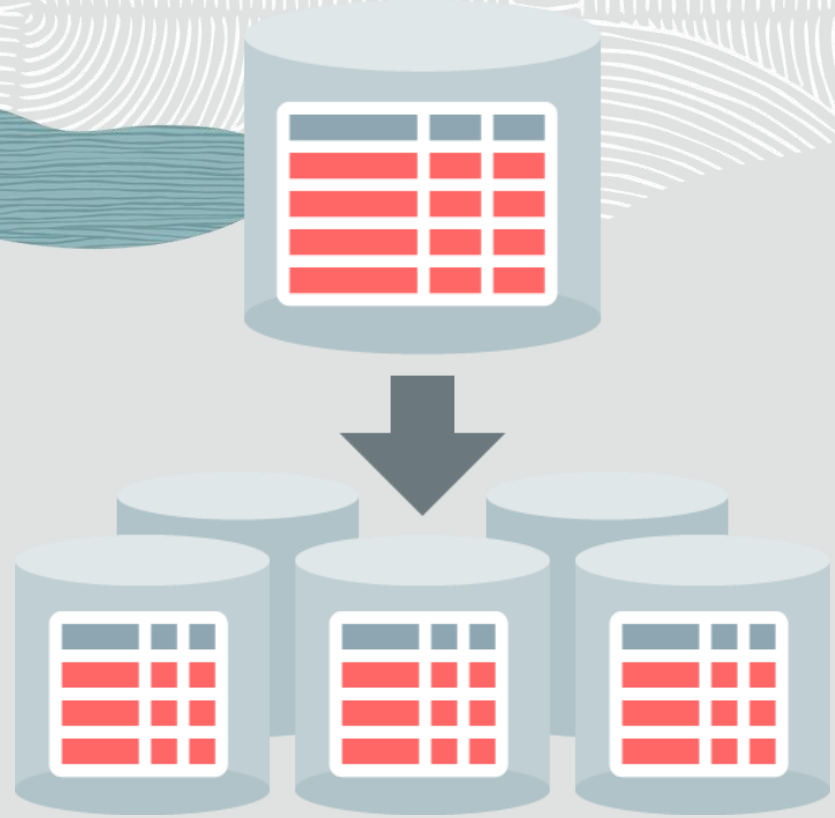
Oracle Sharding Overview



Oracle Korea

Safe Harbor

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, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

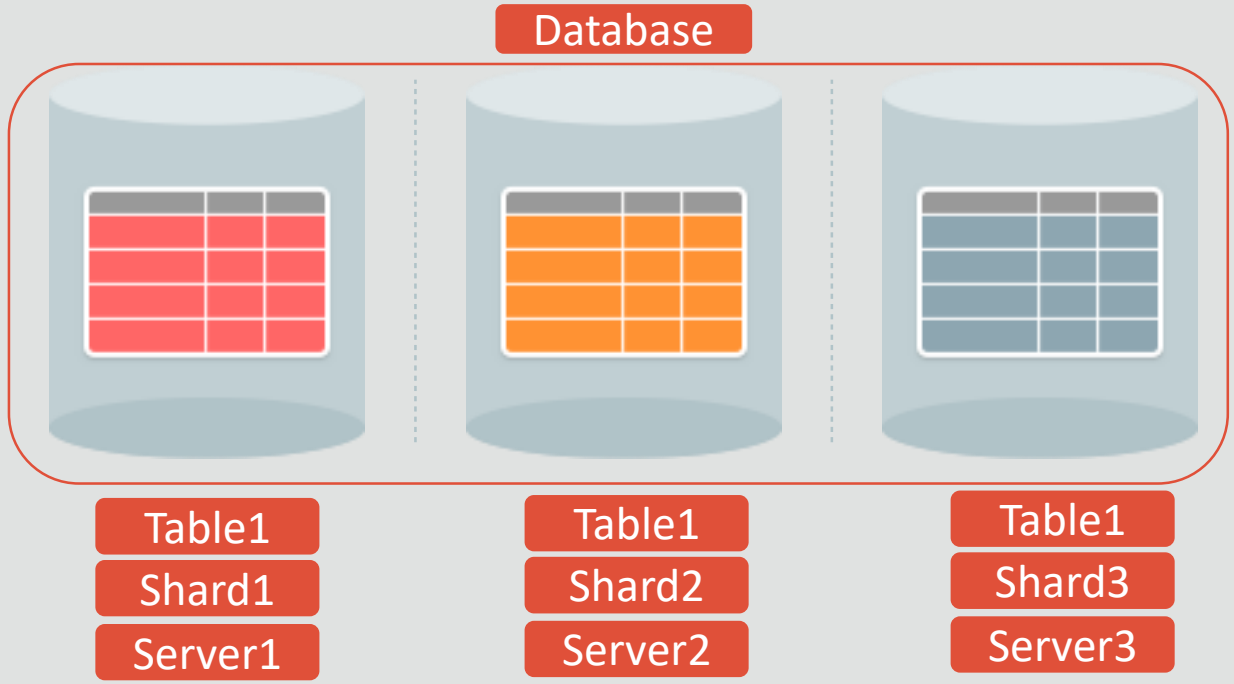
Statements in this presentation relating to Oracle's future plans, expectations, beliefs, intentions and prospects are "forward-looking statements" and are subject to material risks and uncertainties. A detailed discussion of these factors and other risks that affect our business is contained in Oracle's Securities and Exchange Commission (SEC) filings, including our most recent reports on Form 10-K and Form 10-Q under the heading "Risk Factors." These filings are available on the SEC's website or on Oracle's website at http://www.oracle.com/investor. All information in this presentation is current as of September 2019 and Oracle undertakes no duty to update any statement in light of new information or future events.

What Is Database Sharding?

Dominant approach for scaling Internet applications

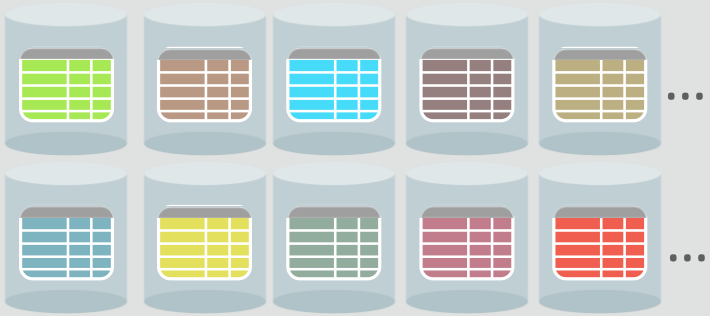
A single **logical DB** sharded into N physical Databases

- Horizontal partitioning of data across independent databases (shards)
  - Each shard holds a subset of the data
  - Can be single-node or RAC or PDB
  - Replicated for high availability
- Shared-nothing architecture:
  - Shards don't share any hardware (CPU, memory, disk), or software (clusterware)



