FROM INTRODUCTION TO PRACTICE

Lesson 1: OceanBase General Introduction

Zhongyan Feng

OceanBase OpenSource Team Leader

Contents

01

OceanBase Brief Introduction

OceanBase – The only Database which break the world record in TPC-C and TPC-H

TPC Transaction Processing Performance Council

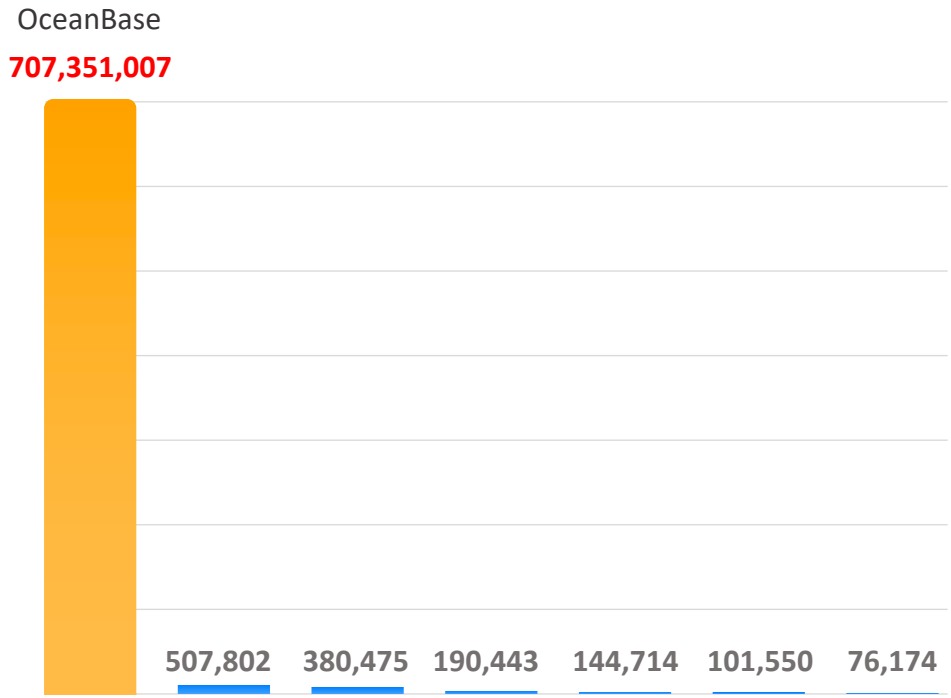
- **TPC-C No.2** official world record

TPC-C - Advanced Filtered and Sorted Results

Sponsor	System	Performance (tpmC)	Price/TpmC	System Availability
ANT FINANCIAL	Alibaba Cloud Elastic Compute Service Cluster	707,351,007	3.98 CNY	6/8/2020
ANT FINANCIAL	Alibaba Cloud Elastic Compute Service Cluster	60,880,800	6.25 CNY	10/2/2019
Oracle	SPARC SuperCluster with T3-4 Servers	30,249,688	1.01 USD	6/1/2011
IBM	IBM Power 780 Server Model 9179-MHB	10,366,254	1.38 USD	10/13/2010
Oracle	SPARC T5-8 Server	8,552,523	0.55 USD	9/25/2013
Oracle	Sun SPARC Enterprise T5440 Server Cluster	7,646,486	2.36 USD	3/19/2010
IBM	IBM Power 595 Server Model 9119-FHA	6,085,166	2.81 USD	12/10/2008
Bull	Bull Escala PL6460R	6,085,166	2.81 USD	12/15/2008
Oracle	Sun Server X2-8	5,055,888	0.89 USD	7/10/2012

- 20 times faster than Oracle

- **TPC-H No.3** official world record (30TB dataset)
- On May 19, 2021, 15.26 million QphH@30000GB



* Source: tpc.org

More than 1000 Customer's Choice.

Financial

Government Telecom

Internet

Manufacturer

OCEANBASE

OceanBase @ Ant Group

“Double Eleven” Performance: 61M QPS

Peaks 544,000 (Alipay Double Eleven) vs Peaks 64,000 (Visa)

- Trade
- Payment
- Accounting
- CIF
- Promotion
- Real time data

61 millions
Responses per second
Peak processing power

> 200
Nodes in one cluster
Cluster Size

> 6
PB data
Data size in one instance

> 320 billions
Billions of rows
One single table size

RPO = 0
RTO < 8
Seconds
Disaster Tolerance

*Statistics above are from real production in Alipay

OCEANBASE

OceanBase : The Open Source Distributed Database for Mission-critical Workloads at Any Scale

1.0

Firmly embrace the distributed architecture

2010

Product initiation

淘宝 Taobao

2013

Internal customers

TMALL天猫

淘宝 Taobao

2014

Support core transactions

支付宝

2.0

Native distributed SQL database

2016

Full service coverage

余额宝

芝麻信用

网商银行

2017

External customers

四川省农村信用社

顺德农商银行

南京银行 BANK OF NANJING

2019

Break the TPC-C world record

PICC 中国人民保险

招商证券 China Merchants Securities

常熟农商银行

西安银行 BANK OF XI'AN

3.0

Unified engine, hybrid deployment

2020

Commercialization

中国工商银行 INDUSTRIAL AND COMMERCIAL BANK OF CHINA

中国移动 China Mobile

中国石化 SINOPEC CORP.

