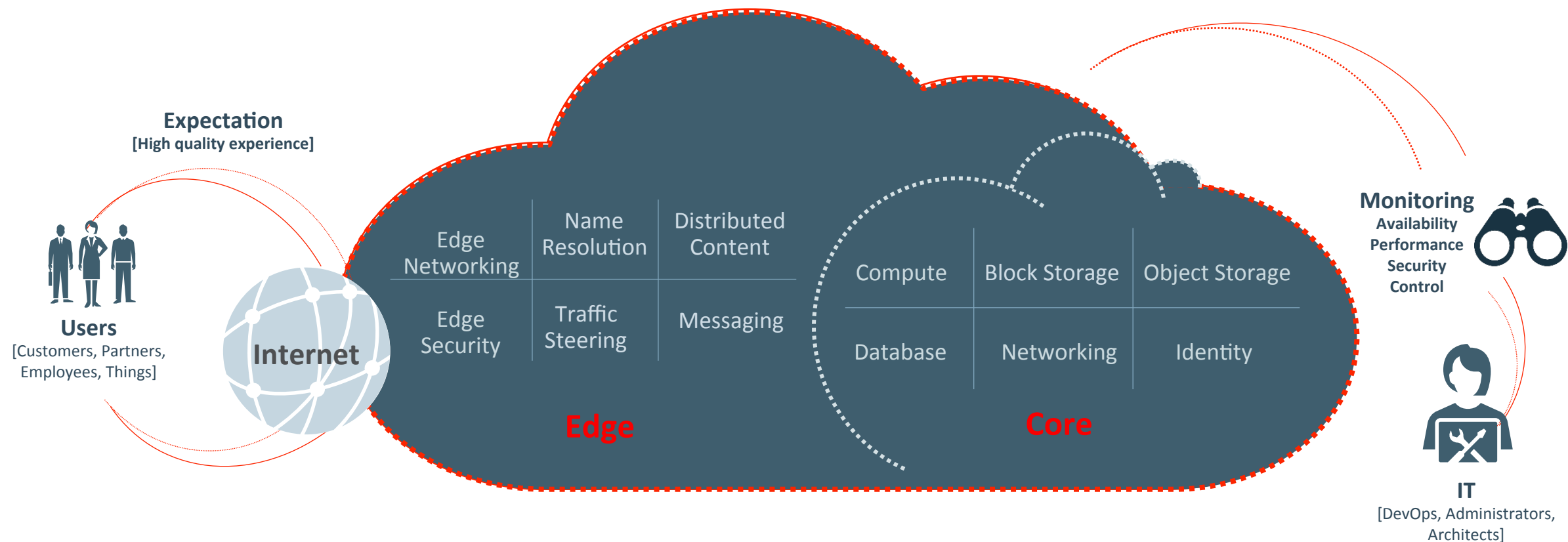
Copyright © 2017, Oracle and/or its affiliates. All rights reserved. |

# Quick Agenda

- 1 Who is Oracle Dyn
- 2 The Problem
- 3 Solutions Evaluated and Benchmarks
- 4 Deployed Solution Architecture
- 5 Benefits and Next Steps

# Dyn Strengthens Oracle Cloud Infrastructure

Mission | Create the World's Most Capable Enterprise Cloud Infrastructure from Edge to Core



The Next Generation Cloud



Trusted by Over 3,500 Customers, Including Some of the Most Preeminent Digital Brands

ORACLE® + Dyn

Over 40 Billion DNS Queries Per Day

Over 214 Billion data points - powering analytics and steering decisions



Visit [dyn.com/OOW17](https://dyn.com/OOW17) for more information and downloads.





















