CMU SCS

Carnegie Mellon Univ. Dept. of Computer Science 15-415/615 - DB Applications

C. Faloutsos – A. Pavlo

How to Scale a Database System

CMU SCS

hag·i·og·ra·phy
(noun)

CMU SCS

ChristosTheGreekGodofDatabases.com

- Pinterest meets Causal Encounters meets Kickstarter meets Twitter
  - With Christos!

CMU SCS

ChristosTheGreekGodofDatabases.com

- More reads than writes.
- All media stored outside of DBMS.
- How do we choose the right database architecture?

CMU SCS

Outline

- Single-Node Databases
- NoSQL Systems
- NewSQL Systems

CMU SCS

Late-1990s / Early-2000s

- All the big players were heavyweight and expensive.
  – Oracle, DB2, Sybase, SQL Server, Informix.
- Open-source databases were missing important features.
  – Postgres, mSQL, MySQL.

Mid-2000s

- MySQL + InnoDB is widely adopted by new web companies:
  - Supported transactions, replication, recovery.
  - Memcache for caching queries.

ChristosTheGreekGodofDatabases.com

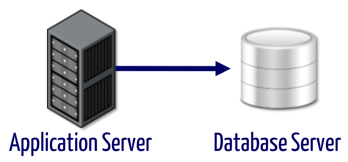
- Let’s go with MySQL.
- We’re getting a lot of traffic.
- Our database server is saturated!

How do we increase the capacity of our database server?

Idea #1: Buy a faster machine.

CMU SCS

Scaling Up



- More disks.
- More RAM.
- Faster CPUs.
- Use SSDs.

(+) Requires no change to application.
(+) Improvements are immediate.

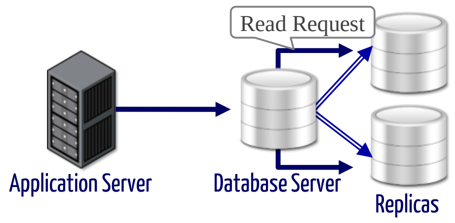
(-) Expensive! Diminishing Returns.
(-) Single Point of Failure.

CMU SCS

Idea #2: Replicate database on multiple servers.

CMU SCS

Replication



(+) Requires no change to application.
(+) Parallelize read operations.
(+) Improved fault tolerance.

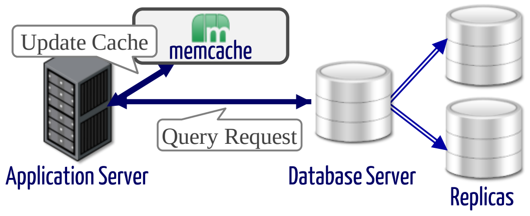
(-) Expensive! Diminishing Returns.
(-) Writes limited to slowest node.

CMU SCS

Idea #3: Cache query results.

CMU SCS

Query Cache



(+) Reduce load on DBMS.
(+) Fast API.

(-) Extra roundtrip per query.
(-) Requires application changes.
(-) Doesn't help write-heavy apps.

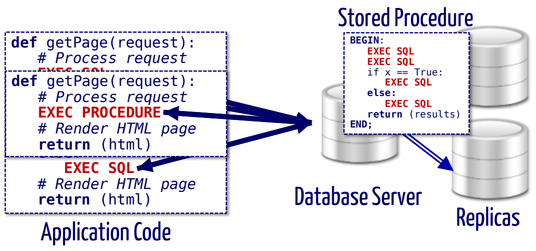
Faloutsos/Pavlo CMU SCS 15-415/615 14

CMU SCS

Idea #4: Push SQL into stored procedures.

CMU SCS

Stored Procedures



```
def getPage(request):
   # Process request
   EXEC SQL
   # Render HTML page
   return (html)
```

```
def getPage(request):
   # Process request
   EXEC PROCEDURE
   # Render HTML page
   return (html)
```

```
BEGIN:
  EXEC SQL
  EXEC SQL
  if x == True:
    EXEC SQL
  else:
    EXEC SQL
  return (results)
END;
```

(+) Reduces network roundtrips.
(+) Less lock contention.
(+) Modularization.

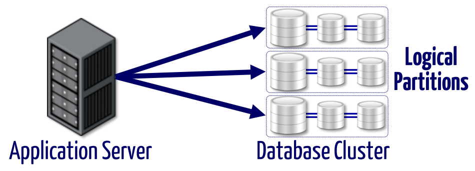
(-) Application logic in two places.
(-) PL/SQL is not standardized.

CMU SCS

Idea #5: Shard database across multiple servers.

CMU SCS

Sharding / Partitioning



(+) Parallelize all operations.
(+) Much easier to add more hardware.

(-) Most DBMSs don't support this.
(-) Joins are expensive.
(-) Non-trivial to split database.

CMU SCS

ChristosTheGreekGodofDatabases.com

- We want to scale out but writing a sharding layer is hard.
- Some parts of our application don't need a full-featured DBMS.