中华保险 CHINA INSURANCE

2021

Massive adoption

江西省人力资源和社会保障厅 Jiangxi Province Human Resources And Social Security Department

交通银行 BANK OF COMMUNICATIONS

国家电网 STATE GRID

Ctrip 携程

DANA

中国人寿 CHINA LIFE

云南红塔银行 YUNNAN HONGTA BANK

4.0

Standalone & distributed integrated architecture

2022 ~

Provide cloud service & go global

理想

致欧

有赞

TELECOM ITALIA

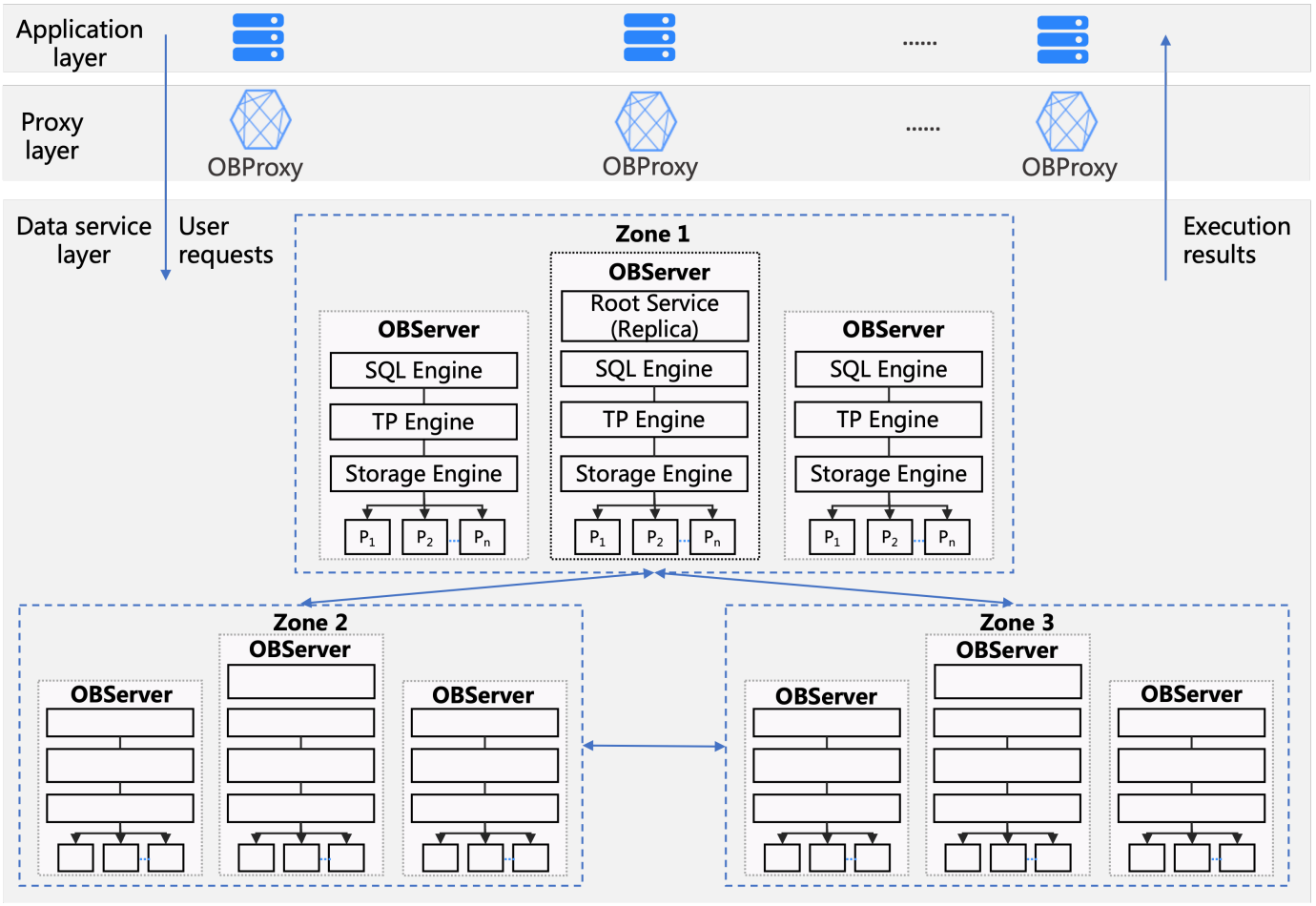
GCash

easypaisa

02

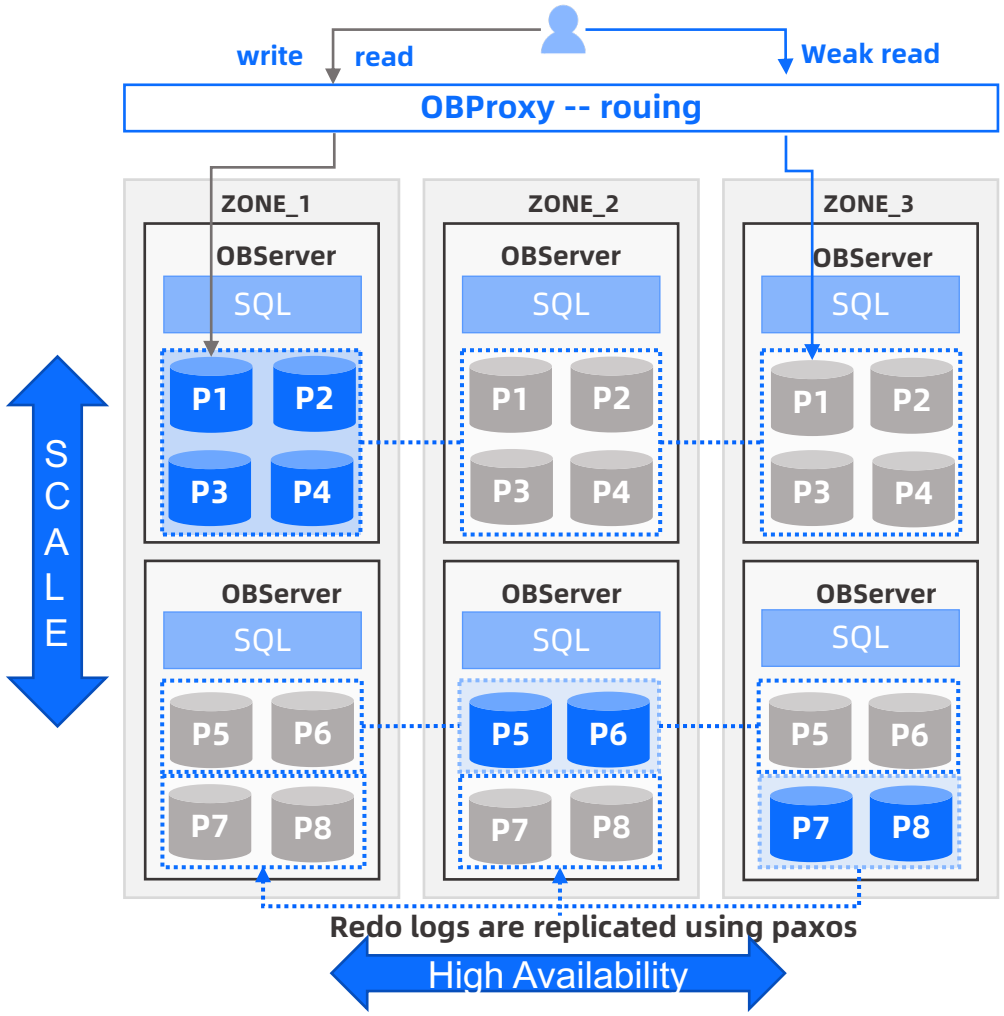
Architecture and Conception

Architecture Overview



- No single failure point, every node are equal
- All OBServers can read and write
- Cluster can be deployed on Private Cloud/Public Cloud/Hybrid Cloud
- ObProxy is used to route requests
- LSM-tree like storaged engine
- SQL Engine: local, remote and distributed execution

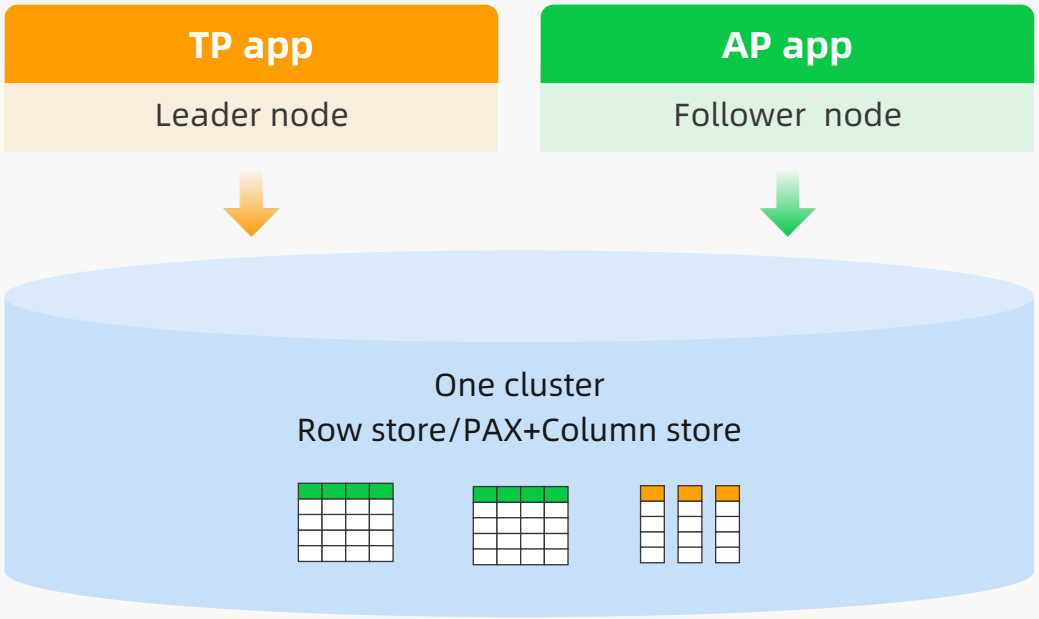
Architecture Overview (2)



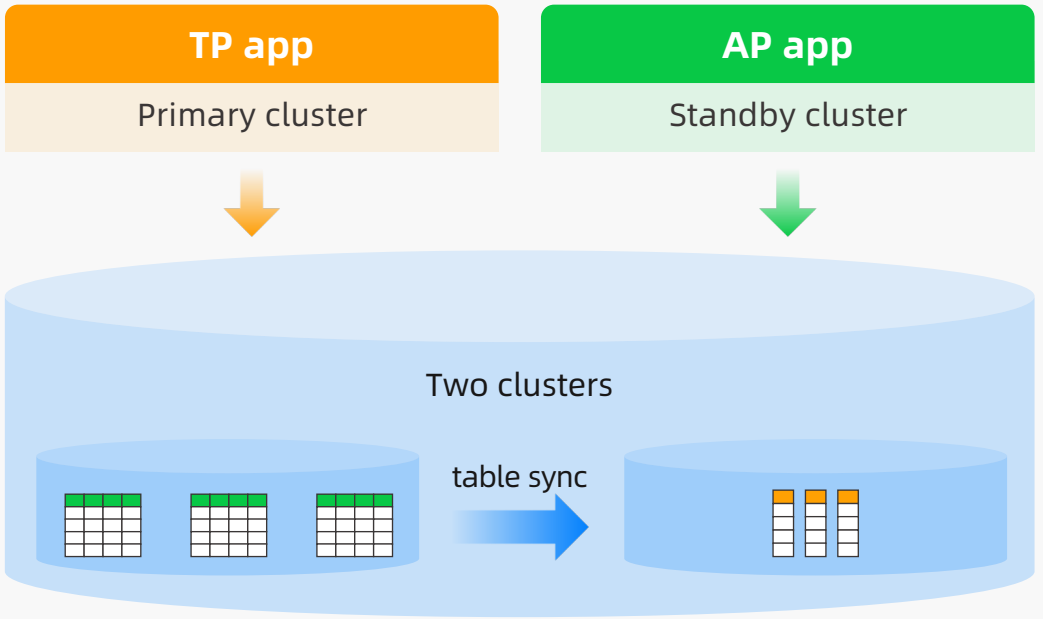
- Cluster is composed of zones
- One zone means one replica
- One zone can contains one or
- Data are distributed by Partitions
- The same partition in different zones form Paxos group
- One partition will be elected as leader in one Paxos Group.
- The leader will provide read/write service.
- Redo logs are replicated using Paxos
- Transactions across servers are executed using 2PC
- Transactions on one node, even multiple partitions, local transaction in single node, reduce distributed transaction