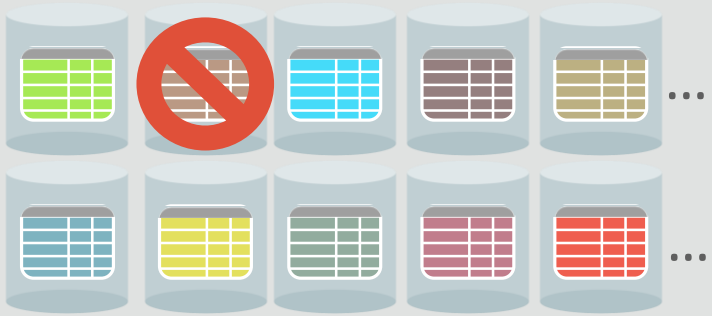
Oracle Database Sharding – Benefits

Linear Scalability



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

Extreme Availability



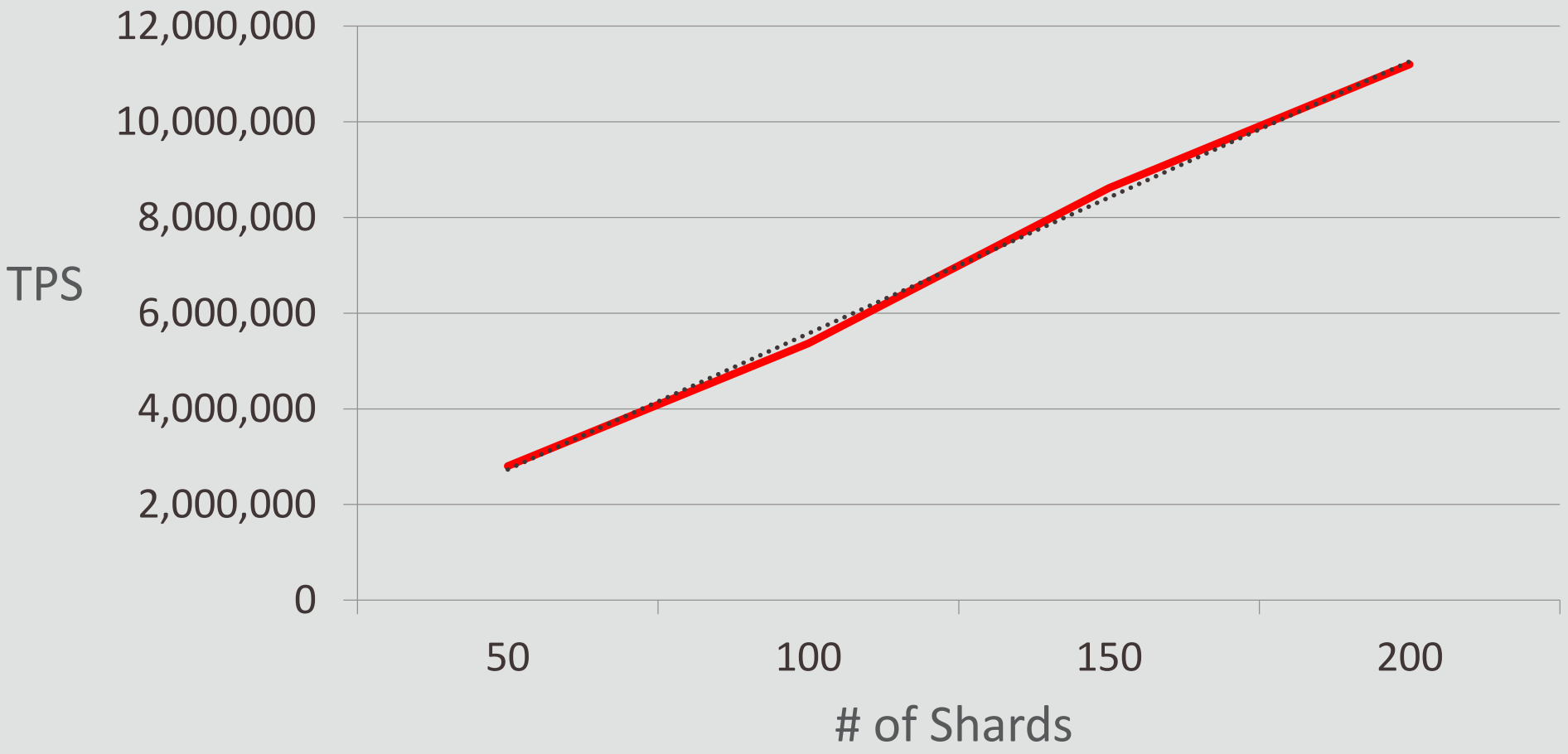
Shared-nothing architecture. Fault of one shard has no impact on others.

Geographic Distribution

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

Sharding – a Different Way to Scale

Frictionless linear scaling due to zero shared hardware or software



Sharding for Extreme Data Availability

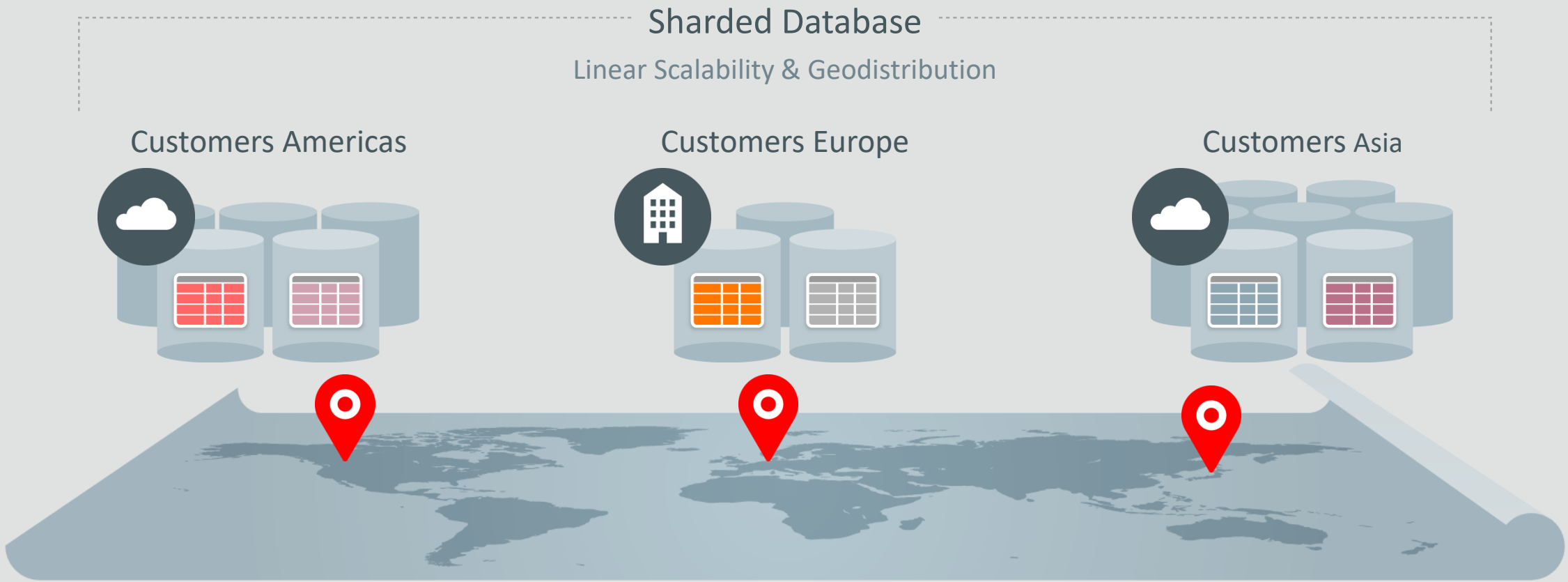
Fault Isolation

1%

The portion of users who undergo brown-out at one time by an unplanned outage or database upgrade in a sharded database with 100 shards

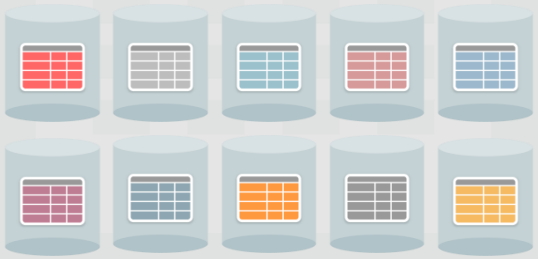
Sharding - Superior Way to Distribute Data

Data Sovereignty and Data Proximity



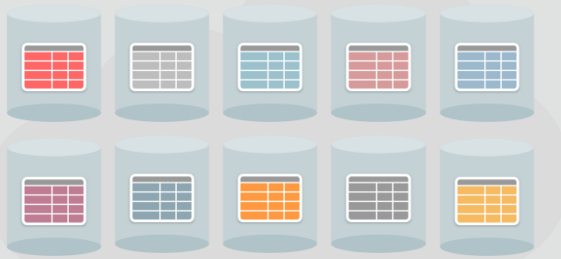
Sharding – Flexible Deployment Models

On-Premises



Hybrid

Cloud



Key Customer Use Cases

Customer Use Case	Other Products Evaluated by Customers before choosing Oracle Sharding
Internet Scale Realtime OLTP	Cassandra, MongoDB, MemSQL, MariaDB, Couchbase, Aerospike, ScyllaDB
Global Databases/Data Sovereignty	Google Spanner, Azure Cosmos DB, AWS Aurora, CockroachDB
Log Store	Apache Lucene, Elastic Search, Solr
Metric/Time Series store, IoT, Infrastructure Monitoring, APM	AWS Redshift/EMR, Druid, Cassandra, Graphite, InfluxDB
Machine Learning	Apache Spark, HDFS, NoSQL and SQL Sharded DBs
Big Data Analytics	Apache Spark, MemSQL

