OCEANBASE CONF42

One Platform, Many Workloads: Powering AI Applications with OceanBase on Kubernetes

Peng Wang

Global Technical Evangelist @OceanBase

2025/08/05

Contents

01 Why AI Workloads Challenge Platform Teams

02 From Real-World Challenges to One Platform Solution

03 OceanBase: A Unified Database for Modern AI Apps

04 Closing Thoughts & Takeaways

AI Workloads Are Changing the Game

IDC

The proportion of unstructured data globally reached 92.9% in 2023.

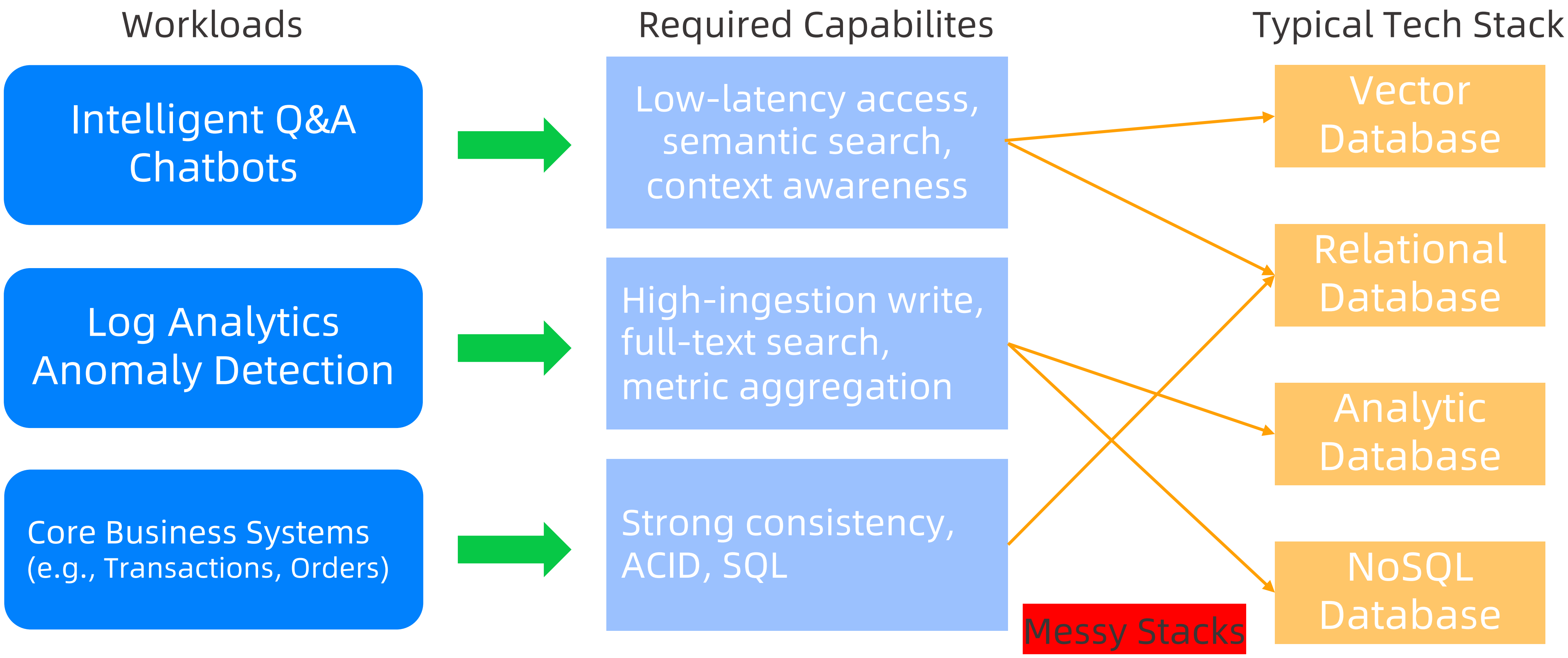
41% of executives see RAG architecture as essential, and 81% of IT leaders believe GenAI models using their own data offer a key competitive edge.

CLOUDERA

Enterprises are embracing agentic AI at scale — 66% are using enterprise AI infrastructure platforms, and 60% are embedding agent capabilities directly into their core applications.