One Storage Engine support both Row and Column Store

One unified system for both TP & AP

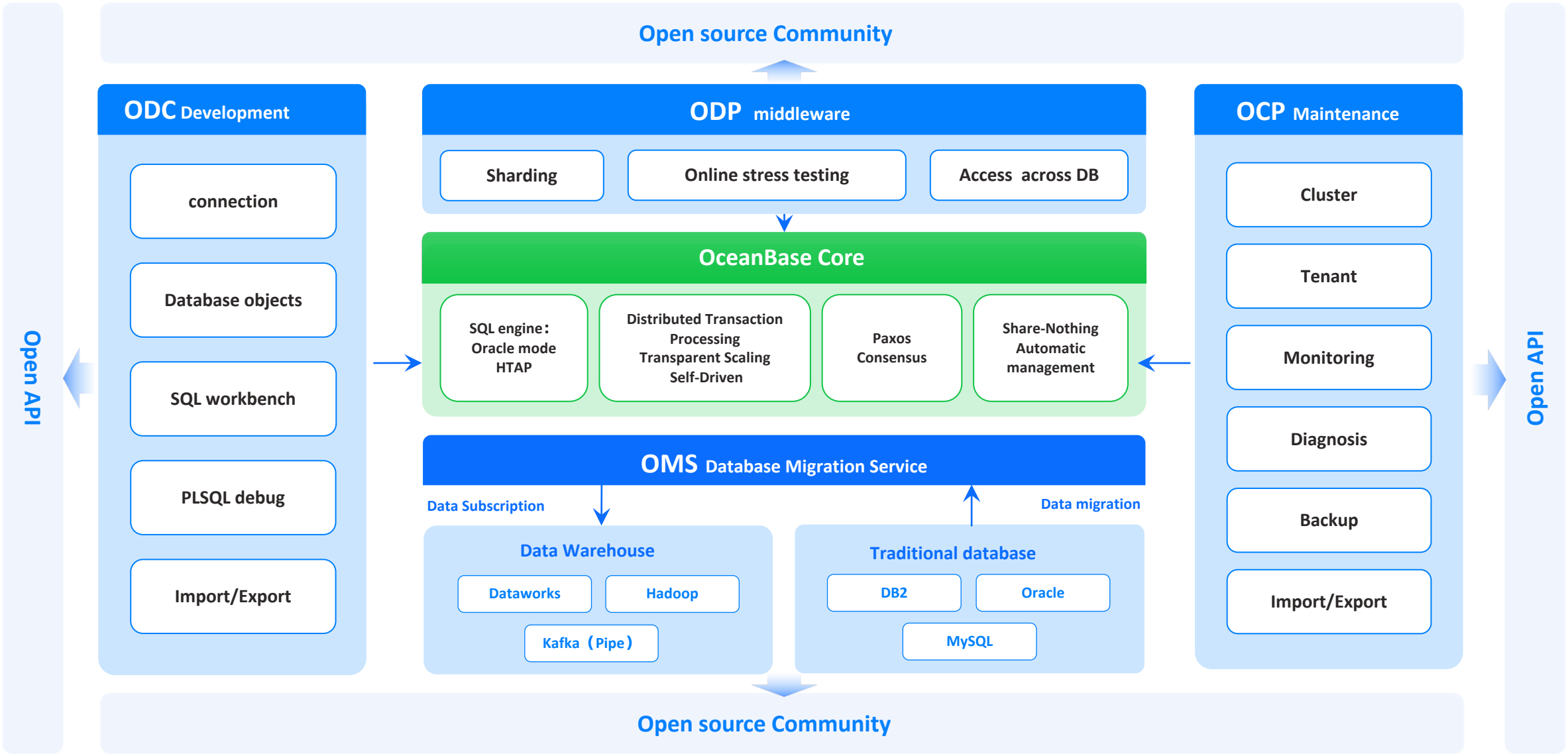


- Low cost, high cost-effectiveness
- Millisecond-level latency and consistency between leader and follower nodes



- High cost, low cost-effectiveness
- Significant latency and data inconsistency between primary and standby clusters

Product Family 1 + 4 Full Stack Solutions



03

Key Features

Features

- High Availability
- Scalable
- Easy Operation
- High Performance
- Cost Effective
- Compatible with MySQL

How to provide 7 x 24 Service?

Key Feature: High availability

High Availability

Flexible DR Tolerance

3 replica in 1 IDC

3 IDCs in 1 city

3 IDCs in 2 cities

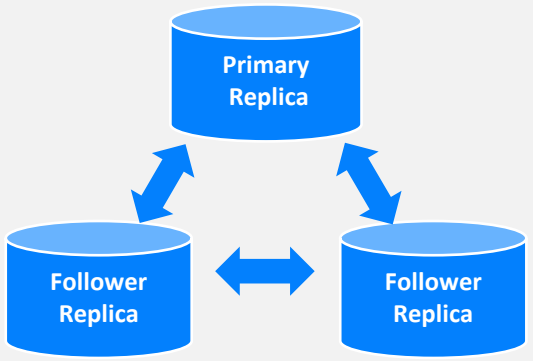
Dual IDCs in master/standby mode

5 IDCs in 3 cities

RPO=0 RTO<8s

Monolithic database

- **Data Loss:** Cannot guarantee consistency in case of failure
- **FO cost:** High maintenance cost for cold stand-by



OceanBase

- **Zero data loss:** Consensus protocol based on Multi Paxos
- **Unattended HA:** Available when a minority of nodes fails RPO = 0, RTO **<8**s
- **Peer Node:** All replicas are active

Standby cluster/tenant

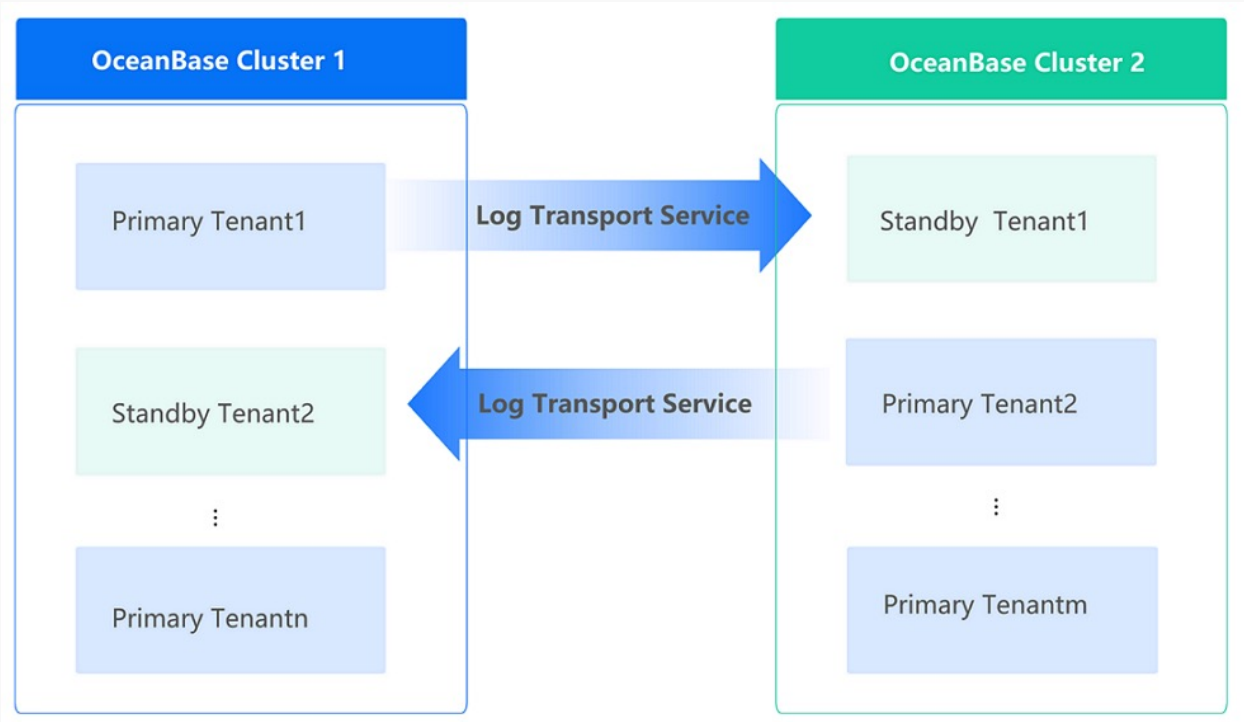
Similar like primary/slave cluster

User scenarios

- Single replica with high available
- Long network latency such as across city

Advantage

- Simple
  - Implementation by kernel, without any other component
  - The primary cluster hardware can be different from that of the slave cluster.
- More robust
  - Sync DDL automatically
  - Easily switchover/failover, support RTO < 8, RPO=0
- Performance better
  - Sync physical modification log(clog)
  - compress

