NOSQL

When RDBMS's don't cut it

Random Access to Planet-Size Data

Scaling up MySQL etc. to massive loads requires extreme measures

- Denormalization
- Caching layers
- Master/slave setups
- Sharding
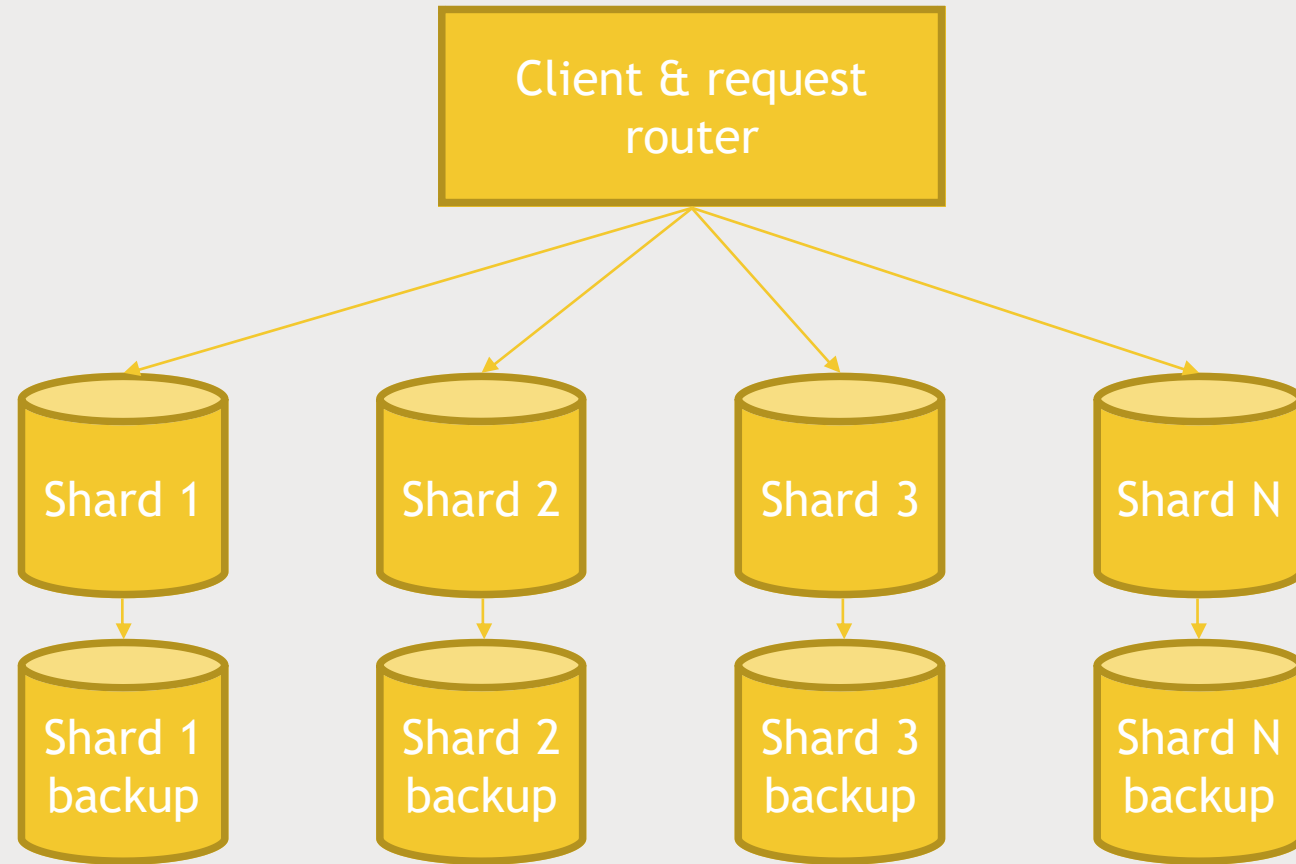- Materialized views
- Removing stored procedures

Support nightmare

Do you really need SQL?