- AI workloads (RAG, Q&A, semantic search, log analytics...) are becoming mainstream
- These workloads rely heavily on vector search, metadata, and hybrid queries
- Platform teams now need to support structured + unstructured data

The Stack Is Getting Complicated



Workloads

Intelligent Q&A Chatbots

Log Analytics Anomaly Detection

Core Business Systems (e.g., Transactions, Orders)

Required Capabilites

Low-latency access, semantic search, context awareness

High-ingestion write, full-text search, metric aggregation

Strong consistency, ACID, SQL

Typical Tech Stack

Vector Database

Relational Database

Analytic Database

NoSQL Database

Messy Stacks

Supporting modern AI workloads requires stitching together fragmented infrastructure, which slows delivery and increases complexity.

What Platform Engineers Want (But Rarely Get)

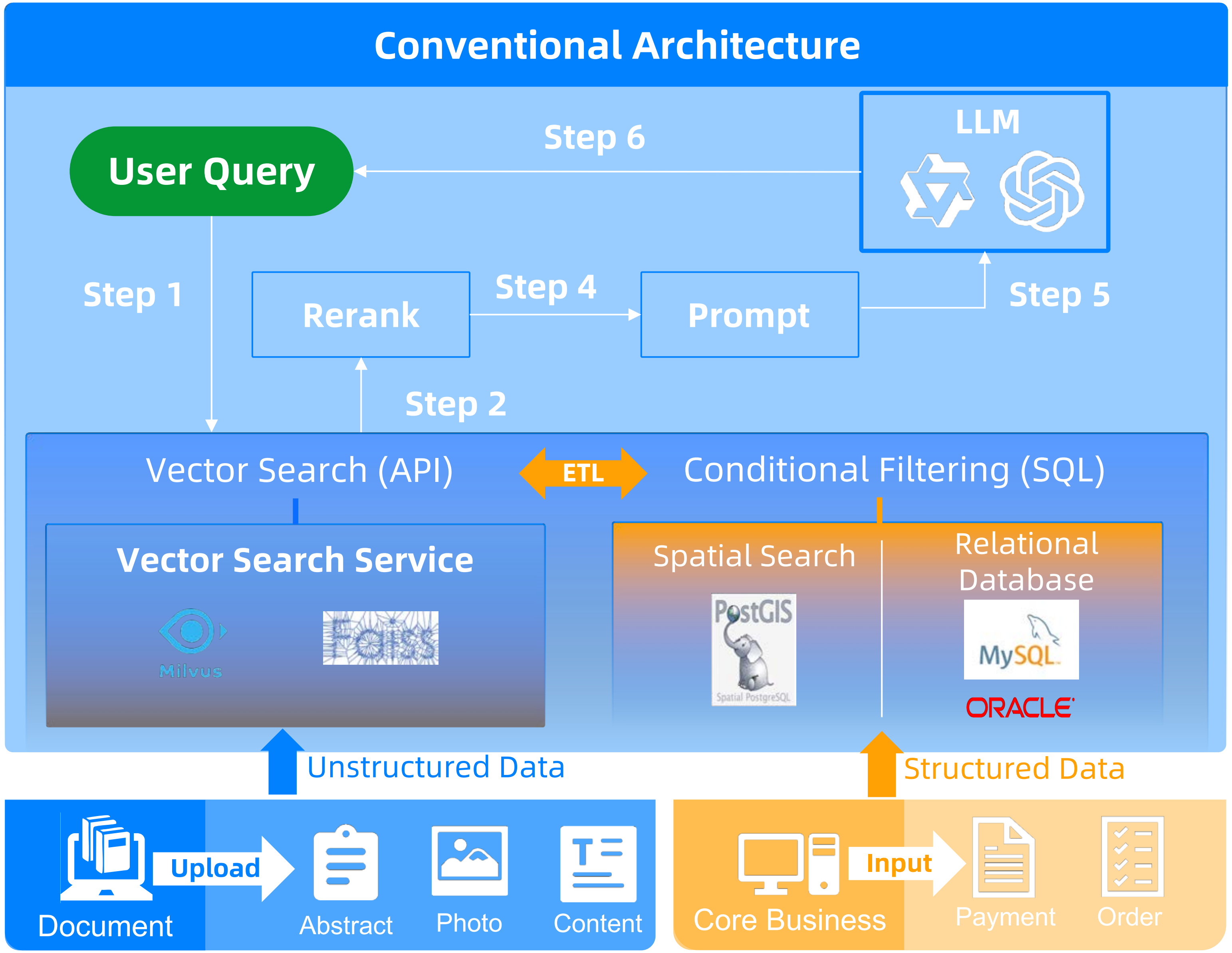
They want ...