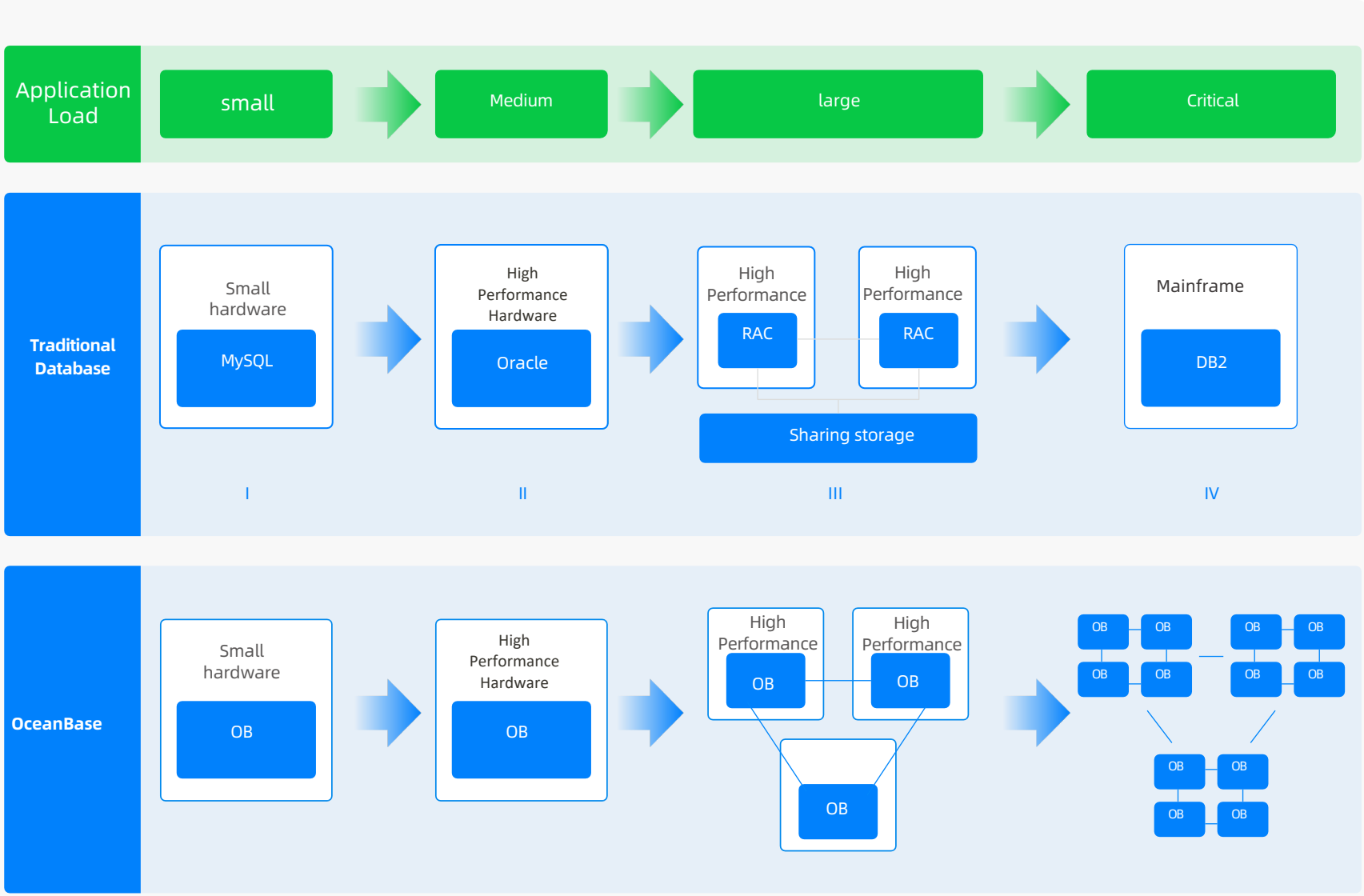


architecture

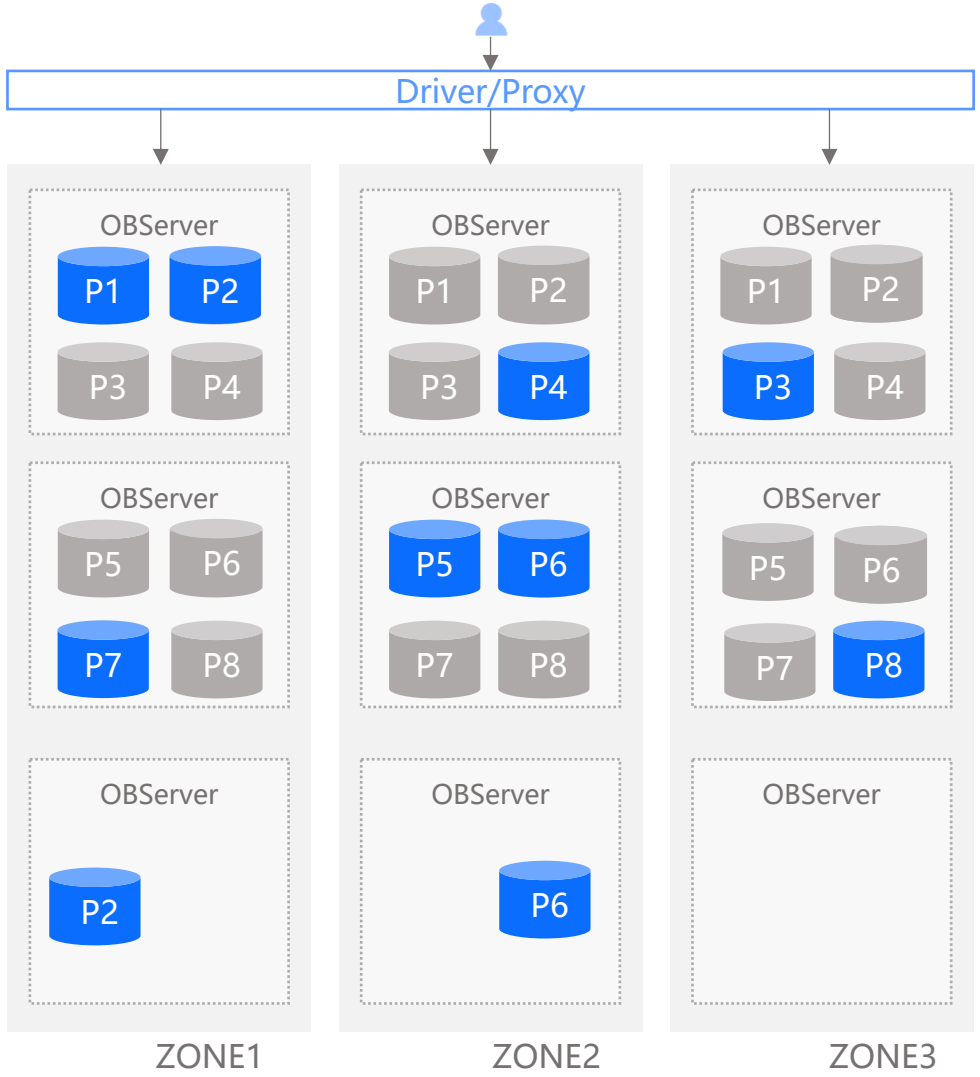
How to scale ?

Key Feature: Scalable

Flexible deployment - different size different architecture



Transparent scalability for applications - vertical scaling



- Transparent for application
- Add machine to cluster
- Modify the number of tenant unit

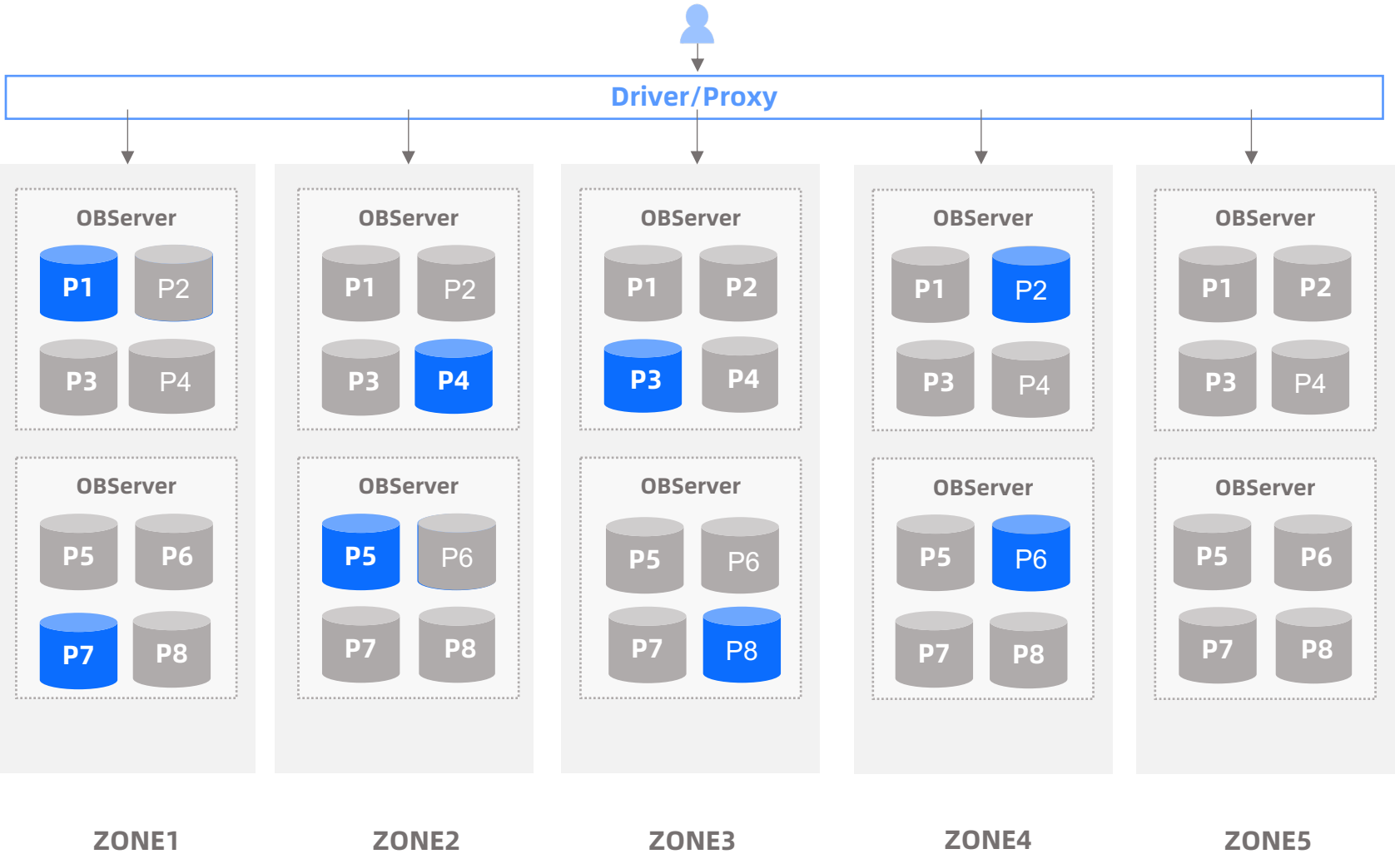
✓ Auto/manual rebalance

- ✓ Sync speed under control
- ✓ 70MB/s per thread, default 500MB/s

P Leader

P Follower

Transparent scalability for applications - horizontal scaling



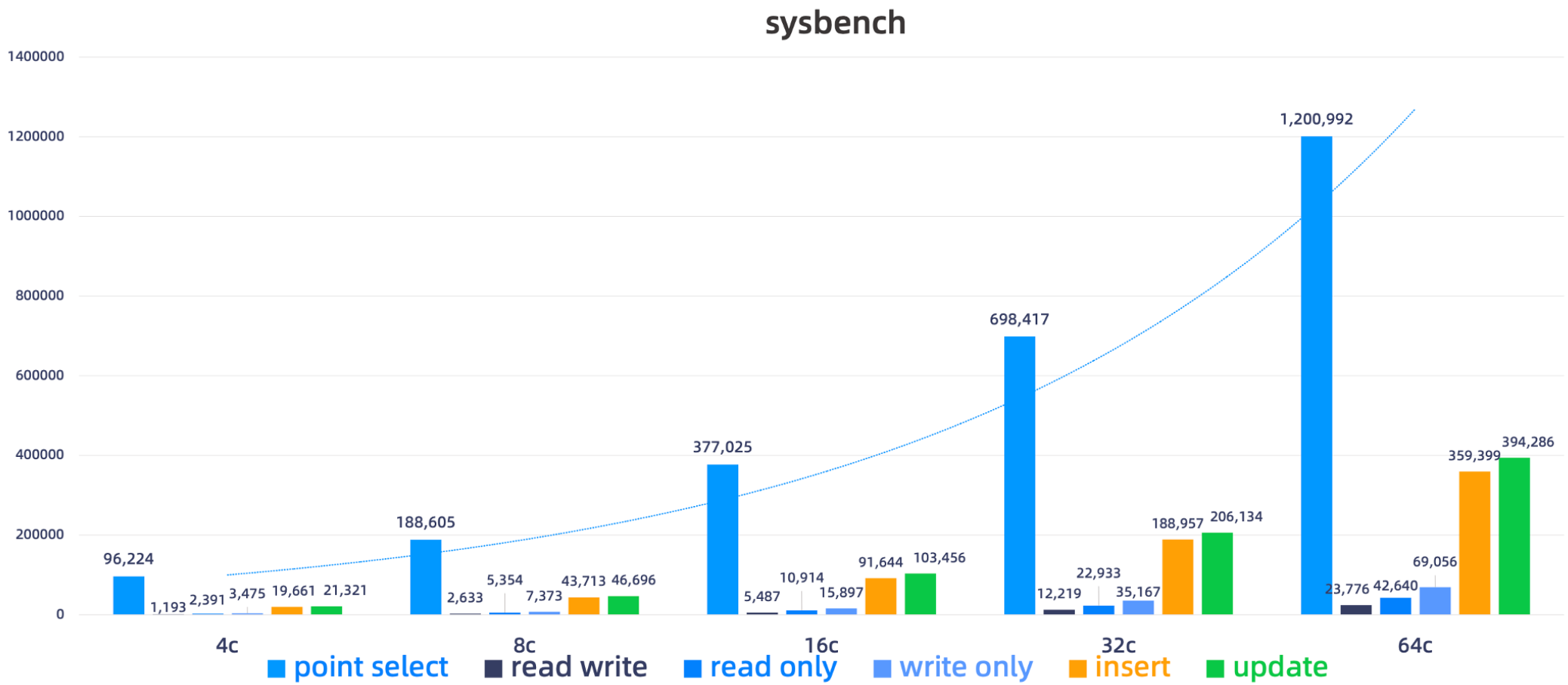
- Transparent for application
- Add zone to cluster
- ✓ Auto/manual rebalance
  - ✓ Sync speed under control
  - ✓ 70MB/s per thread, default 500MB/s
- Auto select Leader
  - ✓ User can set zone priority