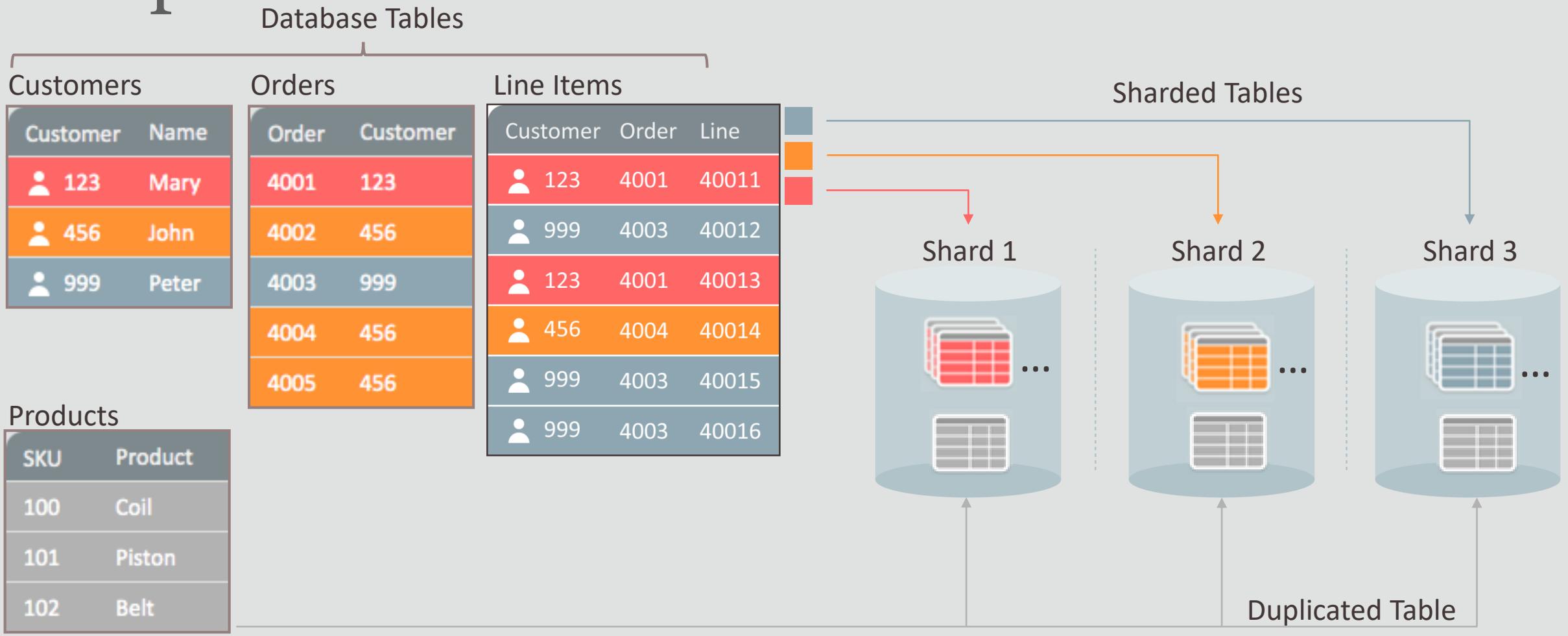
Many of the products evaluated by customers are sharded systems and many lack enterprise grade features like support for complex joins, strict data consistency, security, cross region replication, performance optimizer, backup and recovery, triggers, stored procedures, regular security patches, manageability at scale

Oracle Database Sharding

Best of mature RDBMS capabilities and NoSQL databases

- SQL and all the programmatic interfaces (PL-SQL, OCI, JDBC, etc.) that you expect
- Better consistency than NoSQL databases (strictly consistent within shard)
- Easier application maintenance – schema in database instead of application
- Enterprise features: Advanced Security, RMAN, ASM, Data Guard, GoldenGate, Advanced Compression, Partitioning, etc.
- All the Oracle innovations: high-performance storage engine, SMP scalability, RAC, Exadata, in-memory columnar, online redefinition, JSON document store, etc.
- Leverage in-house and world-wide Oracle DBA skillset
- Enterprise-standard support
- **Plus extreme scalability & availability of NoSQL databases**

Schema Creation – Sharded and Duplicated Tables



Creating a Sharded Table Family with Referential Integrity

Execute DDLs on the Shard Catalog

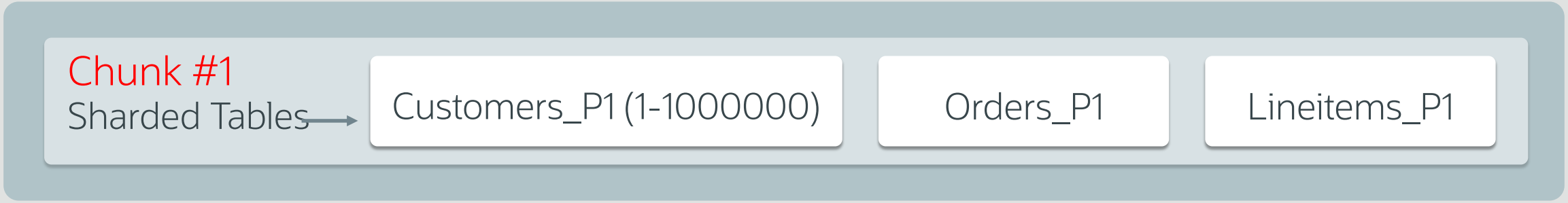
```
CREATE TABLESPACE SET  tbs1 ;

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 ;
```