- ✓ One platform for all data workloads
- ✓ Strong consistency & high availability
- ✓ Seamless integration with AI pipelines
- ✓ Elastic scaling on Kubernetes
- ✓ Unified access (structured + unstructured data)
- ✓ Multi-tenancy with resource isolation

But they get ...

- ✗ Fragmented data systems (RDBMS + vector DB)
- ✗ Trade-offs between performance and consistency
- ✗ Complex data movement between services
- ✗ Difficult scaling across stateful workloads
- ✗ Inconsistent query models & APIs
- ✗ Tenant interference and noisy neighbors

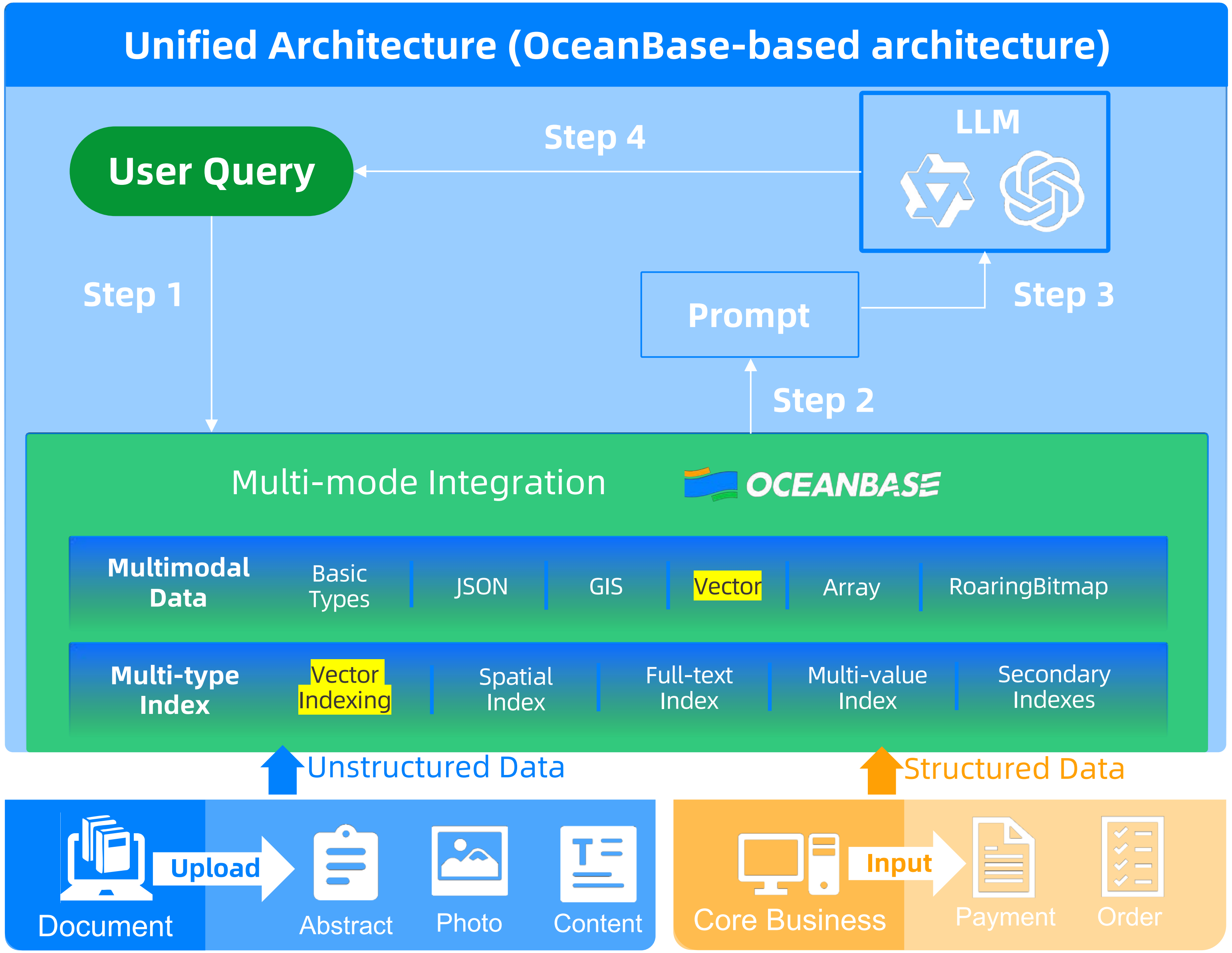
AI-Powered Knowledge Retrieval for Internal Teams