P Leader

P Follower

OCEANBASE OpenInfra FOUNDATION

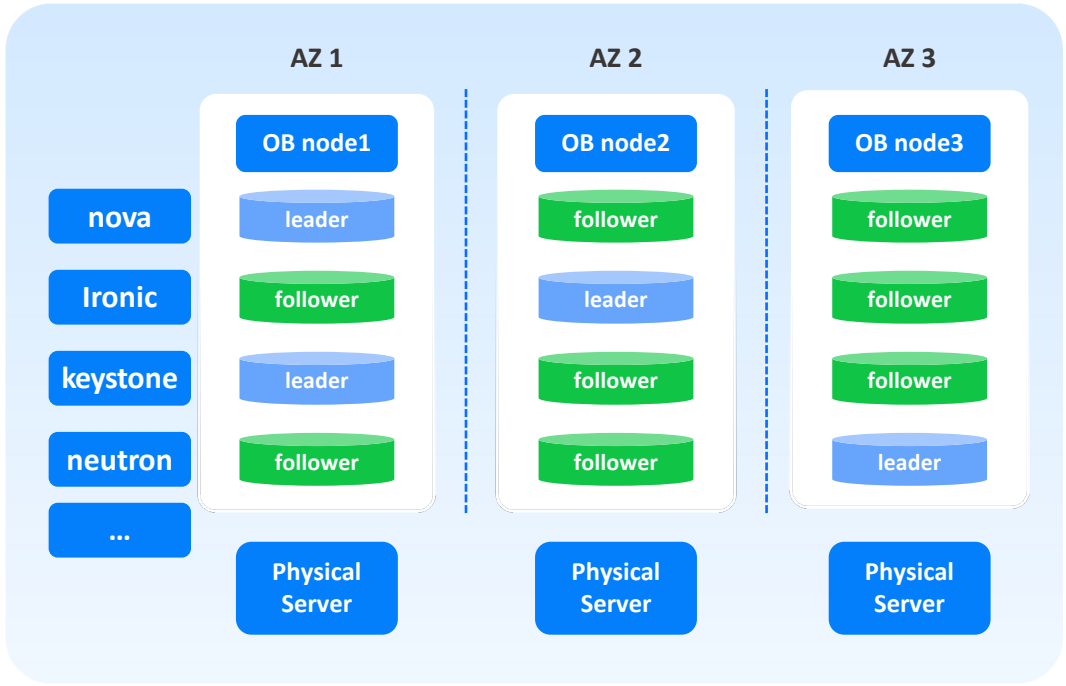
linear scalability



How to simplify maintenance task?

Key Feature: Easy Operation

Multi tenant – One Cluster for all user, but one tenant per module



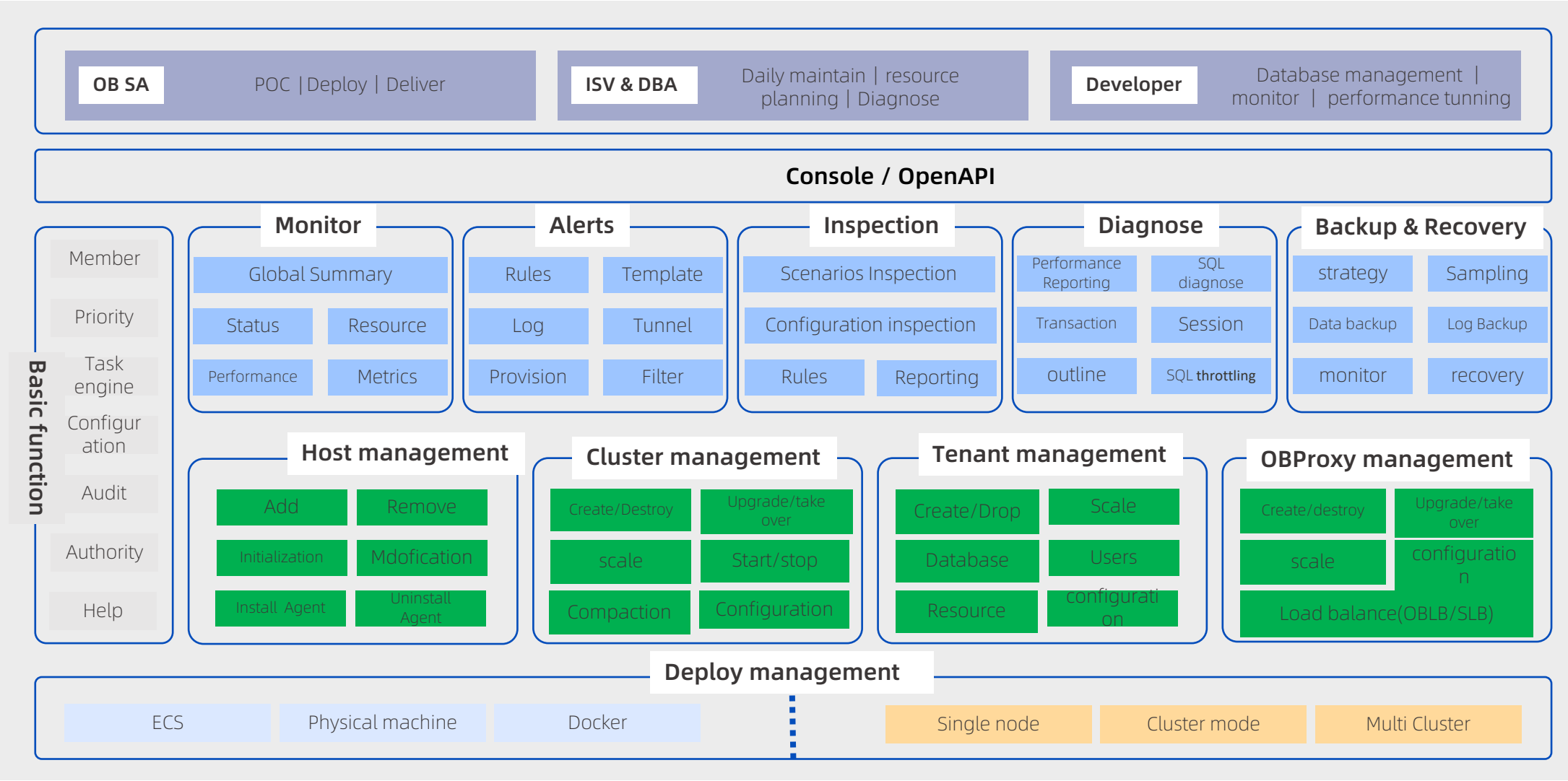
	Zone	sqa.eu95_1		sqa.eu95_2		sqa.eu95_3	
▽租户名	副本大小	100.81.122.74	100.81.122.80	100.81.122.85	11.161.17.95	11.161.17.69	11.161.17.86
agds_sit10_2085	7.7G	2-4C | 10.0G		2-4C | 10.0G		2-4C | 10.0G	
asttship_1965	64.9G	2-5C | 3.0-5.0G		2-5C | 3.0-5.0G		2-5C | 3.0-5.0G	
faasset_1966	45.0G	2-4C | 5.0G		2-4C | 5.0G		2-4C | 5.0G	
fincore0_2431	1.3G		1.5C | 6.0G		1.5C | 6.0G		1.5C | 6.0G
finflux1_1964	142.1G	4-10C | 20.0G			4-10C | 20.0G		4-10C | 20.0G
gdtrans_1967	14.2G		2-3C | 6.0G	2-3C | 6.0G			2-3C | 6.0G
gzcomm1_2079	19.5G	1.5C | 6.0G		1.5C | 6.0G		1.5C | 6.0G	
mapitool_1969	41.4G		1-3C | 10.0G		1-3C | 10.0G		1-3C | 10.0G
mix1_1968	12.4G		2-10C | 5.0G	2-10C | 5.0G			2-10C | 5.0G
mix2_1970	314.4G		4-10C | 15.0G		4-10C | 15.0G	4-10C | 15.0G	
sys	0.0G	2.5-5C | 30.0G		2.5-5C | 30.0G		2.5-5C | 30.0G	
trans_mix4_1973	2.7G		3-6C | 8.0G		3-6C | 8.0G		3-6C | 8.0G
trans_mix5_1974	5.3G		3-4C | 8.0-10.0G		3-4C | 8.0-10.0G		3-4C | 8.0-10.0G
	内存分配	74.0-76.0G	58.0-60.0G	65.0-67.0G	67.0-69.0G	69.0-71.0G	63.0-65.0G
	总内存	80.4-321.8G	80.4-321.8G	80.5-321.9G	80.5-321.9G	80.4-321.8G	80.4-321.8G
	CPU分配	14-29.5/30-120	16.5-37.5/30-120	14-32.5/30-120	16.5-34.5/30-120	14-29.5/30-120	16.5-37.5/30-120
	磁盘使用%	7.8%	11.1%	4.6%	14.4%	12.9%	6.1%
	磁盘空闲	3311.9G	3193.3G	3427.4G	3077.8G	3131.4G	3374.0G
	总磁盘大小	3593.5G	3593.5G	3593.5G	3593.5G	3593.5G	3593.5G
	unit数目	3/6	1/7	2/7	2/6	3/6	1/7
	leader数目	21081	4472	12993	8856	19982	7376
	partition数量	34923	39837	37161	37599	39015	35745
	ob版本	1.4.79	1.4.79	1.4.79	1.4.79	1.4.79	1.4.79