Visit our product experts in the Oracle Cloud Infrastructure expo

# The Problem – High-Scale Billing, Reporting & Analytics

- Globally Distributed Name servers answering DNS Queries
  - Fully attributed with ASN, GEO-ID, Country Code
- Data Volume Generated:
  - Estimated at 2 Trillion Rows for 90 days of query data
  - Raw data of 1.6 TB/day working out to over 144 TB across 90 days
- Query Latency Measured across Multiple Systems:
  - UI
    - Splash page queries < 5s
    - Detailed investigation queries < 20s
  - API
    - Single customer queries < 20s
    - Multiple customer queries (billing/metering) < 1min

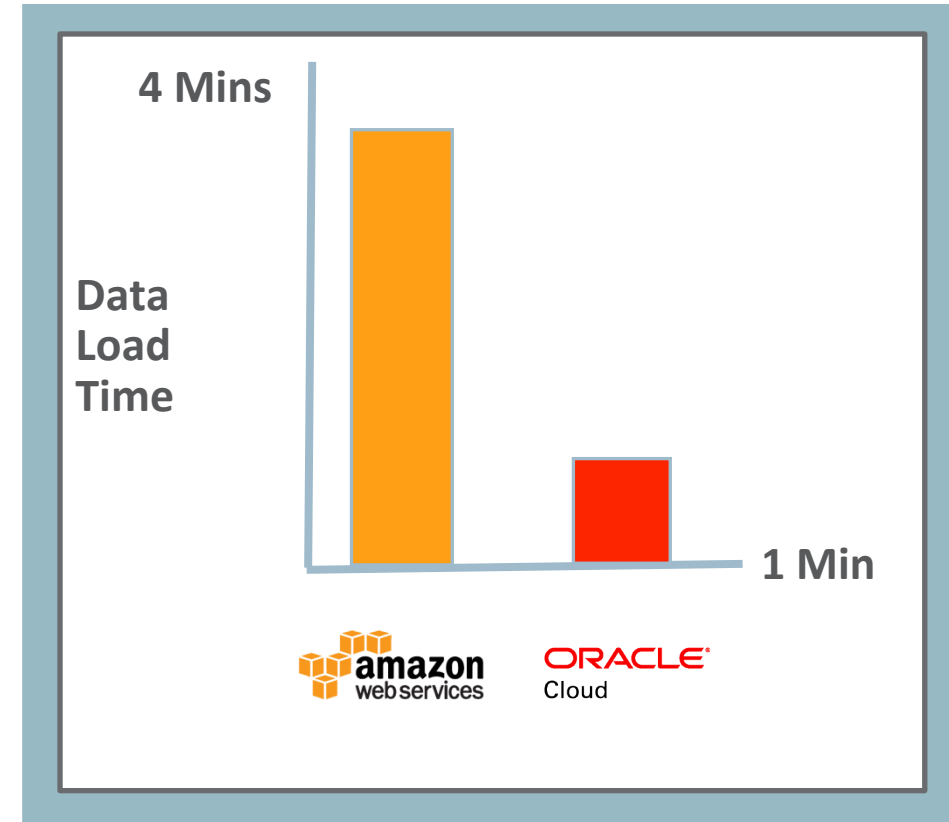


# Solutions Evaluated

 	   <p>Complex system design Data loads could not keep up with incoming data volume Queries were unacceptably slow</p>
	   <p>Could ingest and index at scale Required large shared storage Difficulties running multiple GROUP BY queries</p>
	  <p>Limited expertise at Dyn Unable to support complex queries, forcing application to implement complex query processing logic</p>
 Cloud - Single Database	   <p>Good Performance and Ingest speed Complex query support Performance and Storage limited to Single Database</p>
 Cloud - Sharded Database	   <p>Superior Ingest speeds Scale queries by distributing across as many servers as needed (up to 1000) Near limitless TB of storage by adding more Shards (up to 1000)</p>