Pain Points

- Data scattered across multiple systems
- Hard to mix structured & unstructured data in one query
- High concurrency with low latency requirement

AI-Powered Knowledge Retrieval for Internal Teams



Why Engineers Finally Chose OceanBase

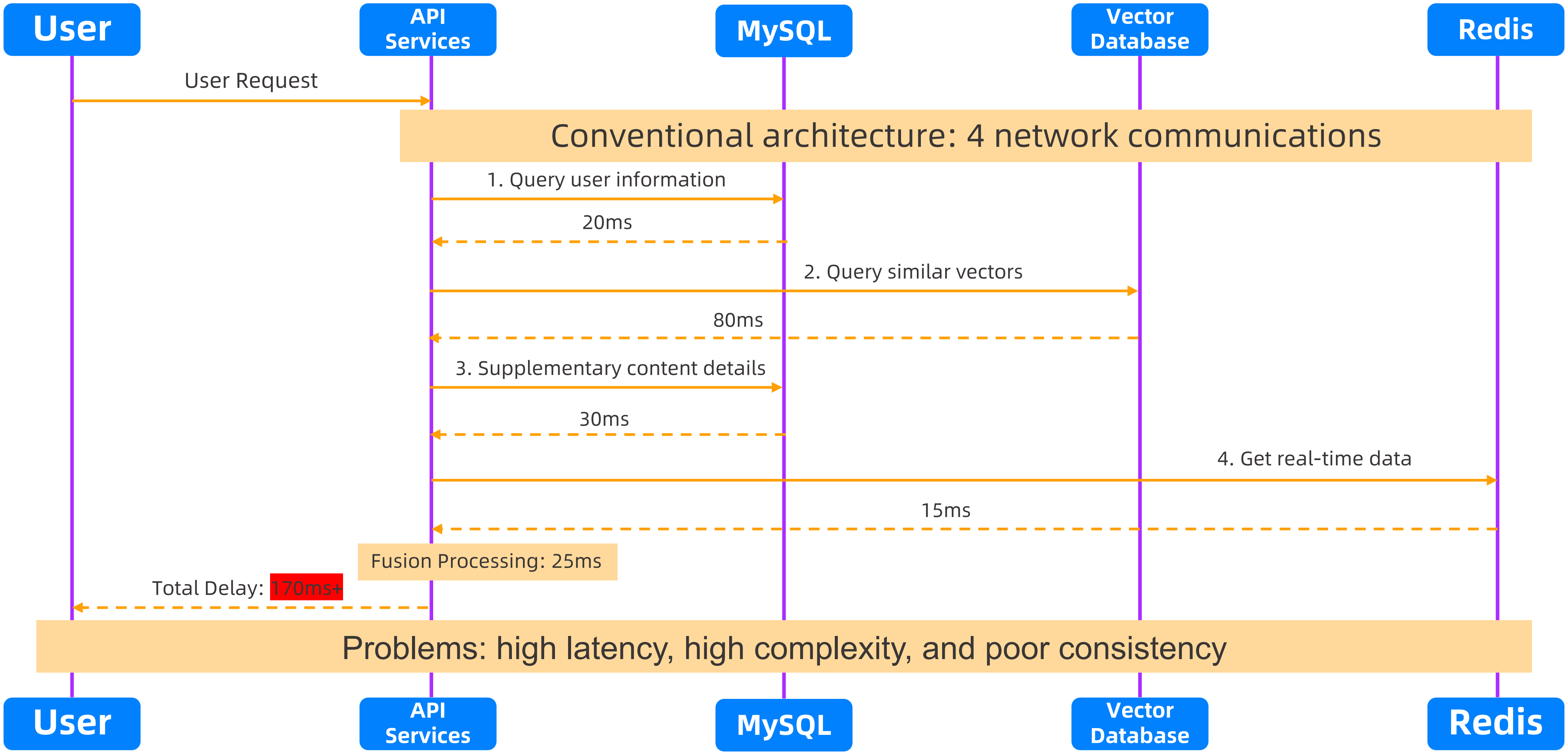
Platform Engineering Approach

- Build a unified data access layer (SQL + vector search)
- Elastic scaling on Kubernetes
- Minimize multi-system integration complexity

Outcome

- Query latency cut by over 70%
- Data freshness in minutes
- 40% less integration code to maintain