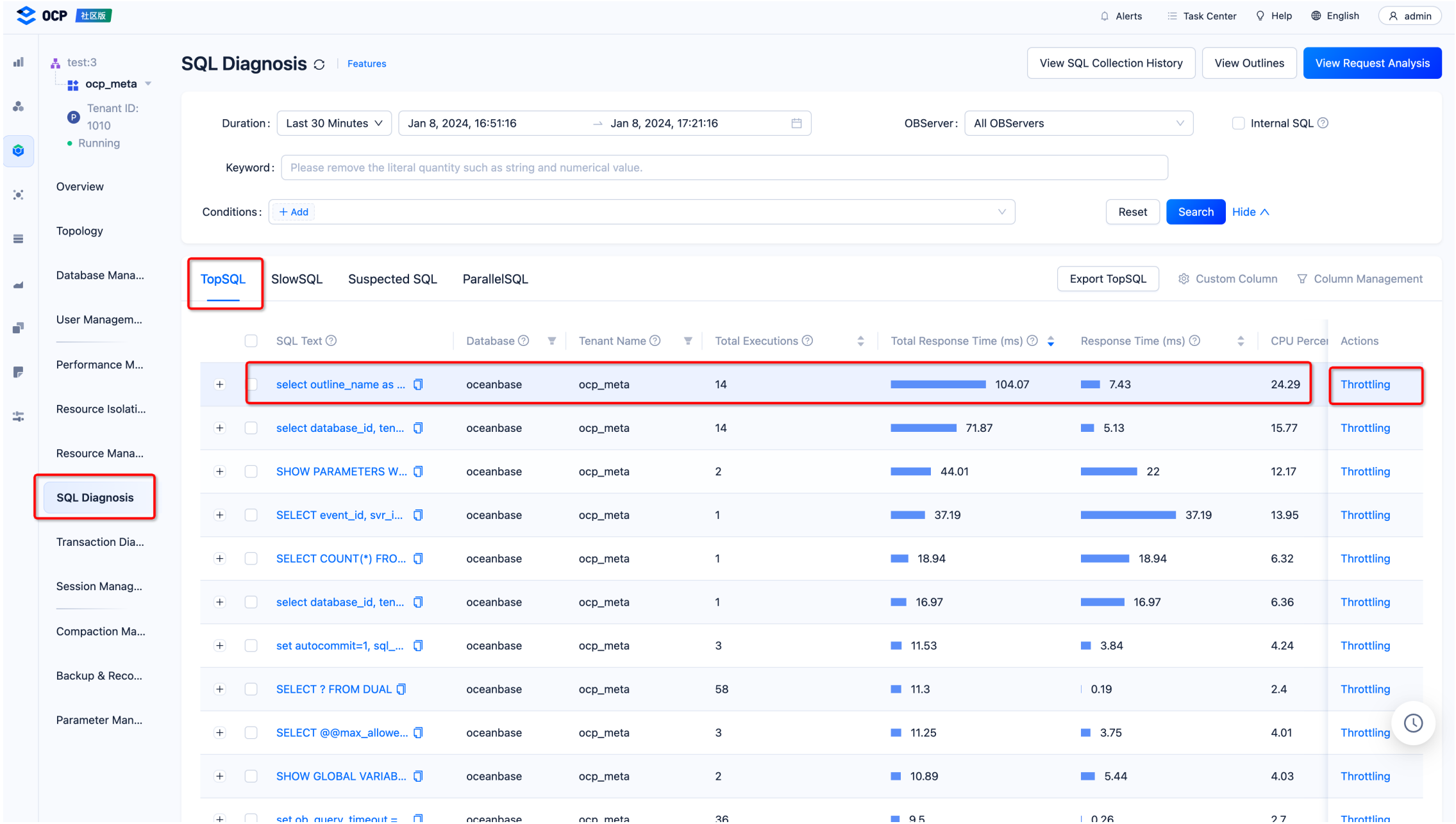
- Multi-tenancy (CPU/Mem/disk resource isolated)
- Scale up for a tenant in second

One of application of Alipay

Tenant strategy is flexible

OceanBase Console Platform - all operation in one




OCP 社区版
Alerts
Task Center
Help
English
admin
test:3
ocp_meta
Tenant ID: 1010
Running
Overview
Topology
Database Mana...
User Managem...
Performance M...
Resource Isolati...
Resource Mana...
SQL Diagnosis
Transaction Dia...
Session Manag...
Compaction Ma...
Backup & Reco...
Parameter Man...
SQL Diagnosis
Features
View SQL Collection History
View Outlines
View Request Analysis
Duration: Last 30 Minutes
Jan 8, 2024, 16:51:16
Jan 8, 2024, 17:21:16
OBServer: All OBServers
Internal SQL
Keyword: Please remove the literal quantity such as string and numerical value.
Conditions: + Add
Reset
Search
Hide
TopSQL
SlowSQL
Suspected SQL
ParallelSQL
Export TopSQL
Custom Column
Column Management
SQL Text
Database
Tenant Name
Total Executions
Total Response Time (ms)
Response Time (ms)
CPU Perce
Actions
select outline_name as ... oceanbase ocp_meta 14 104.07 7.43 24.29 Throttling
select database_id, ten... oceanbase ocp_meta 14 71.87 5.13 15.77 Throttling
SHOW PARAMETERS W... oceanbase ocp_meta 2 44.01 22 12.17 Throttling
SELECT event_id, svr_i... oceanbase ocp_meta 1 37.19 37.19 13.95 Throttling
SELECT COUNT(*) FRO... oceanbase ocp_meta 1 18.94 18.94 6.32 Throttling
select database_id, ten... oceanbase ocp_meta 1 16.97 16.97 6.36 Throttling
set autocommit=1, sql_... oceanbase ocp_meta 3 11.53 3.84 4.24 Throttling
SELECT ? FROM DUAL oceanbase ocp_meta 58 11.3 0.19 2.4 Throttling
SELECT @@max_allowe... oceanbase ocp_meta 3 11.25 3.75 4.01 Throttling
SHOW GLOBAL VARIAB... oceanbase ocp_meta 2 10.89 5.44 4.03 Throttling
set ob_query_timeout = oceanbase ocp_meta 36 9.5 0.26 2.7 Throttling

03

Best Practices

User Scenario

Data Center (stored historical data)

Cost Effective/Simple management

Multi-Tenant

Cost Effective/suitable for SOA & Saas

High availability

Anti disaster by deploy across IDC/Cities/Cloud

High Performance

Replace MySQL Sharding architecture

HTAP & OLAP

Realtime data warehouse

OBKV

Reuse all features of OB, cost effective

Typical user scenarios (1): data center (stored historical data)

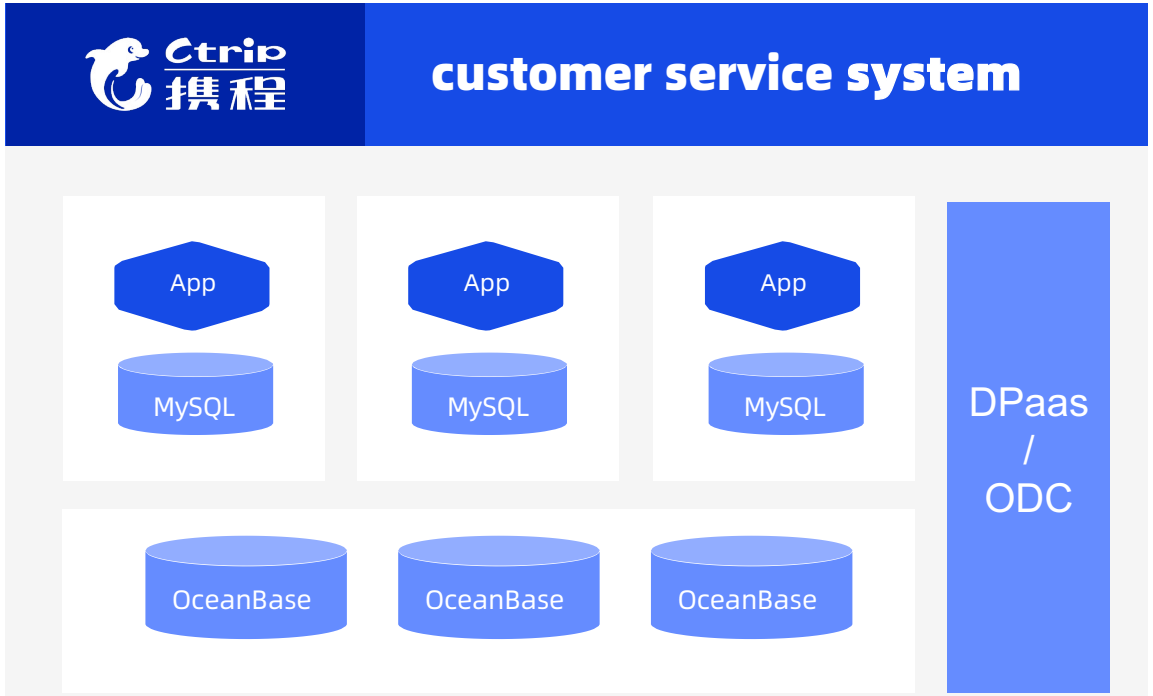
- Cost effective

Challenges

- Cost is high – the data is huge.
- Using sharding middleware
  - hard to scale
  - hard to do complex query.

Benefits

- Save Cost, the disk usage of OceanBase is only 15% of MySQL.
- Easy to scale, no sharding middleware .
- Improve writing performance 3x.