# Evaluation Results

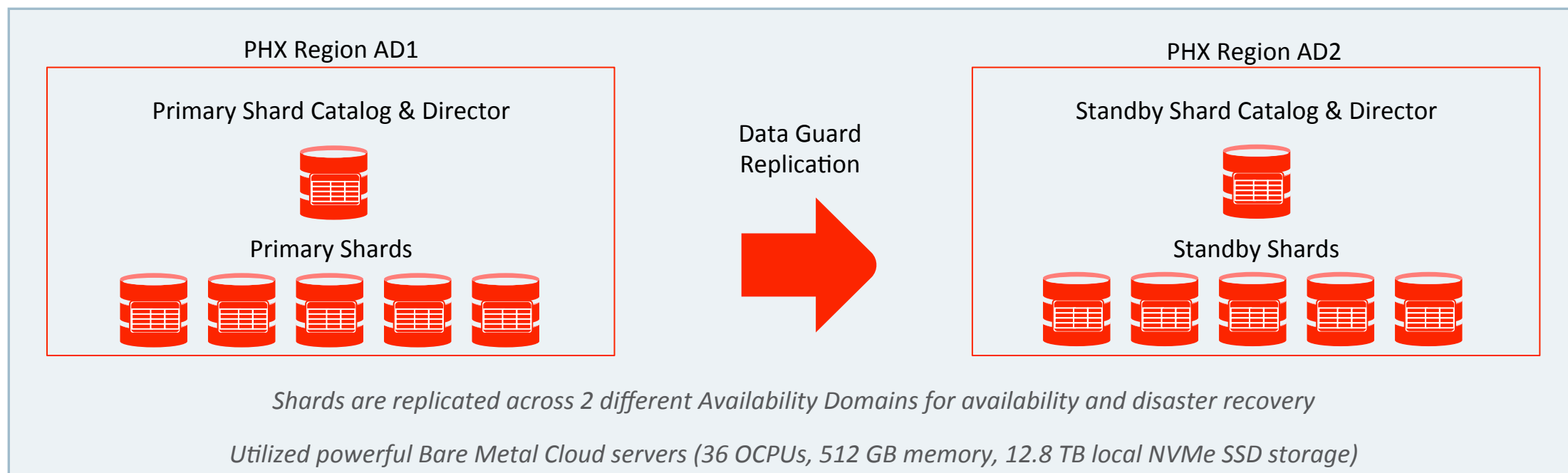
- Loading from S3 to Redshift
  - 3 to 4 minutes
- Loading from Oracle Object Storage to Shards
  - 500k rows / second
  - Full load in 1 minute
  - 4 minutes of headroom for queries



# Deployed Solution

## Oracle Sharded Database

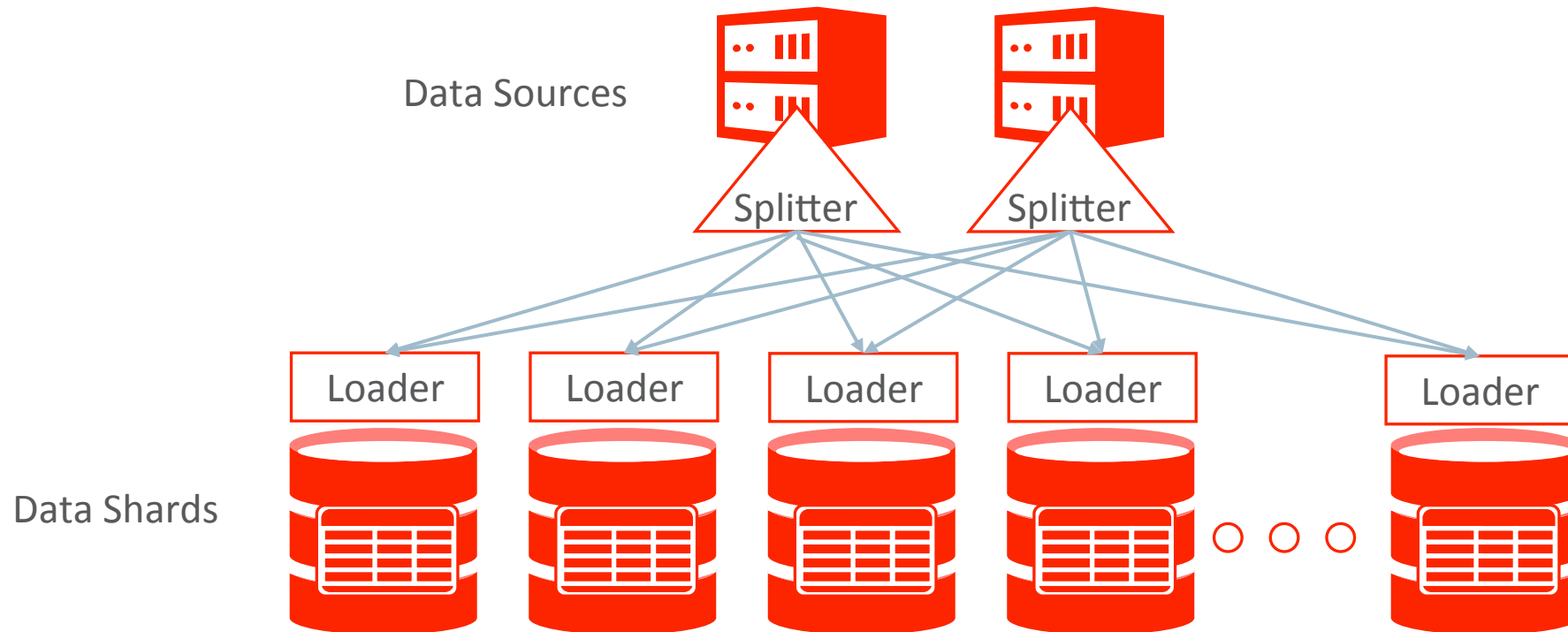
- Proven linear scalability
- Ingest speeds scale with number of shards
- Constant query time even as we grew size of dataset
- Geo-distributed to be close to our customers