High Latency in Conventional Architecture



User

API Services

MySQL

Vector Database

Redis

User Request

Conventional architecture: 4 network communications

1. Query user information

20ms

2. Query similar vectors

80ms

3. Supplementary content details

30ms

4. Get real-time data

15ms

Fusion Processing: 25ms

Total Delay: 170ms+

Problems: high latency, high complexity, and poor consistency

User

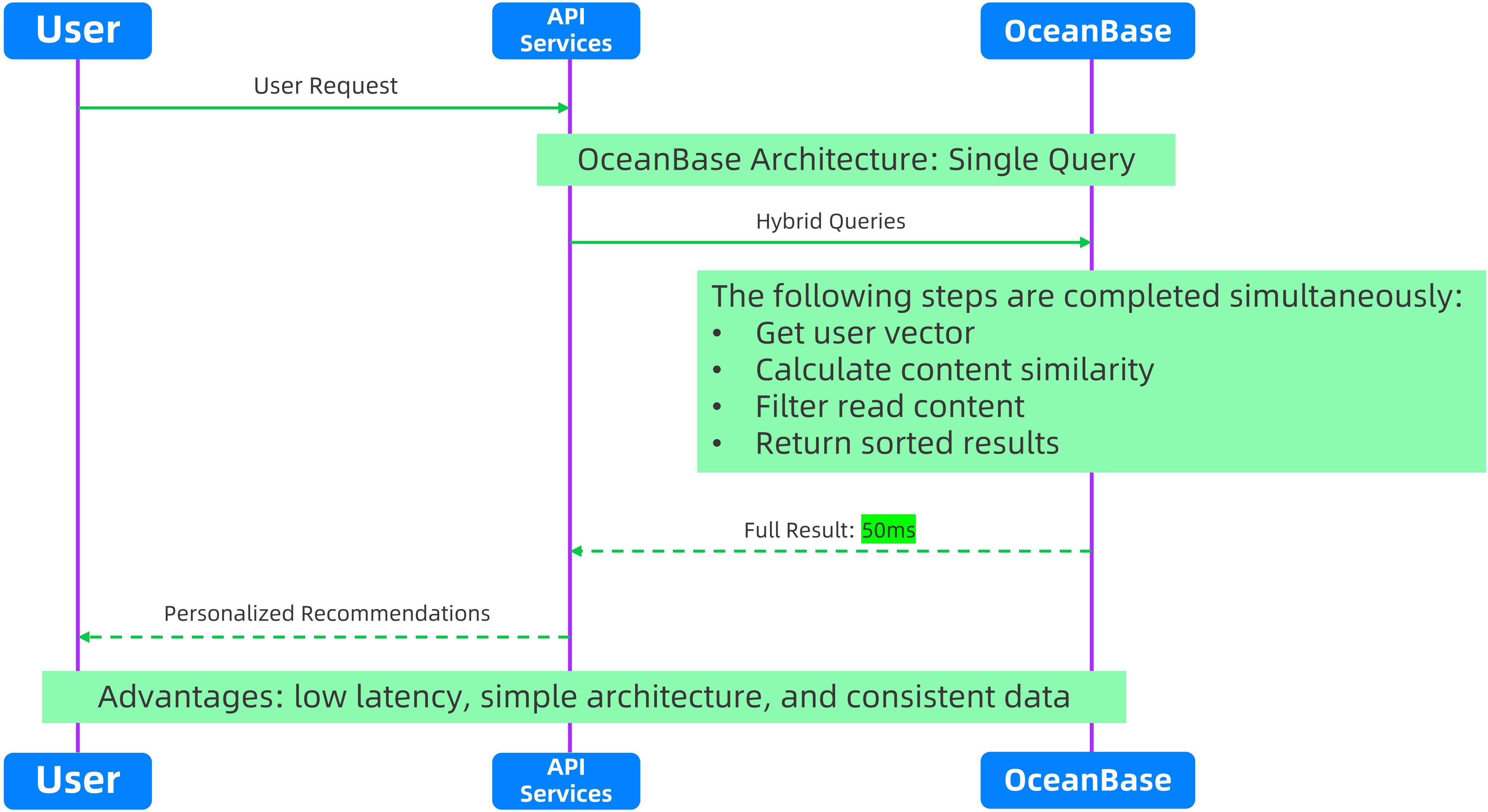
API Services

MySQL

Vector Database

Redis