- Hot data(no more than 1 year) will be stored in MySQL
- All data will be stored in OceanBase
- Automatically archive data from MySQL to OceanBase

Typical user scenarios (2): Multiple Tenants

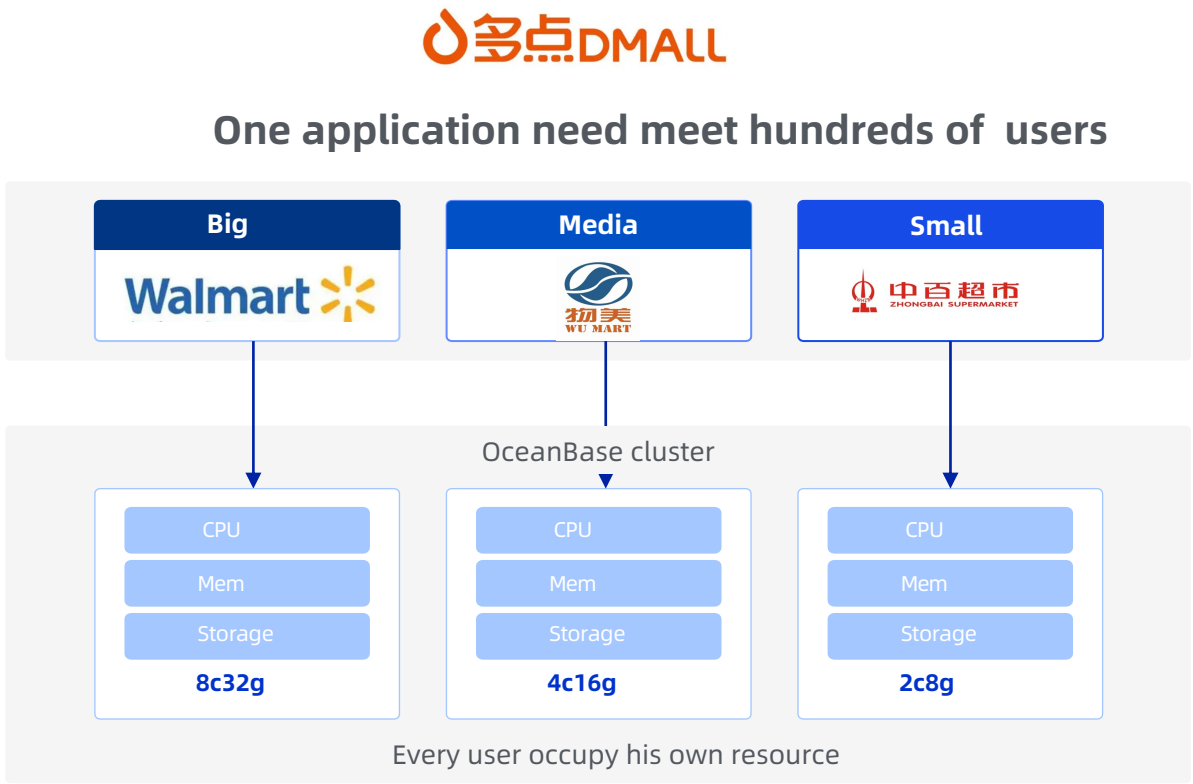
- Make full use of resource

Challenges

- Hard to operation, hundreds of MySQL instance.
- Cost is high, the load of MySQL is pretty low.
- Hard to scale up in short term when Peak time.
- Hard to do analytical task.

Benefits

- Easy to operation, one cluster for all user.
- Save cost, 1/3 hardware, make full use of resource.
- Easy to scale up in short term.
- Some analytical task can be done.



Retail SaaS

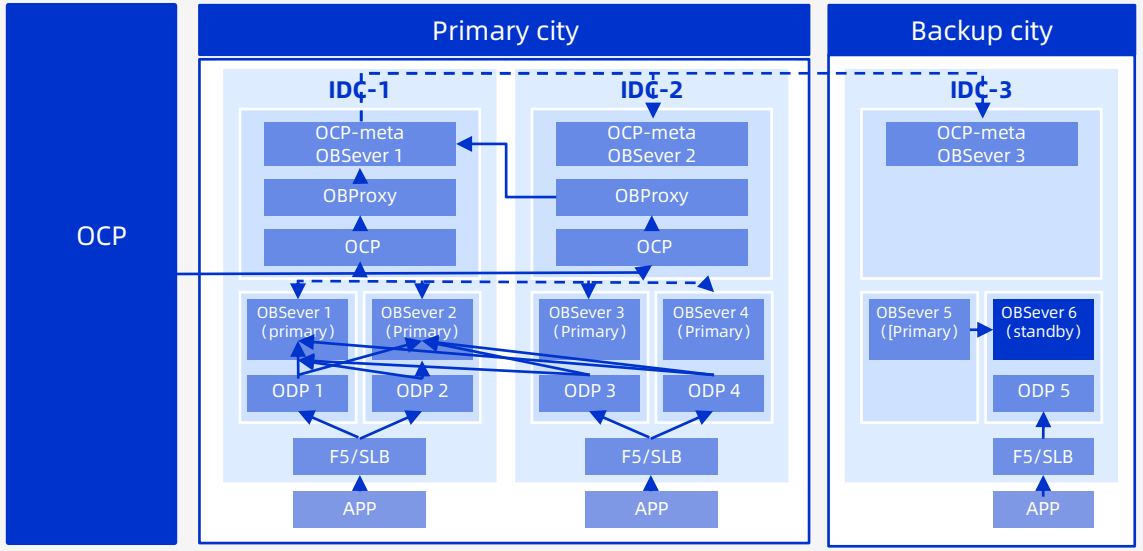
- Resource & data isolation
- Resource elastic scale in seconds

Typical user scenarios (3): High Availability

- 7 X 24 – Anti disaster

中国工商银行 INDUSTRIAL AND COMMERCIAL BANK OF CHINA

Financial Planning System
Anti disaster by across cities



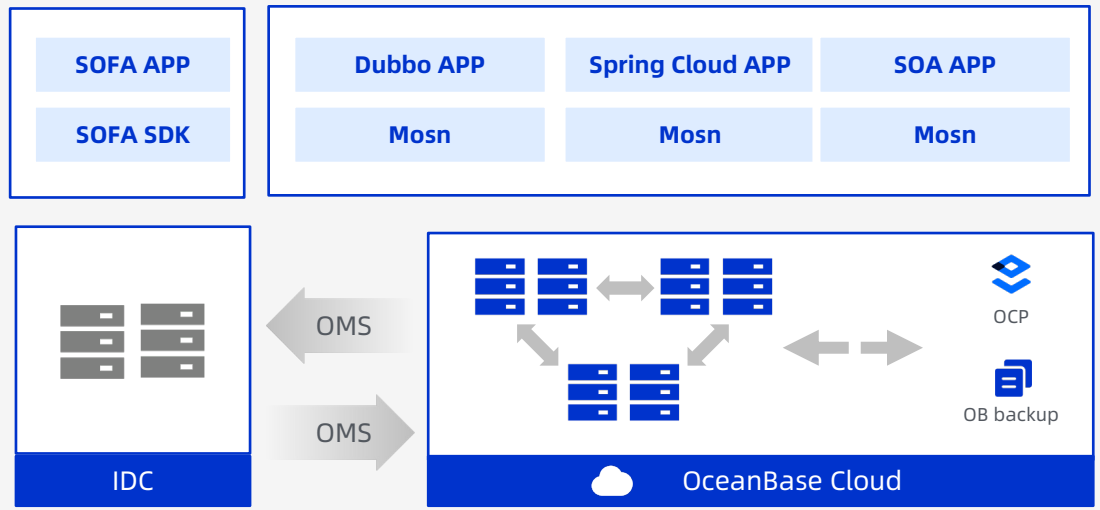
Highest level --5
Financial Industry Disaster Recovery Standards

75%
Resource usage improvement

Reduce
Hardware cost

中华保险 CHINA INSURANCE

Property Insurance System
Anti disaster by hybrid cloud



Level 3
Domestic security requirements

7x24
Disaster Recovery

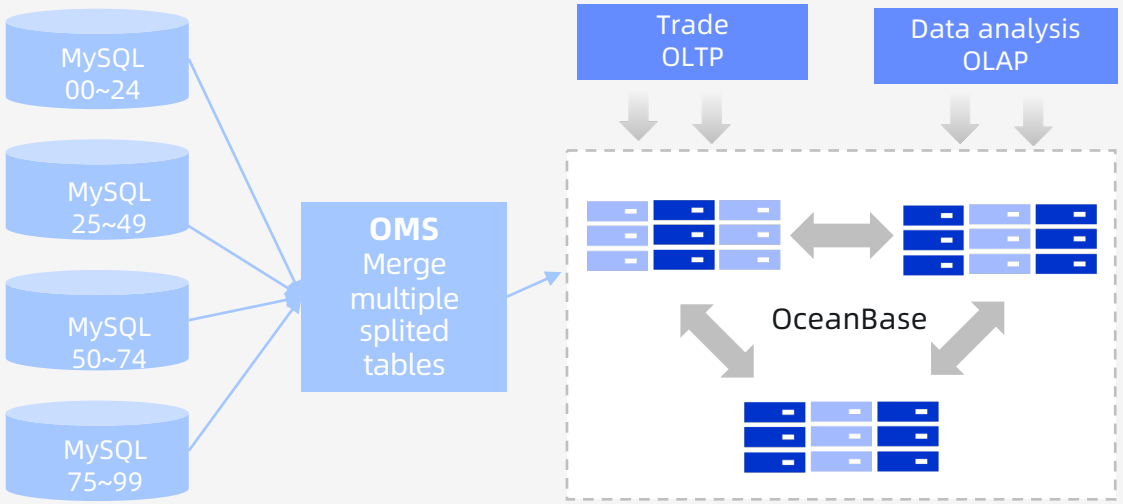
100%
Application compatible

满足等保三级安全要求

Typical user scenarios (4): High Performance

Replace MySQL Sharding architecture

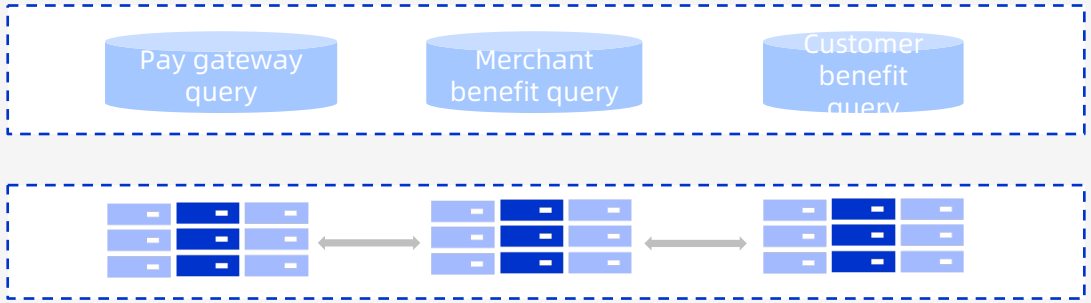
海底捞 Ordering and customer queuing system