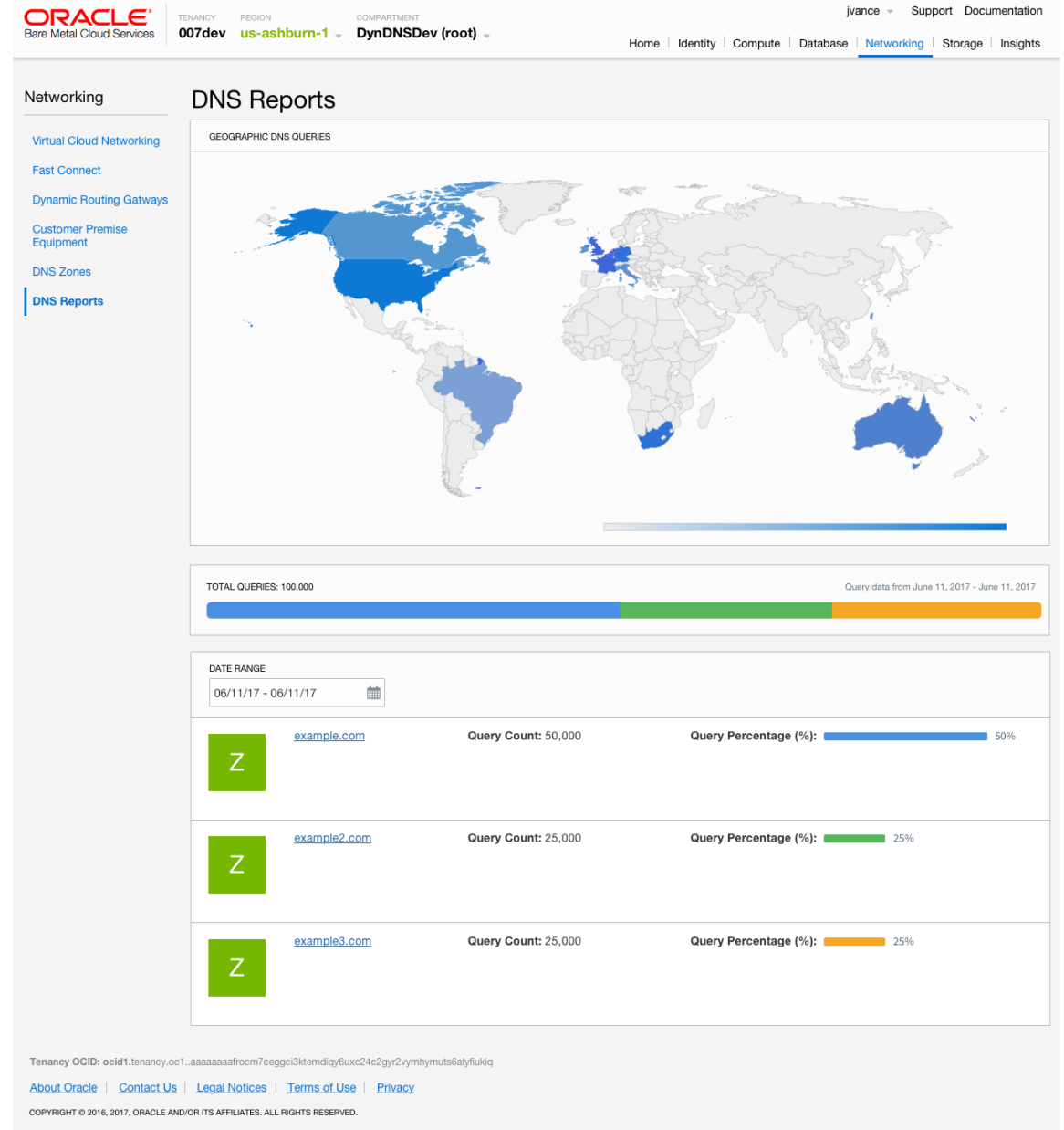
# High Speed Data Load

- Utilized fully parallel direct-to-shard data loader
- Sharded architecture scales the CPU, flash, and network interfaces
- Can add shards as needed to accommodate higher ingest rates
- Architecture applies to IIoT, IoT and edge compute scenarios



# Testing the User Interface

- Basic query time < 2 seconds
  - Total requests over period of time
- Complex query time <20 seconds
  - Requests by geography, operator, recursive over period of time



# Schema Design

**Most important aspect of sharding is proper schema design to minimize cross-shard communication**

This application has one large table

1. Shard by customer ID to minimize cross-shard queries
2. Converted customer ID into an internal customer IDs for better data distribution
3. Normalize data to reduce data size and store external to internal ID mapping
4. No indexes on raw data to keep up with the incoming data rate
5. Sub-partitioning by time interval (3 hours) for partition pruning and ILM (archiving/dropping old data)
6. Use compression to reduce data size (up to 5x) and speedup scans
7. Use rollup table to aggregate data (1 hour granularity) to speedup queries over large time intervals
8. Distributed data splitter/loader using OCI Direct Path API

# Wrap Up

- Oracle Sharding Allows Dyn to...
  - Deliver fast, real-time analytics to our customers, billing and analytics systems
  - Dynamically scale our computing infrastructure to handle ever-growing ingest rates, while keeping query times constant even though the data volume grows