Low Latency with OceanBase Unified Architecture

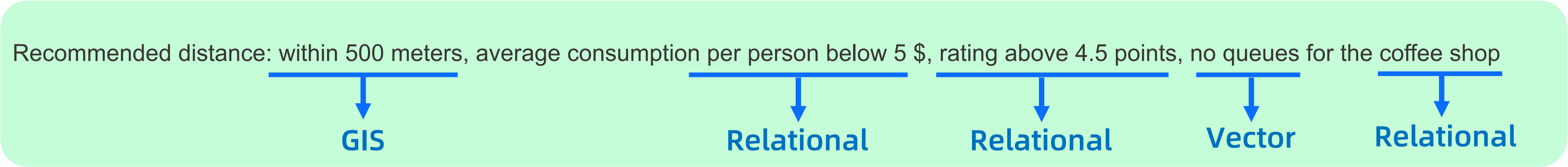


The following steps are completed simultaneously:

- Get user vector
- Calculate content similarity
- Filter read content
- Return sorted results

Advantages: low latency, simple architecture, and consistent data

Executing Hybrid Queries in OceanBase

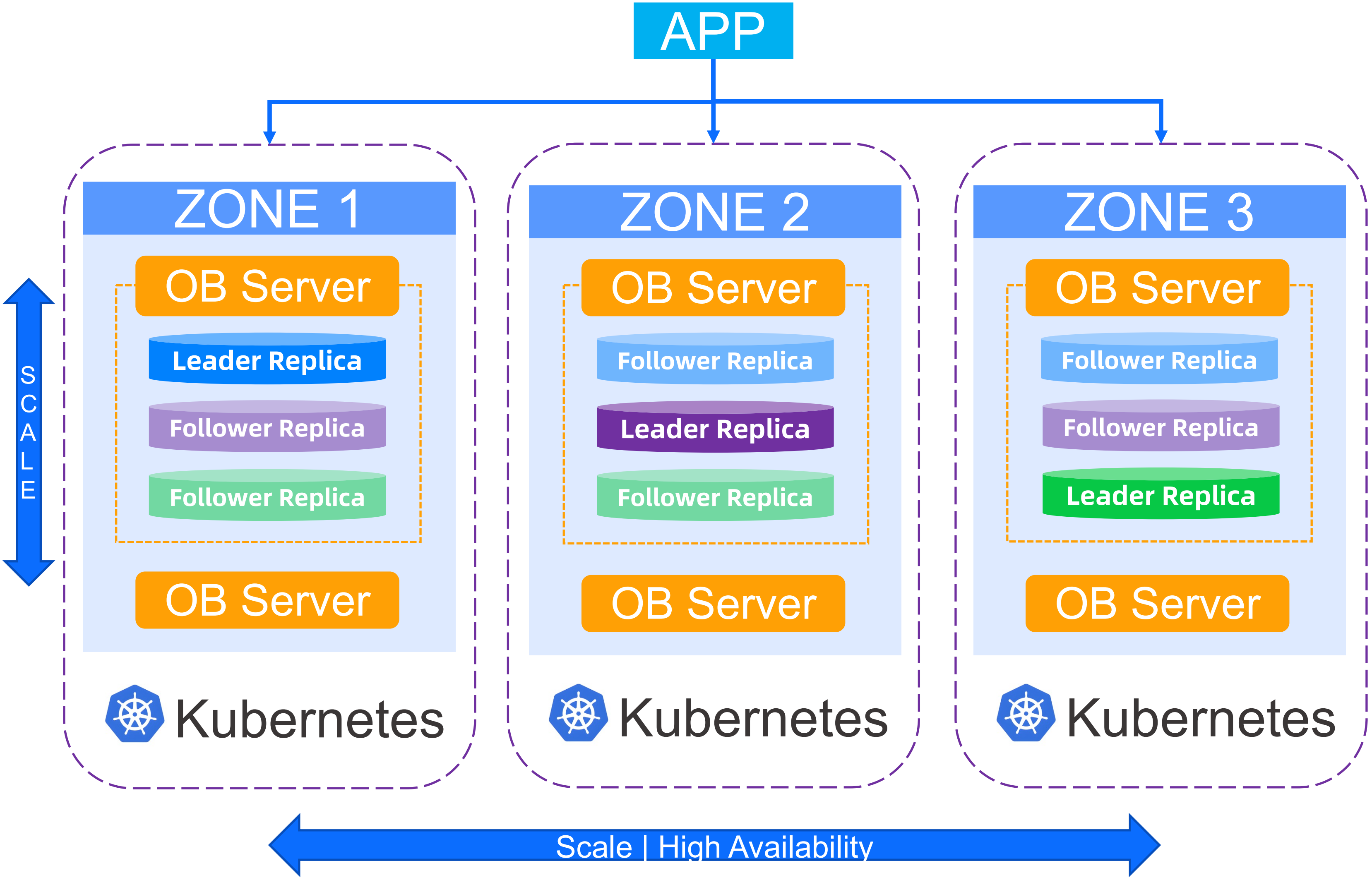


Recommended distance: within 500 meters, average consumption per person below 5 $, rating above 4.5 points, no queues for the coffee shop

