NewSQL Cloud Database / DaaS   CMS: Elastic, Scalable, Highly-Available

ScaleDB

"ScaleDB looks to be the MySQL version of Oracle RAC"
--- Brian Aker, Drizzle Architect, Sun Microsystems

Xin Jia

Mar 05, 2012

What's ScaleDB

- ScaleDB is a pluggable storage engine for MySQL. It turns your MySQL application into an enterprise-class, highly-available, clustered database that scales elastically in a public cloud, private cloud, or on premise.

- ScaleDB is a NewSQL pioneer, delivering the advantages of both SQL and NoSQL, while targeting the cloud.

MySQL Architecture(1)
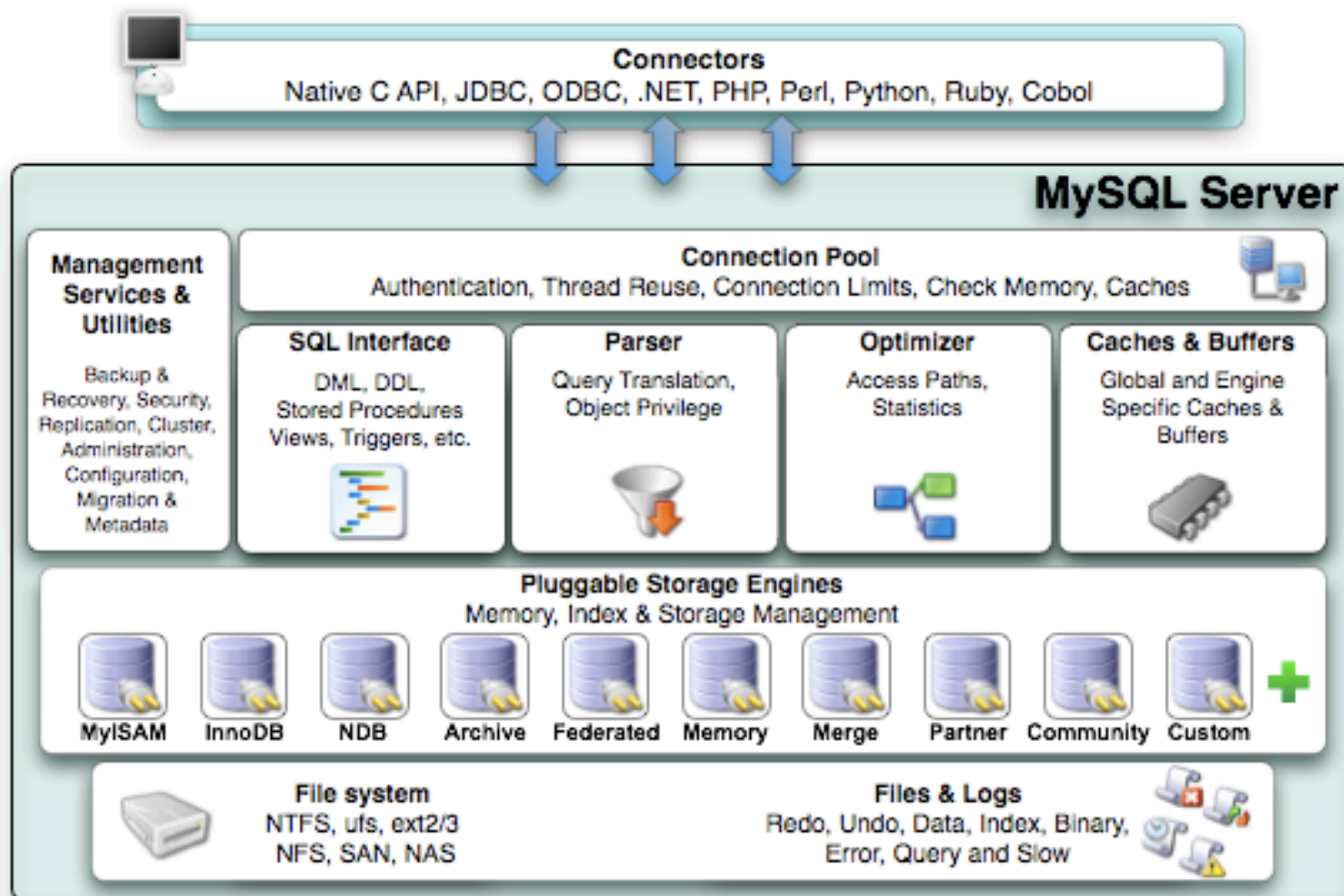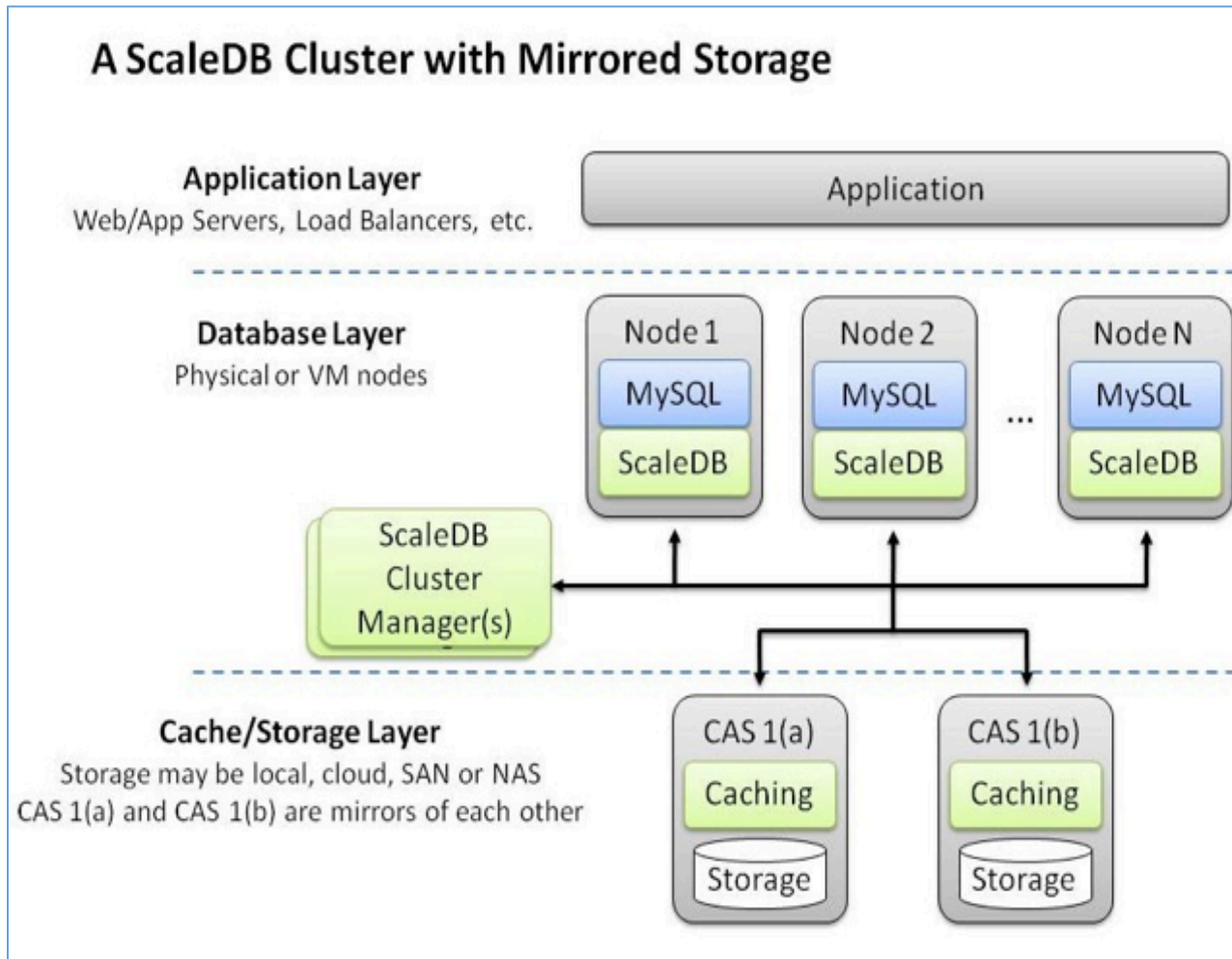
MySQL Architecture(2)
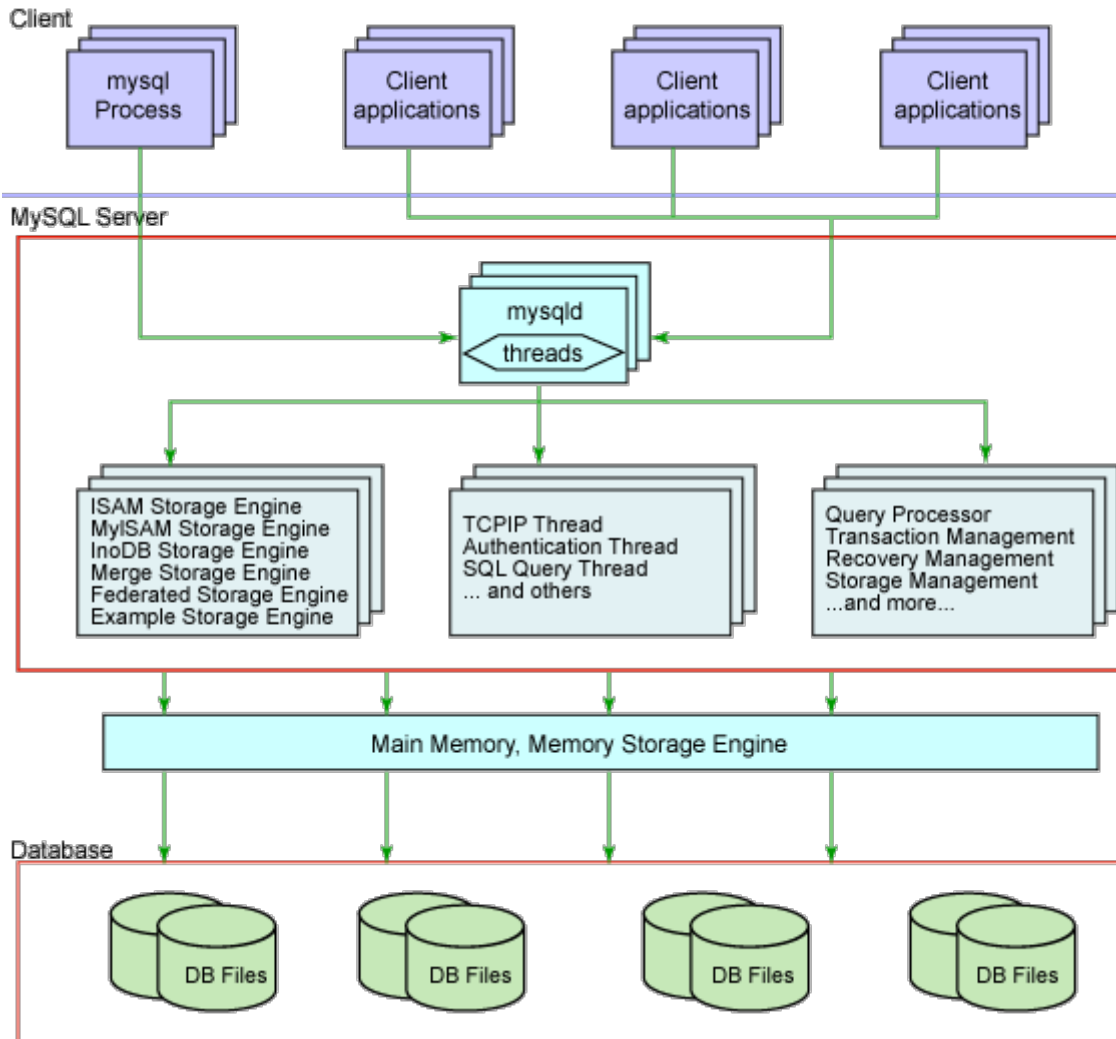
Bottleneck:
1. Disk Speed (I/0)
2. CPU
3. Capacity (users,data)
4. Performance