  - Easily support geographic data distribution so that we can keep data closer to our customers
  - Collect more data per customer, so that we can make better traffic steering decisions and provide more value to customers
  - Create our data lake for analytics – including recursive, RUM, synthetic, remote access, CDN and WAF data sets
- Oracle Dyn DNS and Email services officially launch at OOW – leveraging the sharded DB

# Program Agenda

- 1 China Telecom – Sharding for WeChat IoT Application
- 2 Oracle Dyn's experience with Oracle Sharding
- 3 Summary

# Summary | 3 Takeaways

- Complete **platform for sharding** an Oracle database
  - *Sharding=data partitioning & distribution + colocation of related data + data replication + data dependent routing+ lifecycle management*
- Ideal for applications whose **data access pattern is primarily via a sharding key** and that require one or more of these characteristics:
  - Linear scalability for workload, data size and network throughput
  - Geographic data distribution
  - Fault Isolation
- **Licensing**
  - Oracle Cloud: Extreme Performance
  - On-premises: Enterprise Edition + either Active Data Guard or GoldenGate or RAC

# Oracle Sharding | Resources



[www.oracle.com/goto/oraclesharding](http://www.oracle.com/goto/oraclesharding)



<https://nageshbattula.com>

Oracle Maximum  
Availability Architecture

[www.oracle.com/goto/maa](http://www.oracle.com/goto/maa)

## Social Media:

Follow on Twitter:  
@nageshbattula  
@OracleSharding

ORACLE®