Old system: During traffic peaks, customer queuing system, order system, member system etc are under heavy pressure and occur critical issues

30%	35%	7x24h
Trade TPS improvement	Reduce cost	Improve stability

快手 Pay Gateway System

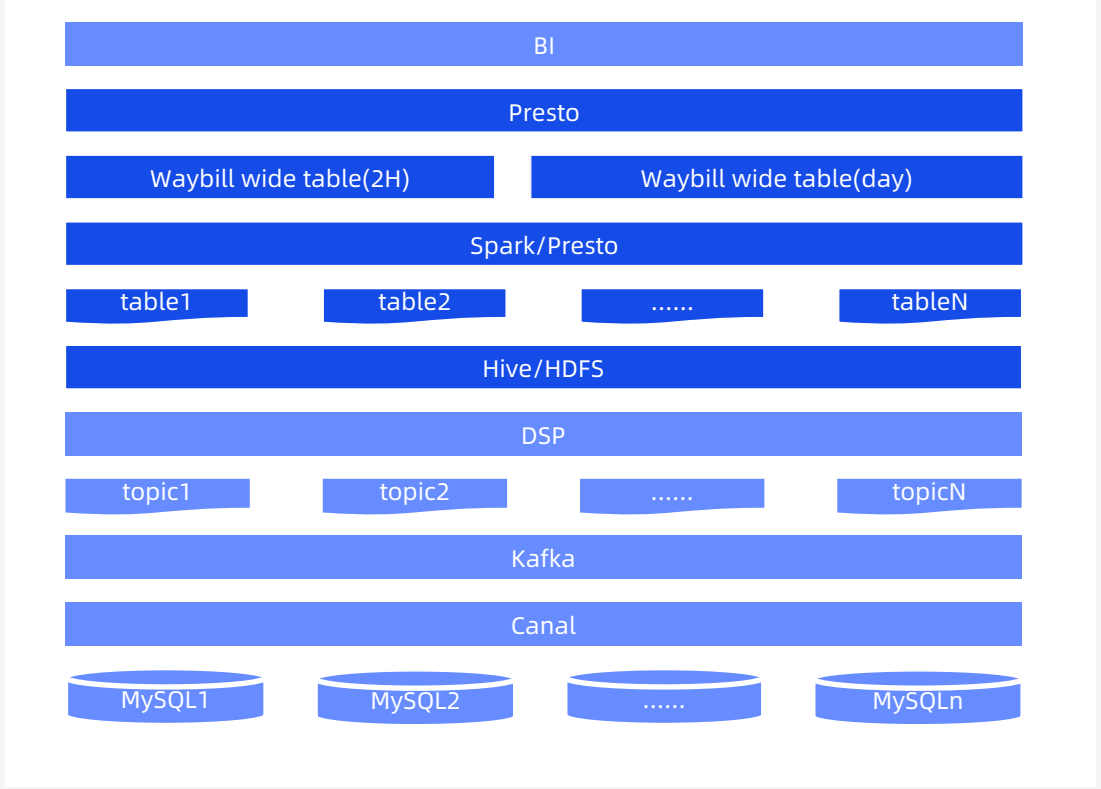


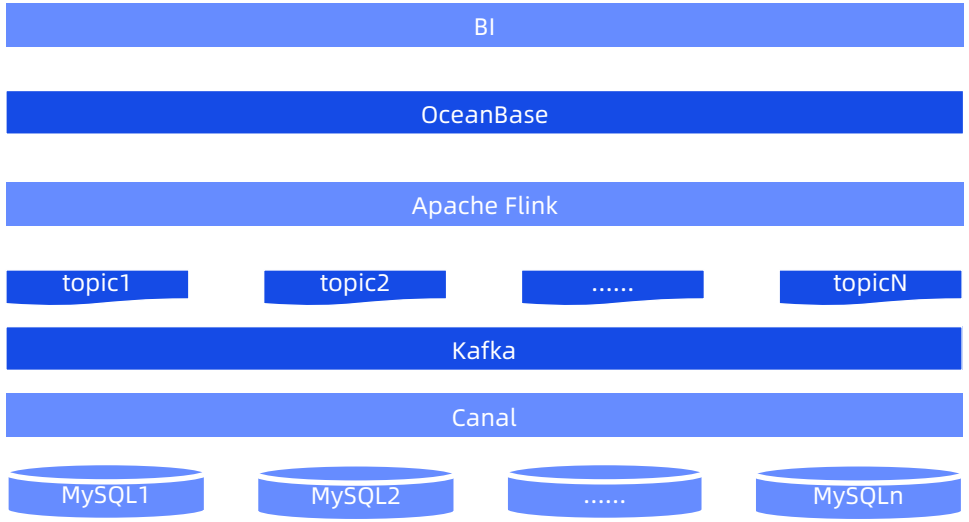
OceanBase Load

Typical user scenarios (5): HTAP & OLAP

- Realtime data warehouse

Logistics Waybill System



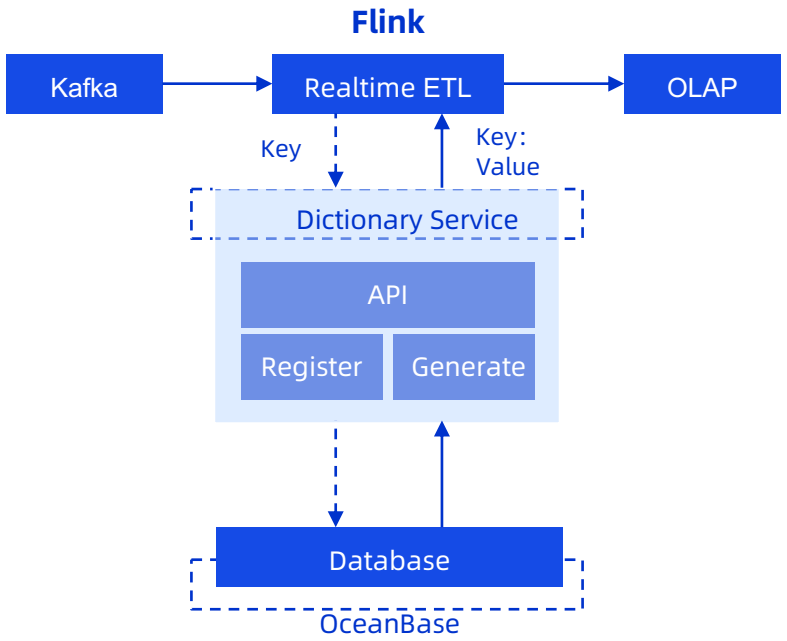


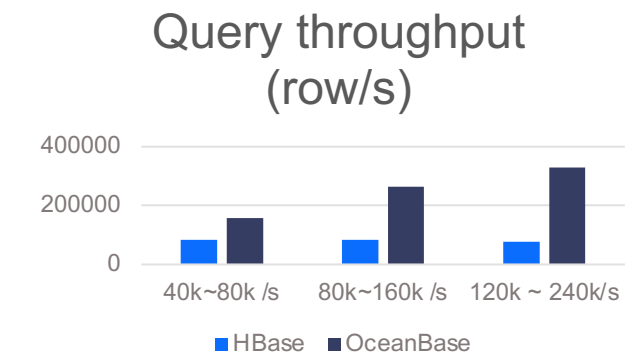
Before: Data latency: 2 Hour
Query performance: 1-10s

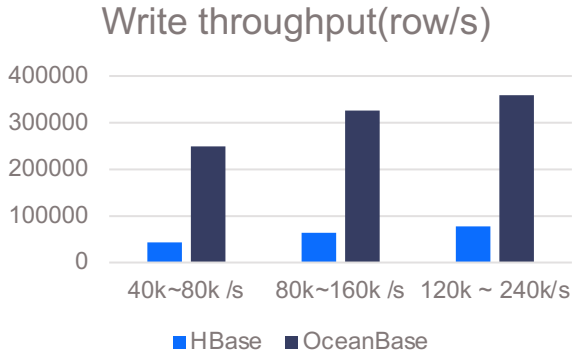
After: Data latency: **<2s**
Query performance: **<3s**
Cost reduce: **50%**

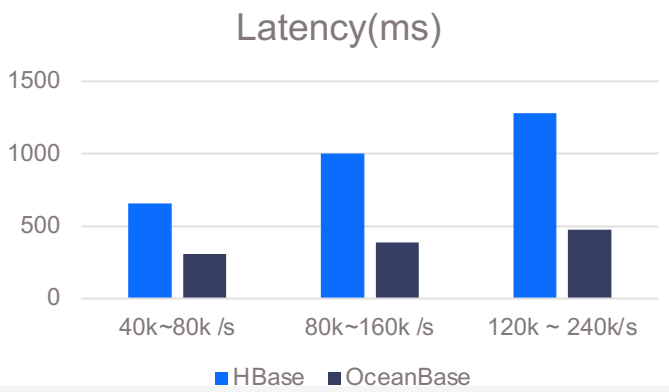
Typical user scenarios (6): OBKV

Realtime Dictionary Service









Performance improvement

- Query: 4.3
- Batch insert: 4.6
- ETL : 2.7

Thank You!

OceanBase Official website: https://oceanbase.github.io/

Forum: https://github.com/oceanbase/oceanbase/discussions

oceanbase.github.io

github.com oceanbase discussions