- Your high-transaction queries are probably pretty simple once de-normalized

- A simple get / put API may meet your needs

- Looking up values for a given key is simple, fast, and scalable
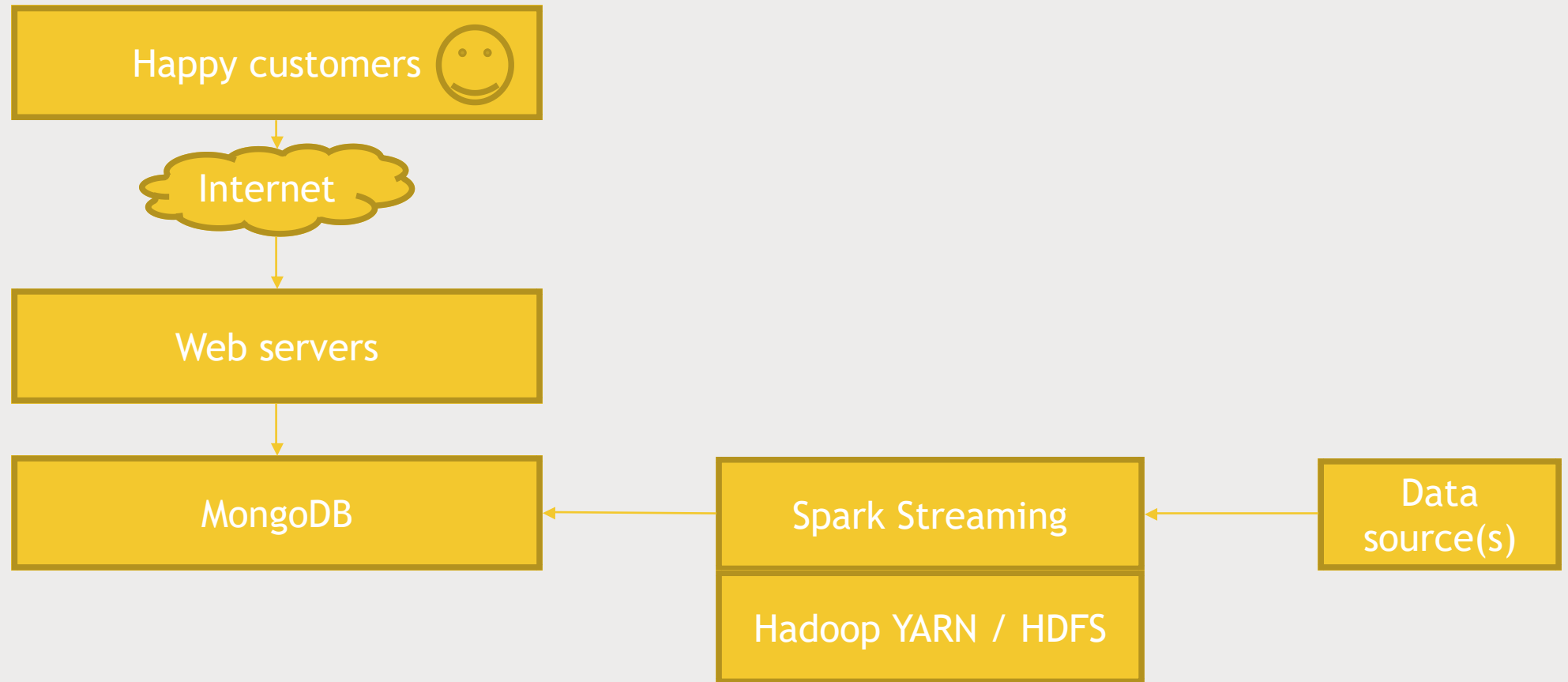
- You can do both...

Sample architecture

Use the right tool for the job

- For analytic queries, Hive, Pig, Spark, etc. work great.

- Exporting data to MySQL is plenty fast for most applications too.

- But if you work at giant scale – export your data to a non-relational database for fast and scalable serving of that data to web applications, etc.

Sample application architecture (greatly simplified!)

Happy customers ☺

Internet

Web servers

MongoDB

Spark Streaming

Hadoop YARN / HDFS

Data source(s)

Every box represents many machines!

Choose a database design that meets your usage patterns