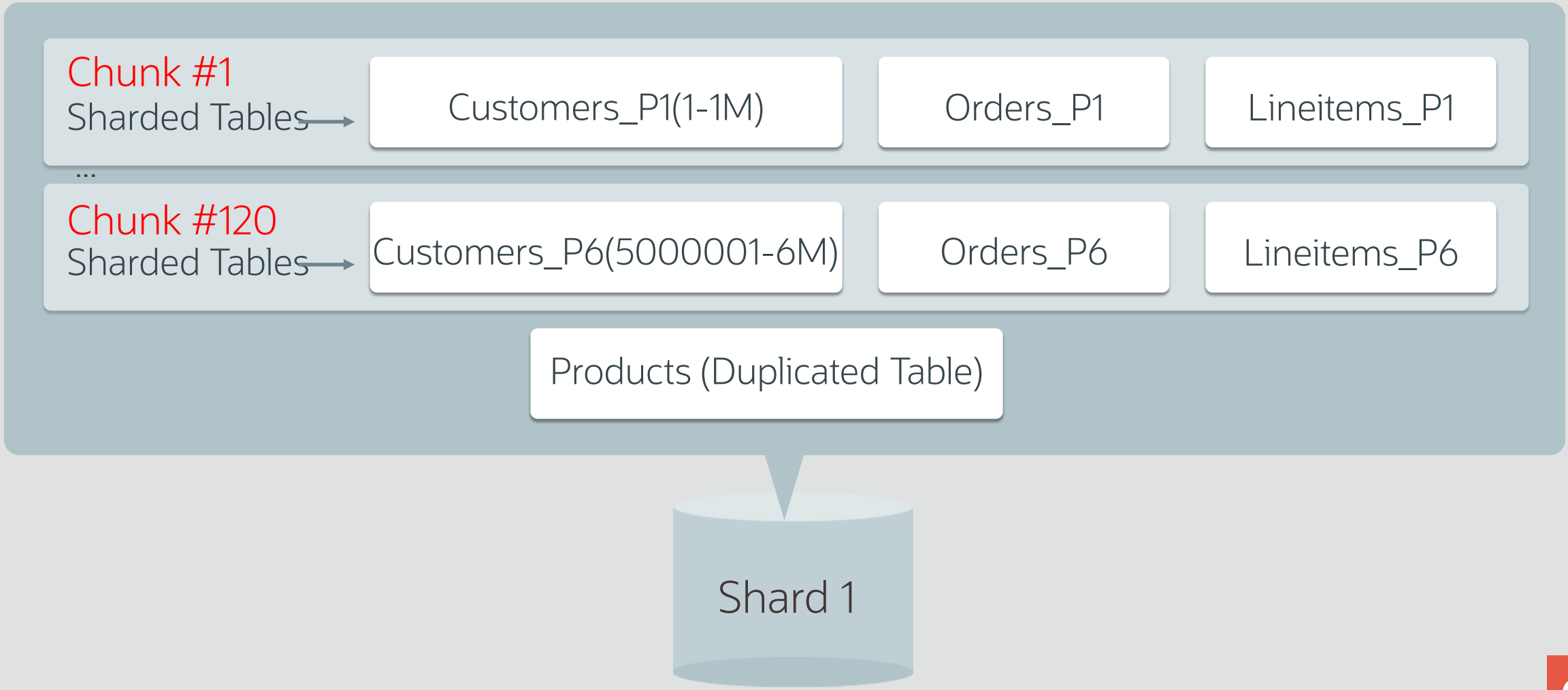
Concept: Chunk



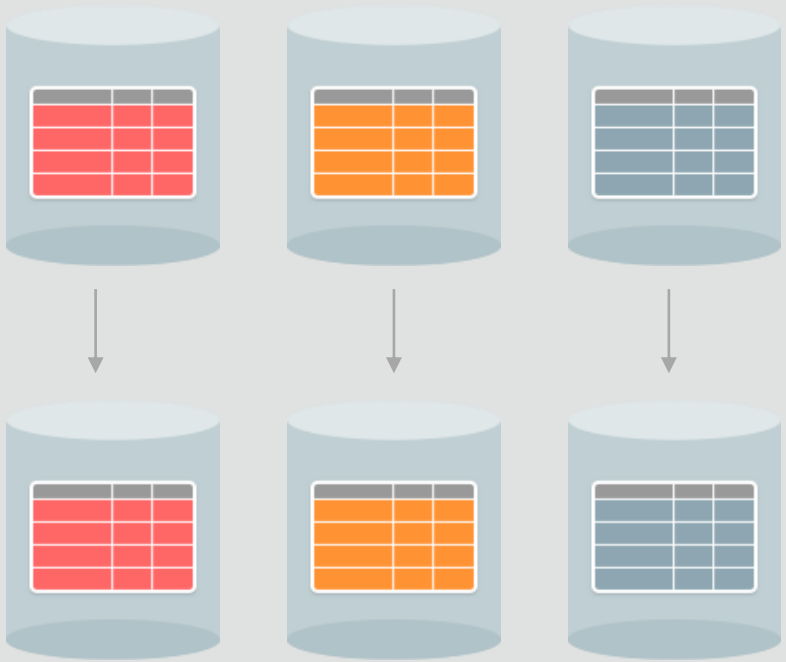
- Group of tablespaces of related partitions of a sharded *table family*
  - Ex: Chunk#1 contains Customers_P1, Orders_P1, LineItems_P1
- All data pertinent to a sharding key resides in a given chunk
  - No need to go to multiple shards
- Unit of data movement for resharding

Contents of a Shard

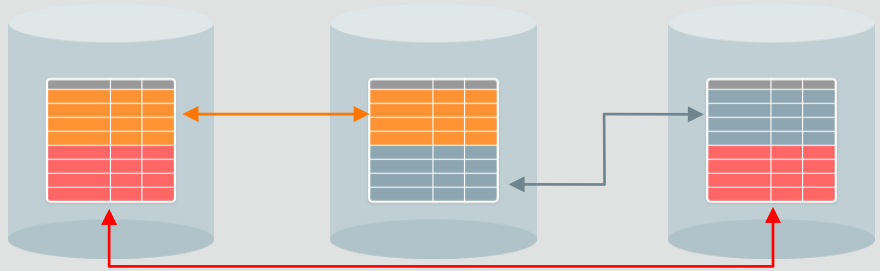
Set of chunks with data from sharded tables + duplicated tables



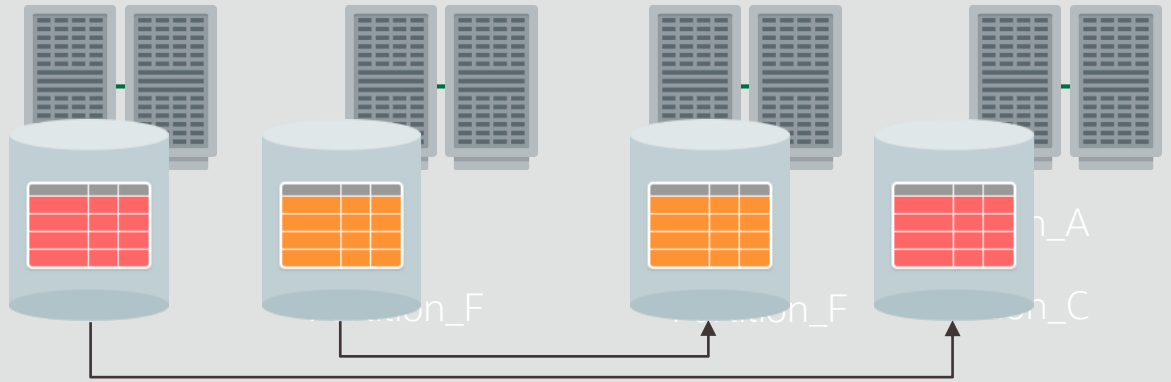
HA Configurations



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

Oracle Sharding Methods

System Managed Sharding

- by Consistent Hash

  Range of hash values assigned to each chunk

User-defined Sharding

- by Range
  - Range of sharding key values assigned to each chunk
- by List
  - Each chunk associated with a list of sharding key values

Composite Sharding

- by Range - Consistent Hash or by List - Consistent Hash

  Two-level sharding, uses two keys

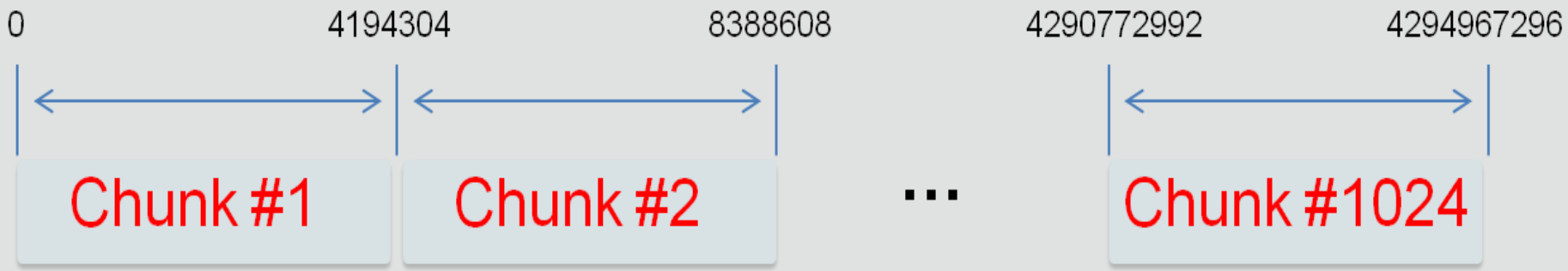
System Managed Sharding

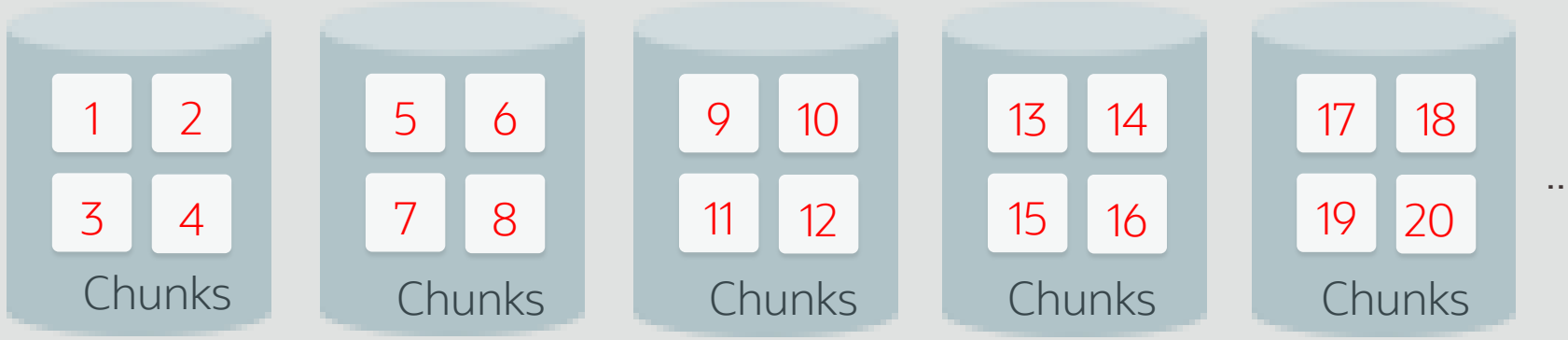
- Based on partitioning by CONSISTENT HASH
- Data is sharded / re-sharded automatically
- Data is evenly distributed across shards
- All shards are managed and replicated as unit
- Many relatively small equally sized chunks

+ Automatic balanced data distribution

- User has no control on location of data

Concept: Consistent Hash





User-defined Sharding

- Based on partitioning by RANGE or LIST
- User specifies mapping of data to shards
- Each shard can have different location, platform, replication topology
- Few large chunks, user-controlled resharding (chunk split and move)

+ Full control on location of data provides:
  - Regulatory compliance
  - Support for hybrid clouds
  - Efficient range queries
  - User knows which data is impacted by failure