Perform mixed computing directly in SQL

```
SELECT *
FROM obAgent
WHERE st_distance(location, st_srid(point(@longitude, @latitude), 4326), ‘metre’) < @query_distance
      AND score > 4.5
      AND avgConsum < 5
      AND storeType = ‘coffee shops’
ORDER BY  l2_distance(featureVec, @query_embedding) approximate limit 20;
```

OceanBase: One Unified Architecture. Multiple Capabilities (1)



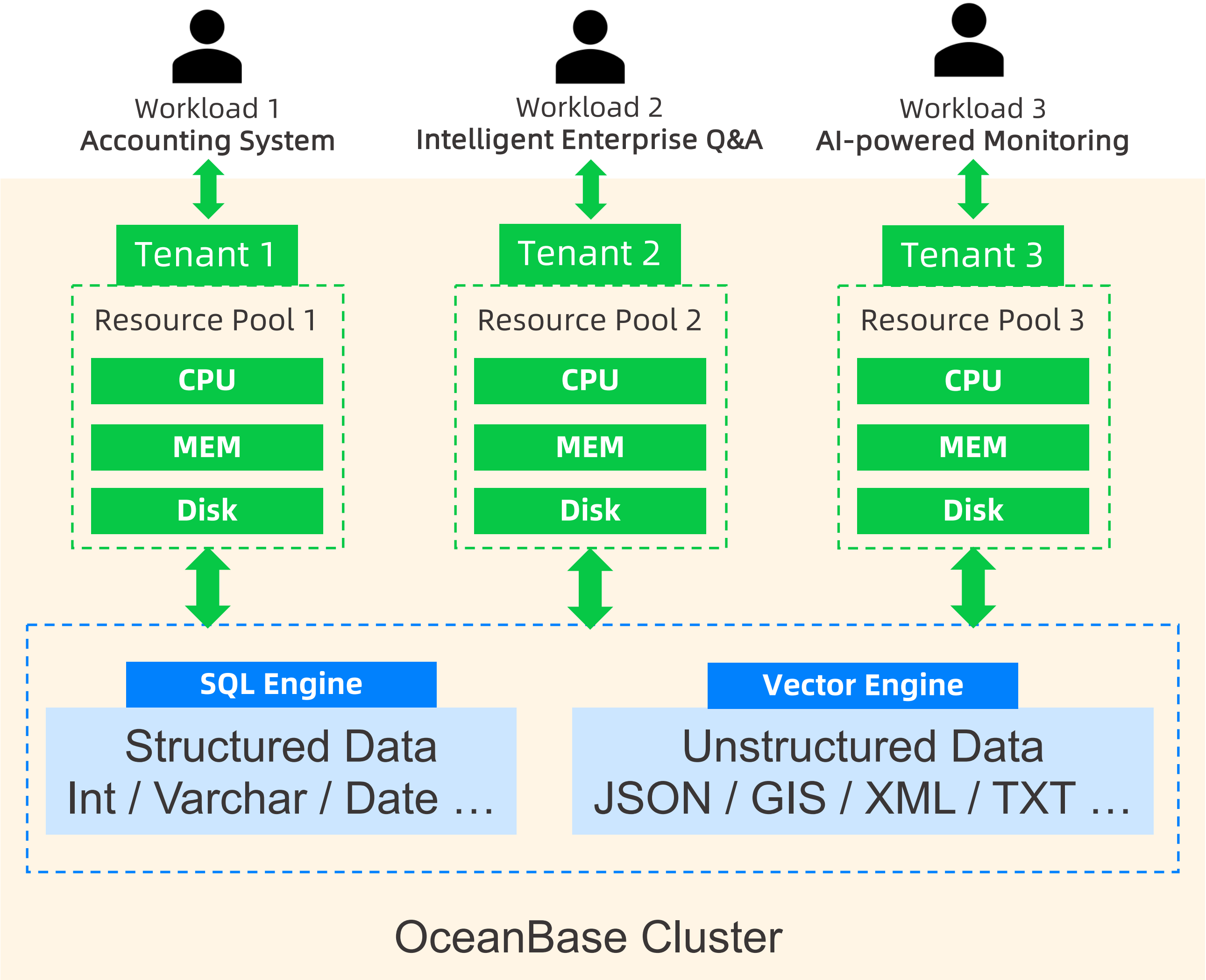
Elastic scaling on Kubernetes

- OceanBase is fully containerized, supports Operator management, and allows for elastic horizontal expansion of computing and storage.