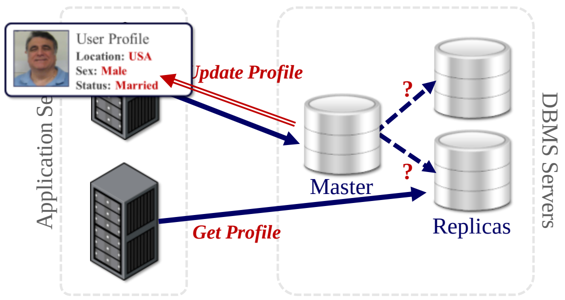
Faloutsos/Pavlo CMU SCS 15-415/615

CMU SCS

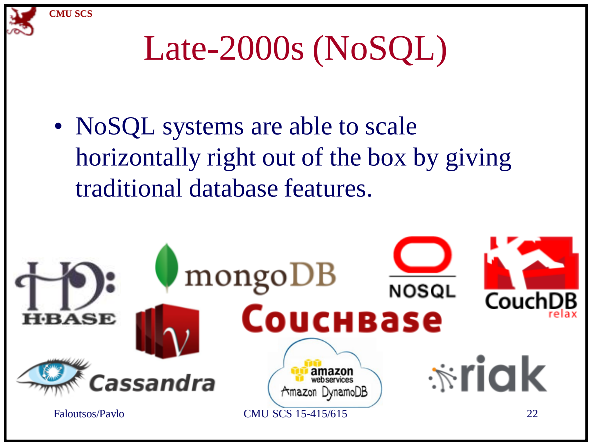
Idea #6: Give up ACID guarantees for scalability.

CMU SCS

Eventual Consistency



Faloutsos/Pavlo CMU SCS 15-415/615 21

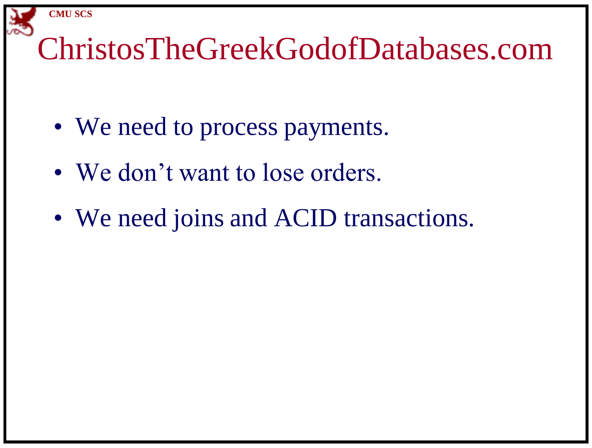


CMU SCS

Late-2000s (NoSQL)

- NoSQL systems are able to scale horizontally right out of the box by giving traditional database features.

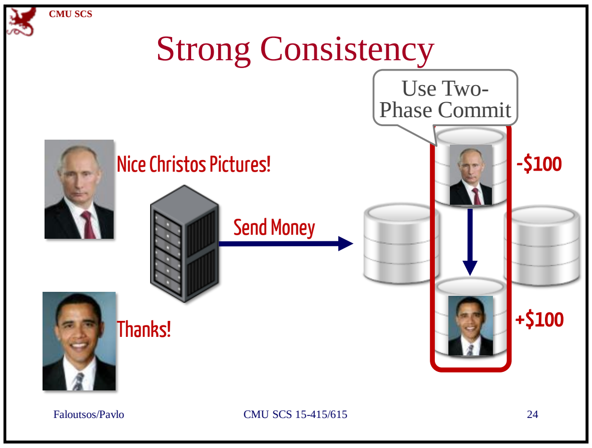
Faloutsos/Pavlo CMU SCS 15-415/615 22



CMU SCS

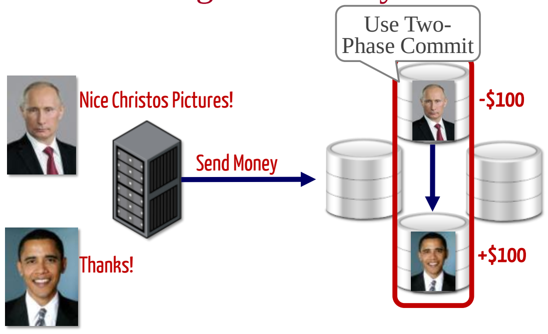
ChristosTheGreekGodofDatabases.com

- We need to process payments.
- We don’t want to lose orders.
- We need joins and ACID transactions.



CMU SCS

Strong Consistency



Faloutsos/Pavlo CMU SCS 15-415/615 24

CMU SCS

Idea #7: Keep guarantees, optimize for workload type.

CMU SCS

Early-2010s (NewSQL)

- New DBMSs that can scale across multiple machines natively and provide ACID guarantees.

NUODB VOLTDB ScaleBase SAP HANA ScaleArc deepdb H-Store ScaleDB Clustrix memsql Tokutek dbShards

CMU SCS

Conclusion

- RDBMS (Single-Node):
  - MySQL, Postgres
- NoSQL (Multi-Node):
  - Key-Value, Documents, Graphs
- NewSQL (Multi-Node):
  - Transaction Processing, MySQL Sharding

Faloutsos/Pavlo CMU SCS 15-415/615 27

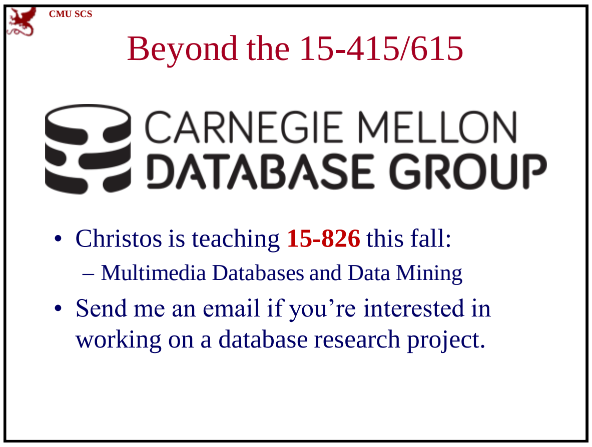
CMU SCS

What DBMS should my start-up use?

PostgreSQL

MariaDB

WebScaleSQL



CMU SCS

Beyond the 15-415/615

CARNEGIE MELLON DATABASE GROUP

- Christos is teaching **15-826** this fall:
  - Multimedia Databases and Data Mining
- Send me an email if you're interested in working on a database research project.