- Need to manually maintain balanced data distribution

Composite Sharding

- Combination of user-defined and system-managed sharding
- Set of shards is divided into subsets
- Data is partitioned across subsets by LIST or RANGE
- Within each subset data is partitioned by CONSISTENT HASH

+ Provides benefits of user-defined and system-managed sharding

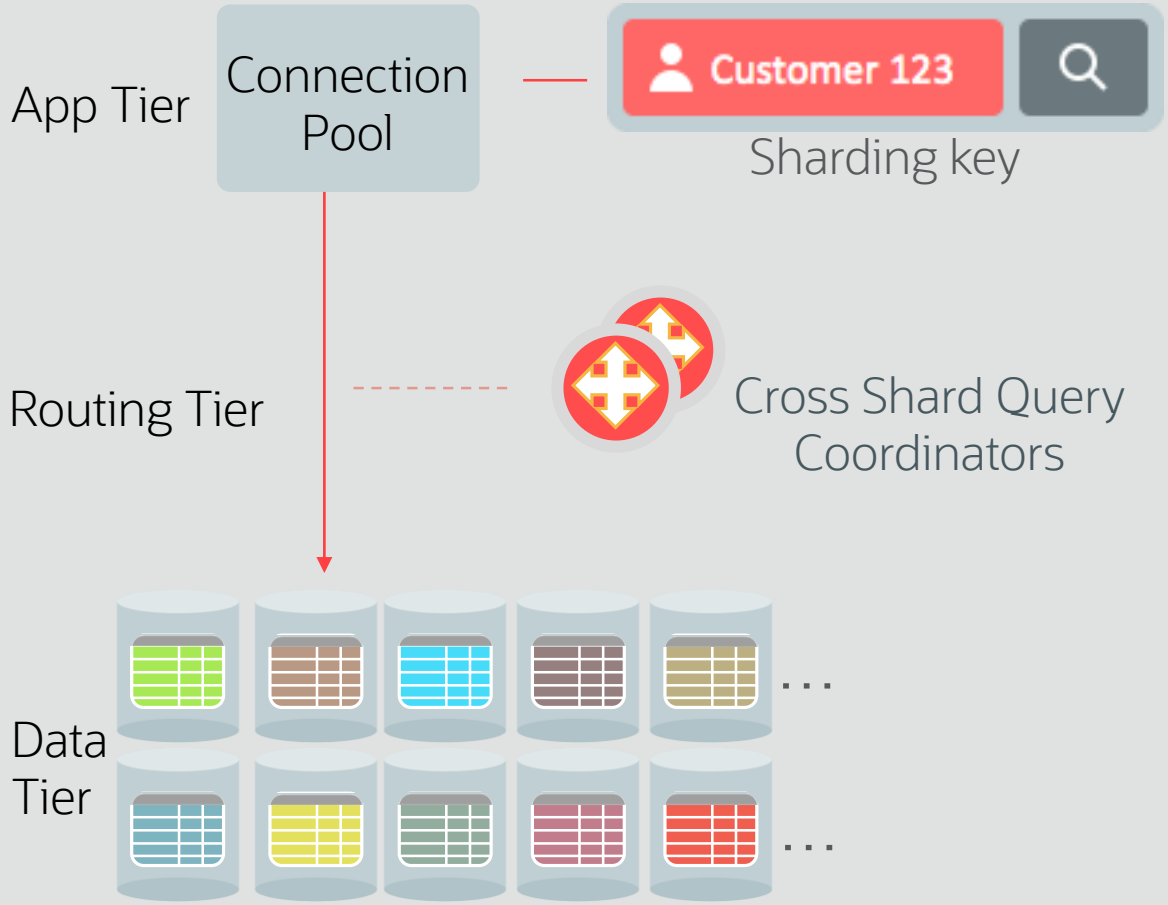
- Requires more complex database schema and two sharding keys

Composite Sharding

Geographic Distribution and Linear Scalability

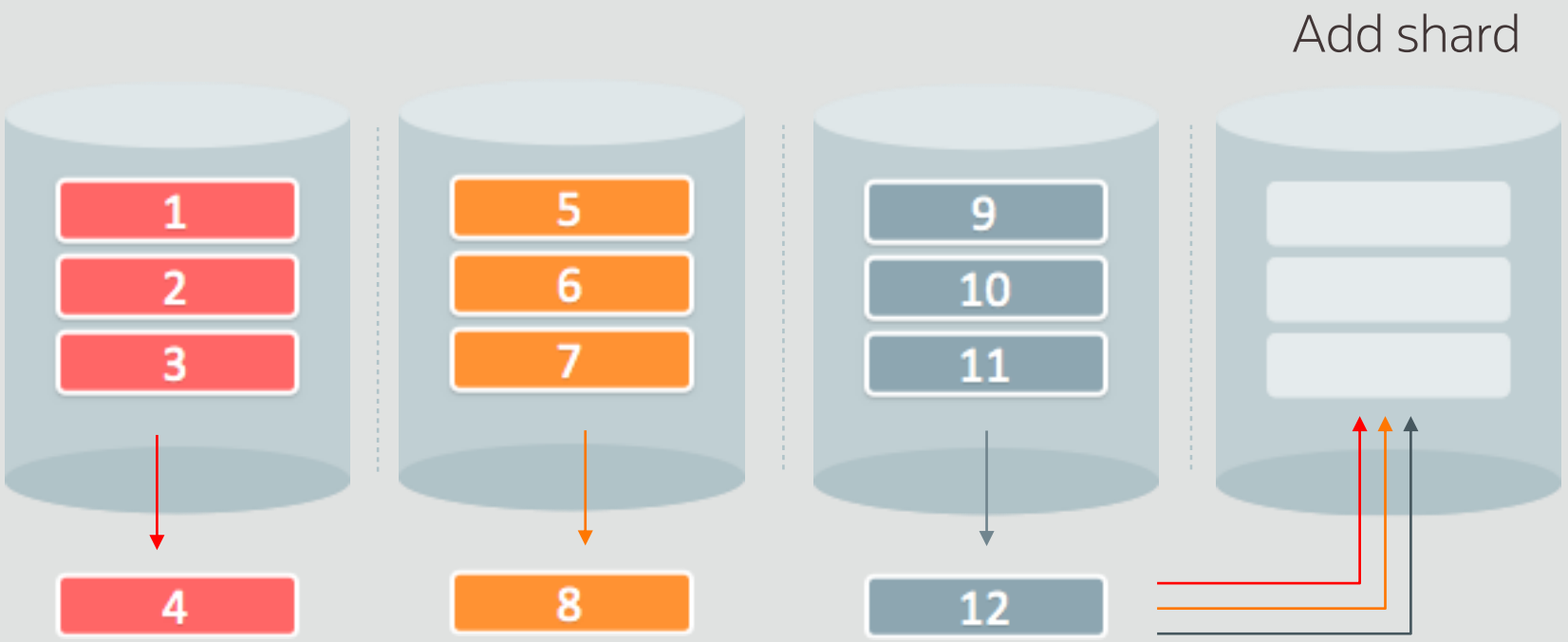
Client Request Flow

- **Client Routing** (JDBC, OCI, UCP, ODP.NET)
  - Direct routing from Connection pools
  - Proxy routing for Multi-shard queries
- **Shard Catalog**
  - Stores SDB metadata
  - Acts as a coordinator for multi-shard queries
  - Contains app gold schema & duplicated tables
- **Shard Director**
  - A global service manager for direct routing of connection requests to shards
  - Publishes run-time SDB topology map, load balancing advisory, FAN events via ONS