ScaleDB improvements:

1. Large numbers of concurrent users
2. Large data sets
3. Large numbers of tables queries
4. Load balancing
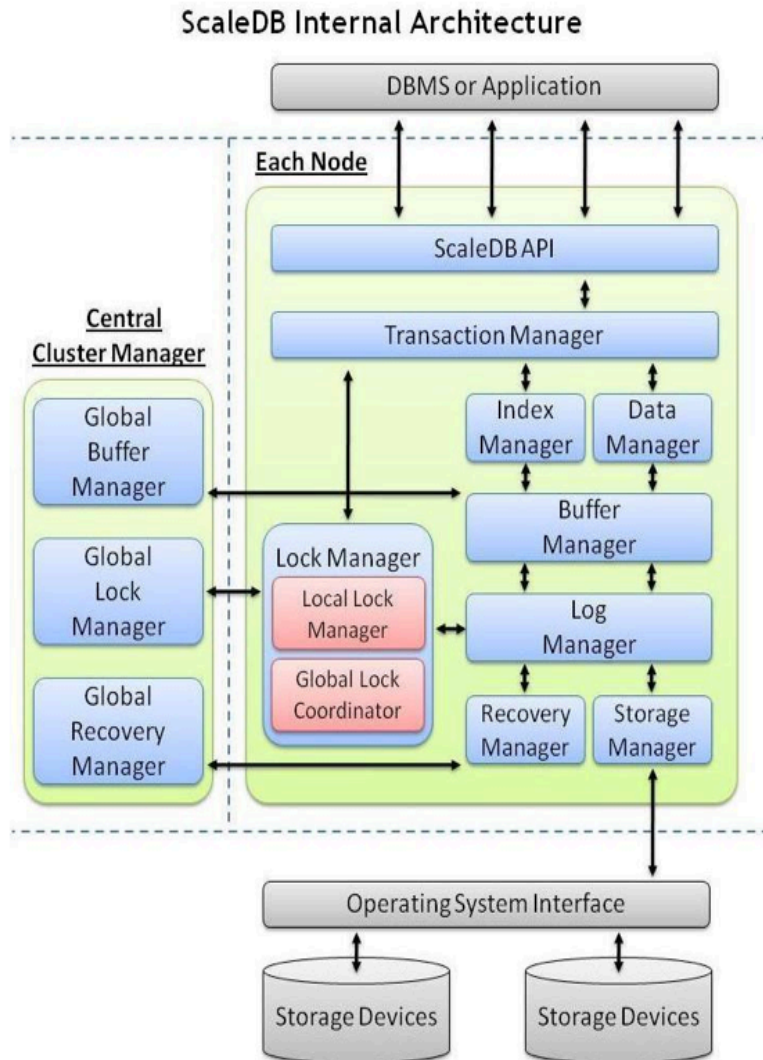5. Cloud elasticity
6. Lower TCO