Strong consistency & high availability

- Paxos consensus algorithm, triple replica mechanism, and automatic failover ensure high availability and strong consistency

OceanBase: One Unified Architecture. Multiple Capabilities (2)



OceanBase provides native multi-tenant architecture, resource quotas, and data isolation between tenants

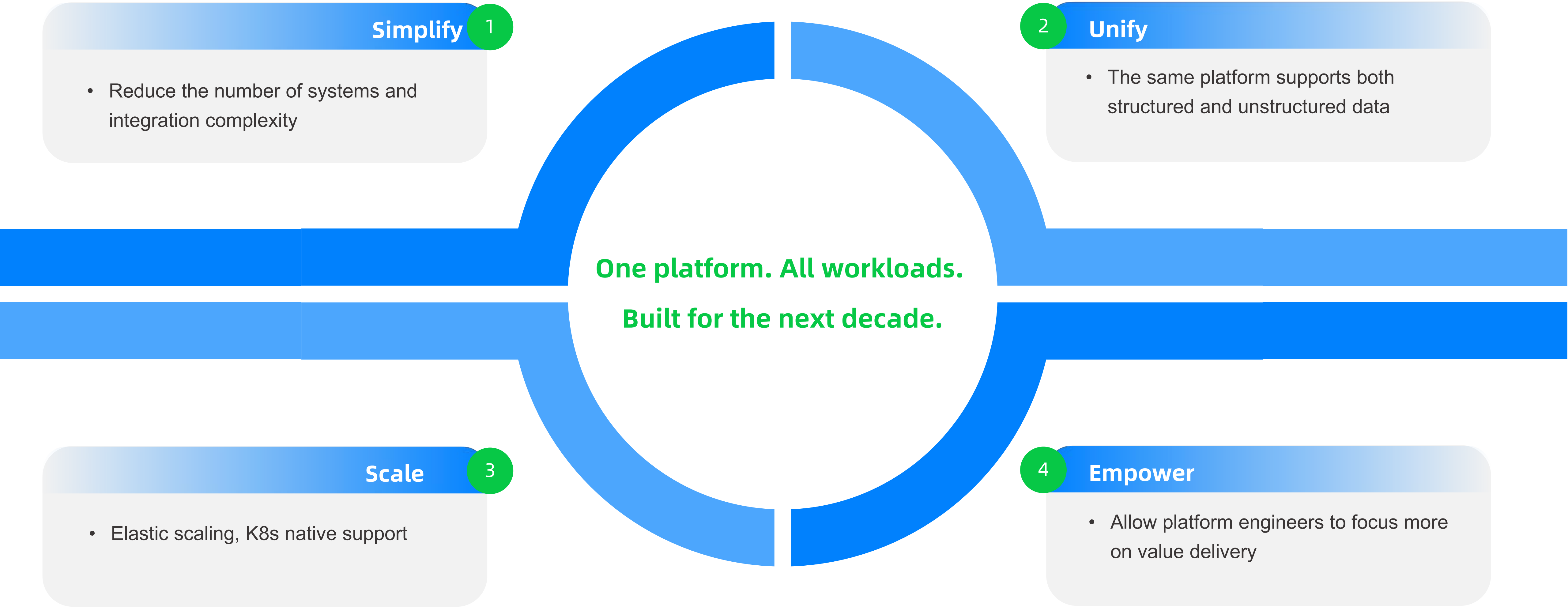
One platform for all data workloads

- One engine supports OLTP, log analysis, semantic retrieval, AI applications, etc.

Unified access (structured + unstructured)

- SQL engine supports structured query, full text search and vector search for unstructured

Closing Thoughts & Takeaways



Simplify 1

- Reduce the number of systems and integration complexity

Unify 2

- The same platform supports both structured and unstructured data

One platform. All workloads.

Built for the next decade.

Scale 3

- Elastic scaling, K8s native support

Empower 4

- Allow platform engineers to focus more on value delivery

Thank you!