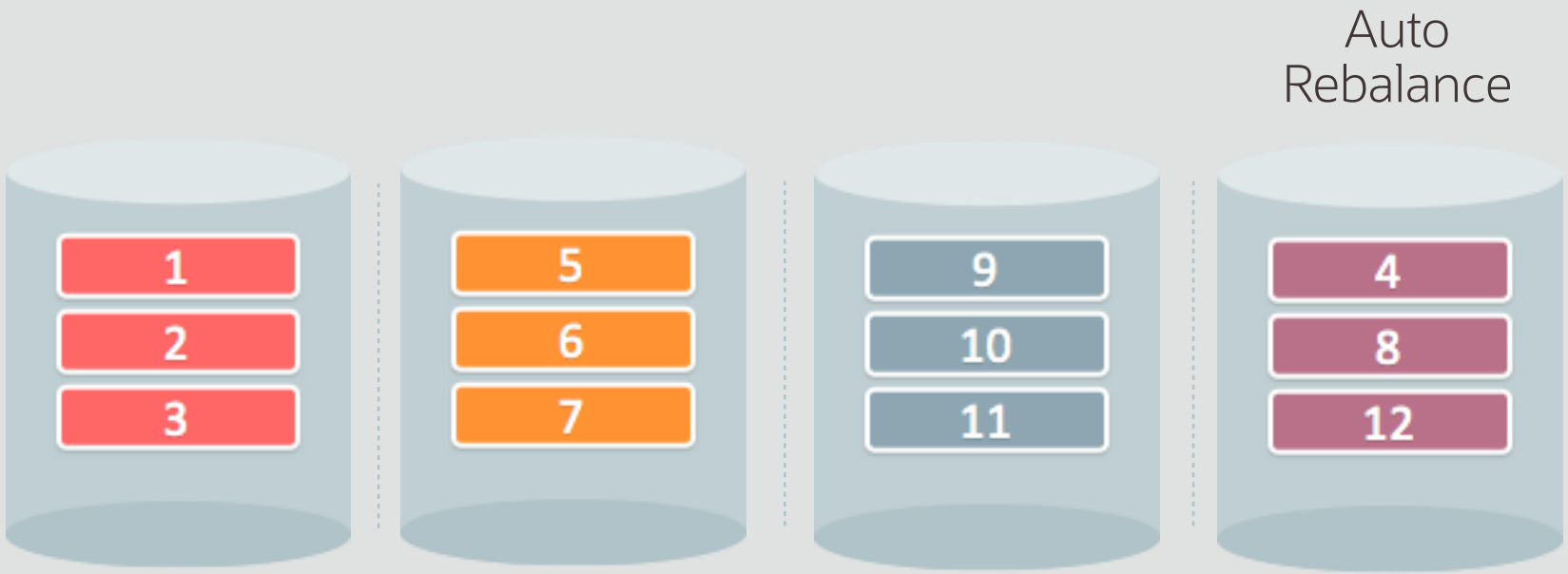
Online Addition and Rebalancing of Shards

- A chunk is a unit of resharding
- Chunk move is initiated automatically or manually (by DBA)
- Uses RMAN Incremental Backup & Transportable Tablespace



Online Addition and Rebalancing of Shards

- A chunk is a unit of resharding
- Chunk move is initiated automatically or manually (by DBA)
- Uses RMAN Incremental Backup & Transportable Tablespace



Automated Patching of SDB

- OPatchauto supports
  - All sharding schemes and replication methods
  - Single instance and clustered databases (also handles Grid Infrastructure)
- To patch a sharded database :

<CATALOG_DB_HOME>/OPatch/opatchauto apply <patch loc> -sdb -wallet <wallet file loc> -sid <sid of shardcat> -port <shardcat port>

- For Data Guard
  - OPatchauto supports rolling mode (default: parallel mode)
    - For a given configuration, standbys are patched first followed by primary

shardcat

Page Refreshed Sep 17, 2016 12:12:05 AM GMT

Summary

Sharded Database Name	orasdb
Configuration Name	oradbcloud
Catalog Database	shardcat
Catalog Version	12.2.0.1.0
Sharding Type	System-managed
Replication Type	Data Guard
Shard Directors	1 (↑1)
Master Shard Director	sharddirector1

Shard Load Map

Total Active Sessions : 0.05

Instance: shardcat
Total Active Load: 0.020 active sessions

Load Summary
CPU: 0.013
IO: 0.000
WAIT: 0.007

View Level : Database Instance

Members

Shardspaces | Shardgroups | Shard Directors | Shards

Name	Shardspace	Shardgroup	Data Guard Role	Region
sh1	shardspaceora	shgrp1	Primary	availability_domain1
sh1s1	shardspaceora	shgrp2	Active Standby	availability_domain2
sh2	shardspaceora	shgrp1	Primary	availability_domain1
sh2s1	shardspaceora	shgrp2	Active Standby	availability_domain2

Services

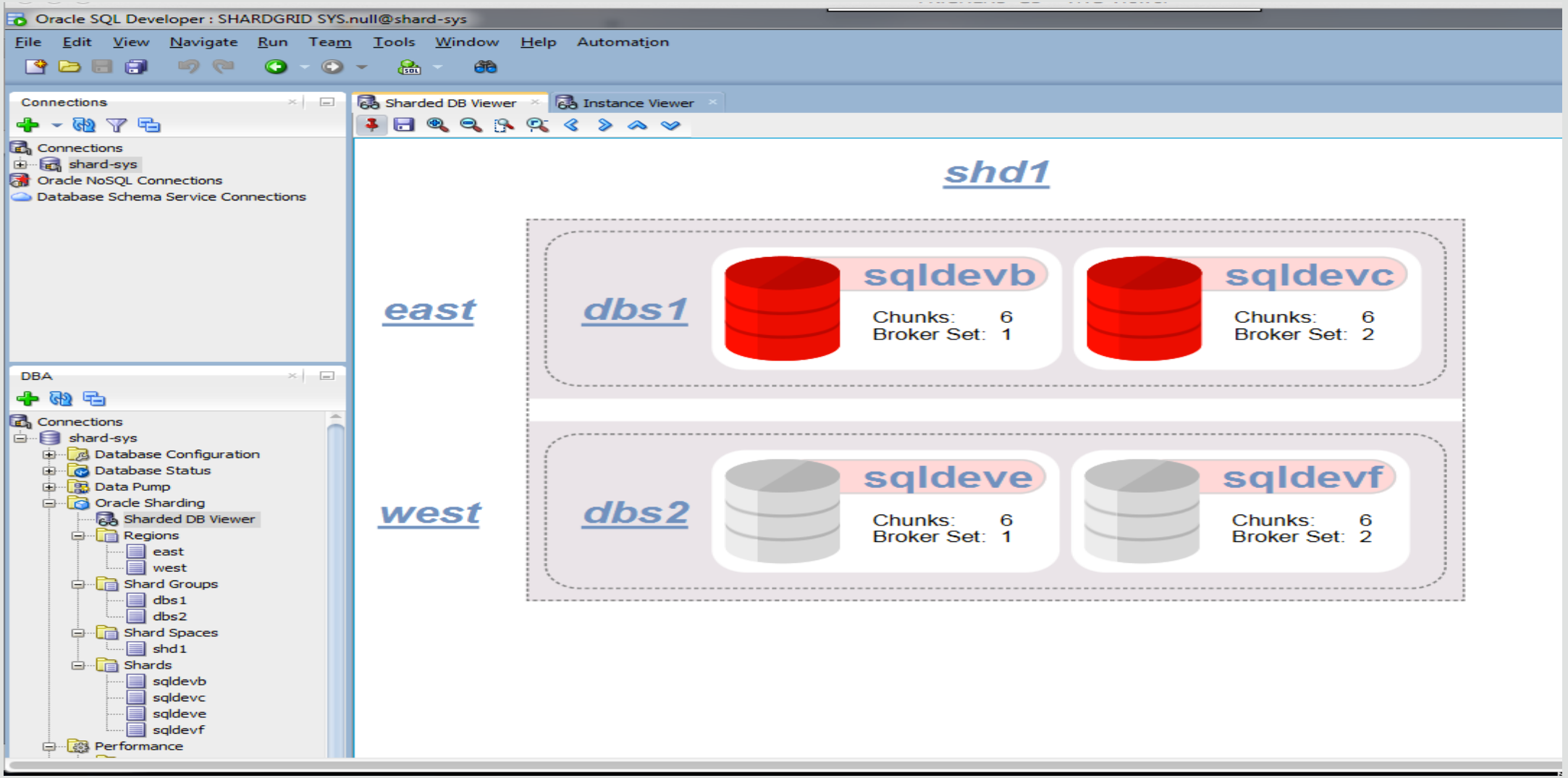
Name	Status	Data Guard Role
No services found.		