ScaleDB cluster with Mirrored Storage

A ScaleDB Cluster with Mirrored Storage

Application Layer
Web/App Servers, Load Balancers, etc.

Application

Database Layer
Physical or VM nodes

Node 1
MySQL
ScaleDB

Node 2
MySQL
ScaleDB

...

Node N
MySQL
ScaleDB

ScaleDB Cluster Manager(s)

Cache/Storage Layer
Storage may be local, cloud, SAN or NAS
CAS 1(a) and CAS 1(b) are mirrors of each other

CAS 1(a)
Caching
Storage

CAS 1(b)
Caching
Storage

What's New:

1. Shared-Disk Clustering
2. Multi-Table Indexing
3. CAS (Cache Accelerator Server)
4. Locality

Source: http://www.scaledb.com/architecture.html

ScaleDB Internal Architecture

ScaleDB Internal Architecture

DBMS or Application

Each Node

ScaleDB API

Transaction Manager

Index Manager

Data Manager

Buffer Manager

Log Manager

Recovery Manager

Storage Manager

Central Cluster Manager

Global Buffer Manager

Global Lock Manager

Global Recovery Manager

Lock Manager

Local Lock Manager

Global Lock Coordinator

Operating System Interface

Storage Devices

Storage Devices

The Performance Challenges

1. Data contention
- Locality to reduce data movement (certain nodes in the database specialize in certain data)

2. Sharing data via the disk
- Shared cache done through CAS

3. Maximize local cache
- Improving cache hit ratio using CAS

4. Network traffic for messaging
- Locality eliminate synchronazing messing

5. Network traffic for large chunks of data
- Local processing by CAS to send results only

Features Comparison

Feature	Shared-Disk	Shared-Nothing	ScaleDB
Elasticity	Yes	No	Yes
High-Availability	Yes	No	Yes
Ease of Use	Medium	Easy	Easy
Ease of Scaling	Easy	Hard	Easy
Software Costs	High	Low	Low
Storage Costs	High (SAN/NAS)	Low (PC)	Low (PC)
Performance	Medium	High	High

Source: http://www.scaledb.com

Features Summary

- **High-Availability**: Node failure affect less

- **Flexibility**: Share workload

- **High Performance**: MTI (Multi-Table Index)

- **Seamless scalability**: Add nodes & storage with shutting down

- **Less TCO**(Total Cost of Ownership):

Innovative high-speed indexing	High-performance transaction processing
Plug-and-Cluster™ simplicity	Graceful fault-tolerance
Automatic data recovery	Row-level locking
ACID compliance	Multi-node concurrency control
Shared-everything architecture	Eliminates the requirement to partition data

Source: http://www.scaledb.com

Different with H-Store

- ScaleDB
  - Locality
  - CAS other than disk
  - Paired CAS on different nodes
  - Data contention lock
  - Cluster Manager

- H-Store
  - CTA
  - Main Memory
  - Partition Data replica on different nodes
  - Multi-partition txns
  - Site Coordinator

Answer questions

- 1. Does the system support distributed transactions?
  - No.

- 2. Is the system open-source?
  - NO. (Except for connectors for MySQL)

- 3. Does the system support stored procedures?
  - NO.

Sources