Incidents

View ▾ Target Local target and Related targets ▾ Category All ▾ 0 5 »

Summary	Target	Severity	Status	Escalation Level	Type	Time Since Last Update
Problem: KUP-600			New	-	Problem	0 days 0 hours
Problem: KUP-600			New	-	Problem	0 days 2 hours
The Data Guard fast-start failover observer status is Error Fast-Start Failover observer is no longer observing this database.			New	-	Incident	0 days 2 hours
Checker run found 1 new persistent data failures.			New	-	Incident	0 days 4 hours

SQL Developer Integration with Sharded Databases



Announcing | Deployment Automation

- Sharding Advisor
  - Tool to advise on schema migration from non-sharded databases to Sharding
  - Key goals are to maximize parallelism (spread query execution across all shards), minimize cross shard operations and minimize duplicate data
  - Analyze existing database schema, user workloads and makes recommendations like which tables to Shard, which column to use as Sharding Key, Sharding Method to use, which tables to duplicate
- Deployment Automation with Terraform, Kubernetes and Ansible
  - Simple input file describing deployment topology
  - Run from one of the host for distributed setup
  - Reentrant / Resume/Cleanup in case of errors
  - Scale out sharding components independently
  - Terraform deployment download link here

Shard Advisor Sample Output

rank	tname	type	tlevel	parent	shardBy	cols	size	unenforced
1	CUSTOMER	S	1		HASH	C_CUSTKEY	44	CUSTOMERFK
1	ORDERS	S	2	CUSTOMER	REFERENCE	ORDERSFK	289	
1	LINEITEM	S	3	ORDERS	REFERENCE	LINEITEMFK1	1472	LINEITEMFK2
1	NATION	D			NONE		1	
1	PART	D			NONE		43945	
1	PARTSUPP	D			NONE		23340	
1	REGION	D			NONE		1	
1	SUPPLIER	D			NONE		260	

Terraform Script Input File

```
shards = {
  "shard-1" = {
      host = "den02ffv"
      port = "1521"
      sid = "sh1"
      globalDBName = "sh1"
      shard_group = "primary_shardgroup"
  },
  "shard-2" = {
      host = "den02ffw"
      port = "1521"
      sid = "sh2"
      globalDBName = "sh2"
      shard_group = "primary_shardgroup"
  }
```

Oracle Sharding | 19c Features

- Sharding of multiple PDBs to allow consolidation and fault isolation
  - A CDB can now support multiple PDB shards which helps with consolidation
  - Different PDBs from same sharded databases are on different CDBs to provide for fault isolations
- Scalable multi shard query coordinators for reporting and analytical workloads
  - Shard catalog's Active Data Guard standbys can act as multi-shard query coordinators
- Improve resource utilization by allowing sharding different tables by different keys in the same database
  - Allows an sharded databases to support multiple table families, each of which can be sharded with a different sharding key
- High speed data ingest
  - Data is split and loaded directly to shards in parallel using direct path write

20c | Oracle Sharding

- Federated Sharding
  - Allows queries across existing similar databases in multiple geo-regions
- Multi Shard DML and Query Enhancements
  - Updates in parallel across all Shards
  - Cross Shard Query execution continuation in case of a Shard failover
- Automatic Identification of Sharding Key
  - Simplifies application design and maintenance
- Deployment Automation Enhancement
  - Schema design advise for moving from non-Sharded to Sharded deployment with Sharding Advisor
  - Deployment Automation with Terraform scripts
- Native Support for Databases in Persistent Memory

Use Case: Dyn

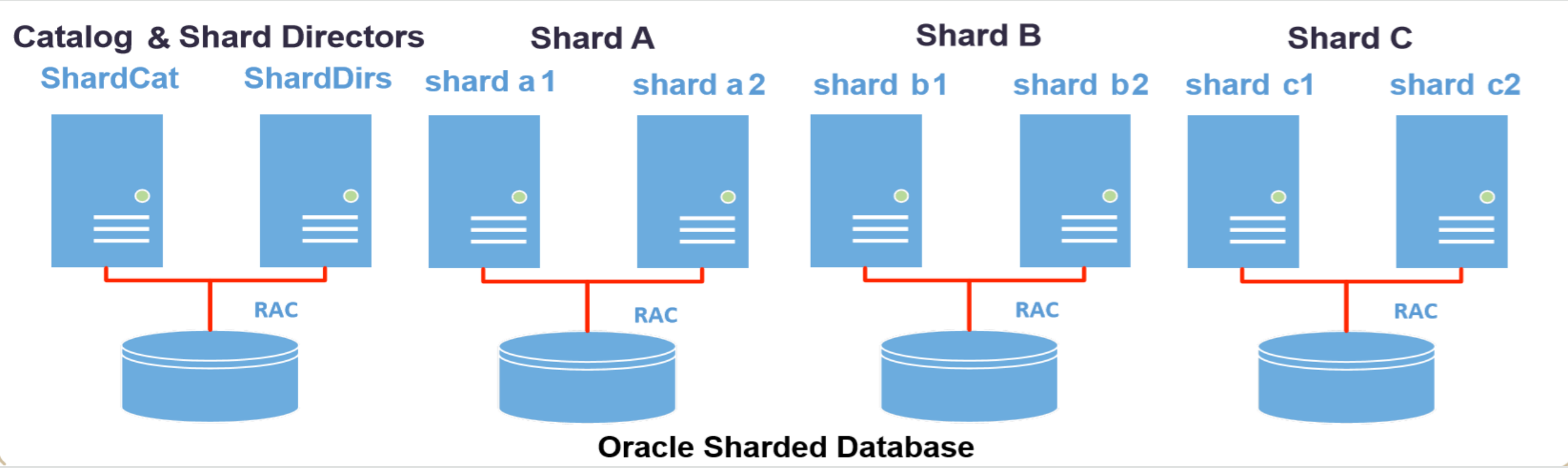
- Proven linear scalability
- Ingest speeds scale with number of shards
- Constant query time even as we grew size of dataset
- Geo-distributed to be close to our customers

Shards are replicated across 2 different Availability Domains for availability and disaster recovery

Utilized powerful Bare Metal Cloud servers (36 OCPUs, 512 GB memory, 12.8 TB local NVMe SSD storage)

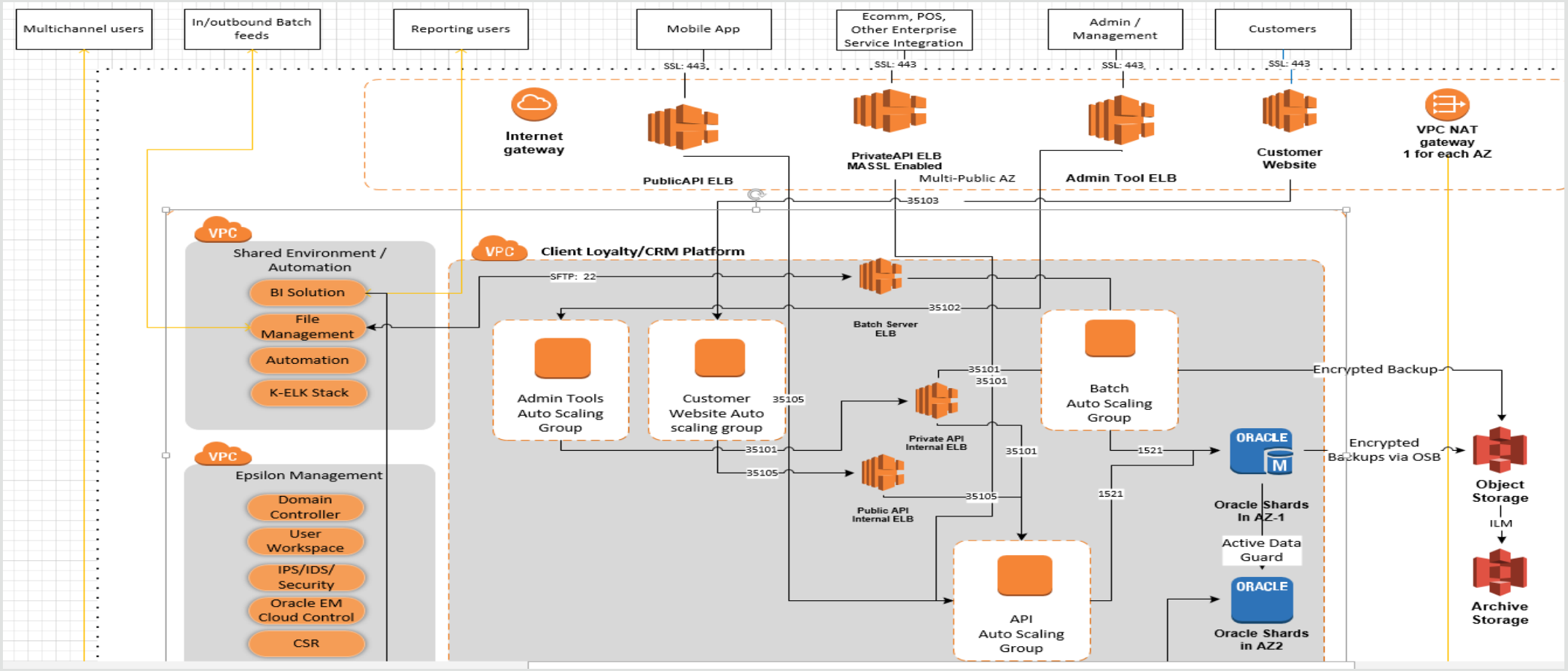
Use Case: China Telecom's WeChat IoT Application

- Current Oracle 12.2 sharded environment has 8 database servers
- Create 4 independent databases: Shard Catalog and Shards across 8 nodes
- Used 2-node RAC at shard-level



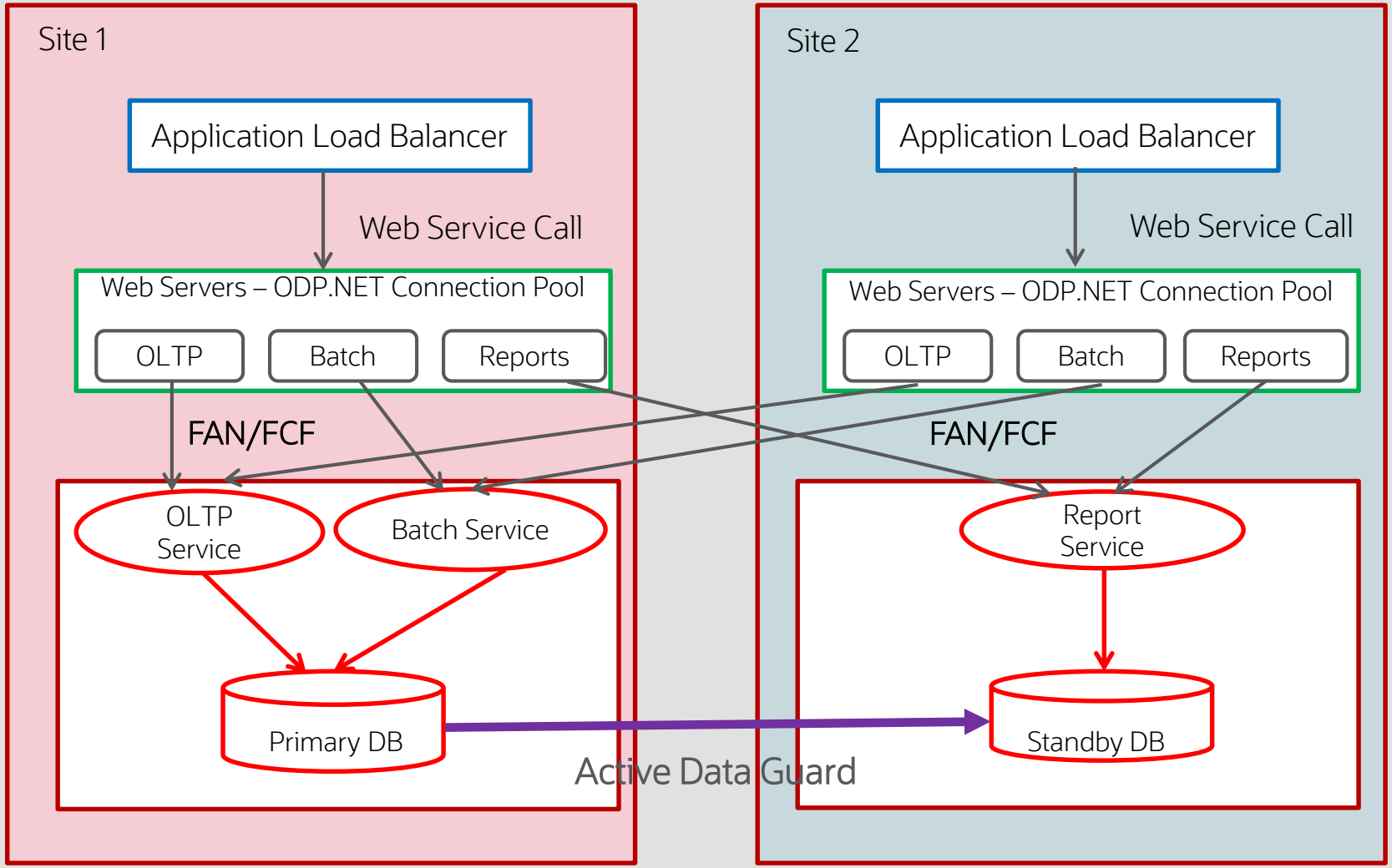
Use Case: Epsilon

OLTP System Deployment Architecture at Public Cloud



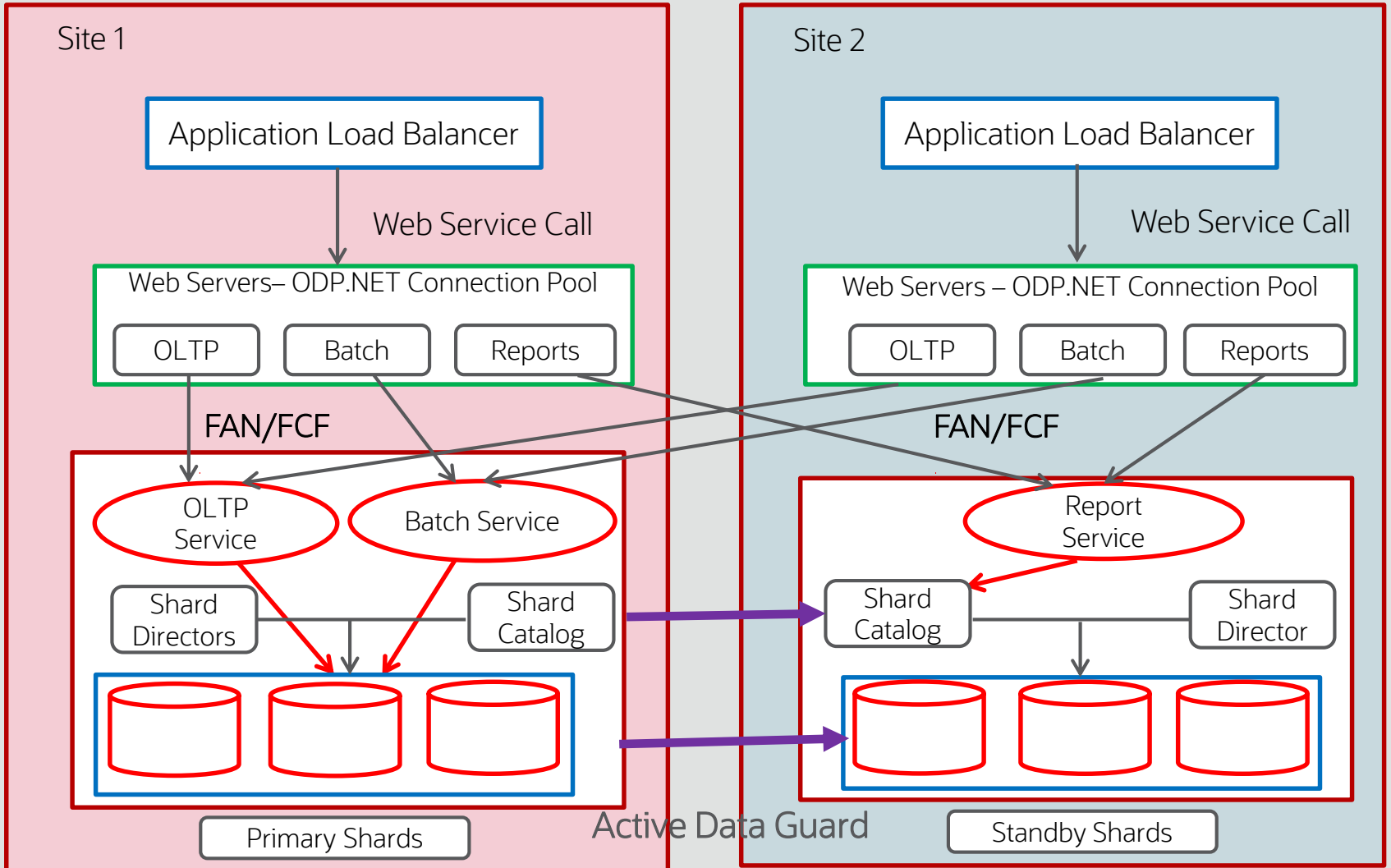
Use Case: Epsilon

Application Service Placement : Current State



Use Case: Epsilon

Application Service Placement : Target State with Sharding



Oracle Sharding | Resources

ORACLE TECHNOLOGY NETWORK

https://www.oracle.com/goto/oraclesharding

ORACLE Blogs

https://blogs.oracle.com/database/

Oracle Maximum Availability Architecture

http://www.oracle.com/goto/maa

https://github.com/oracle/db-sharding

Product Documentation

https://docs.oracle.com/en/database/oracle/oracle-database/19/shard/index.html

Q & A