# NoSQL databases

# Agenda

- The NoSQL Database Model
- Types of NoSQL Database Engines
- NoSQL vs RDBMS
- Popular NoSQL Database Engines

# The NoSQL Database Model

# The NoSQL database model

- NoSQL databases:
  - are a broad class of database management systems that do not use SQL as the query language
  - may not require fixed table schemas
  - usually do not support join operations

# The NoSQL database model

- NoSQL databases:
  - may not give ACID guarantees (performance, scalability and real-time nature are typically more important than consistency! )
  - typically scale horizontally
  - typically provide eventual consistency (reads may not be up-to-date but we have additional benefits such as easier load distribution and multi-data center support)

# Types of NoSQL Database Engines

# Types of NoSQL Database Engines

- Categories:
  - ✓ Big Table
  - ✓ key-value stores
  - ✓ document stores
  - ✓ graph databases
  - ✓ object databases
  - ✓ RDF databases

# Types of NoSQL Database Engines

- Key-value stores:
  - store data in key-value pairs (in either disk or memory)
  - allow for the storage of data in a schema-less way
  - basic form of API access:
    - get(key) – extract the value given a key
    - put(key, value) – create or update the value given its key
    - delete(key) – remove the key and its associated value

# Types of NoSQL Database Engines

- Key-value stores:
  - advanced form of API access (execute user-defined function on the server-side):
    - execute(key, operation, parameters) – invoke an operation on the value of a specified key
    - mapreduce(keyList, mapFunc, reduceFunc) – invoke a map/reduce function across a key range

# Types of NoSQL Database Engines

- Key-value stores:
  - Implementations: BigTable, Keyspace, LevelDB, membase, MongoDB, Apache Cassandra, Dynamo, Hibari, OpenLink Virtuoso, Project Voldemort, memcached, OpenLink Virtuoso, Oracle Coherence and more

# Types of NoSQL Database Engines

- Document stores:
  - subcategory of key-value stores (also called key-document stores)
  - designed for storing document-oriented (semi-structured data) information
  - typically assume that documents can be represented in some well-known format (XML, YAML, JSON, BSON, PDF …)

# Types of NoSQL Database Engines

- Document stores:
  - Implementations: BaseX(XML), Clusterpoint, Apache CouchDB(JSON), eXist(JSON), Jackrabbit, Lotus Notes database, IBM Lotus Domino database, IBM X Pages, MarkLogic Server (XML), MongoDB (JSON), OpenLink Virtuoso C++, SPARQL middleware, OrientDB, SimpleDB, Terrastore and more

# Types of NoSQL Database Engines

- Big table:
  - variants of Google's BigTable database
  - designed to scale on hundreds of machines (and add easily more machines)
  - when data grows beyond a predefined limit then it is typically compressed
  - Implementations: Apache Accumulo, Apache Cassandra, HBase, Hypertable, KDI, LevelDB and more

# Types of NoSQL Database Engines

- Graph databases:
  - uses graph structures to represent information
  - very suitable for data that requires graph-like queries
  - map more directly to the structure of object-oriented applications than relational databases
  - useful for storing RDF information
  - Implementations: AllegroGraph, DEX, FlockDB, InfiniteGraph, Neo4j, OpenLink Virtuoso, OrientDB, Pregel, Sones GraphDB and more

# Types of NoSQL Database Engines

- Object databases:
  - information is represented in the form of objects (as used in object-oriented programming)
  - typically targeted at particular object-oriented languages
  - provide easy persistence of objects
  - Implementations: db4o, GemStone/S, InterSystems Caché, JADE, VelocityDB and more

# Types of NoSQL Database Engines

- RDF databases:
  - typically store data in tuples
  - queries are highly optimized for RDF data
  - Implementations: Meronymy SPARQL Database Server and more

# NoSQL vs RDBMS

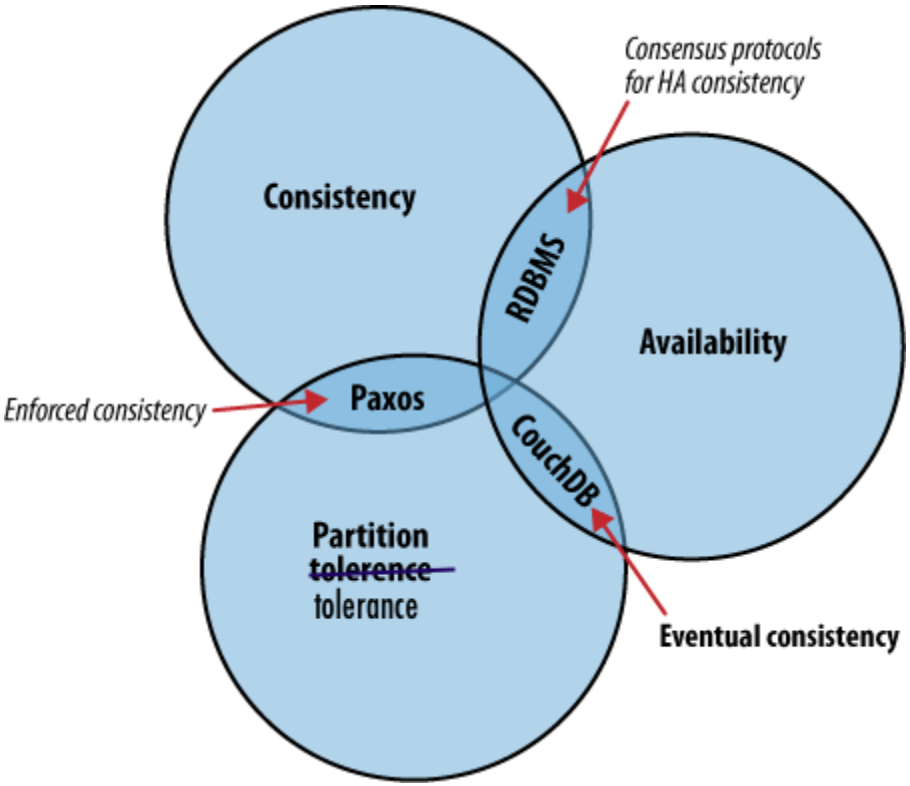
# NoSQL vs RDBMS

- Typical RDBMS implementations are tuned either for small but frequent read/write transactions or for large batch transactions with rare write accesses. NoSQL, on the other hand, can service heavy read/write workloads

- NoSQL architectures often provide weak consistency guarantees, such as eventual consistency, or transactions restricted to single data items (some systems such as AppScale provide a middleware layer to provide consistency)

# NoSQL vs RDBMS

- In terms of the CAP theorem (Consistency, Availability and Partition Tolerance):



# Popular NoSQL Database Engines

# Popular NoSQL Database Engines

- Let's see how the following popular NoSQL implementations work in practice:
  - **MongoDB** - document-store
  - **Redis** - key-value store
  - **Neo4j** - graph database
  - **Hadoop** - distributed storage and processing of Big Data

# Popular NoSQL Database Engines

- MongoDB:
  - MongoDB stores documents in binary JSON (BSON)
  - The MongoDB shell is a javascript shell
  - Logical organization of documents:

Databases

Collections (can be nested for logical separation)

Documents

Key-Value pairs

# Popular NoSQL Database Engines

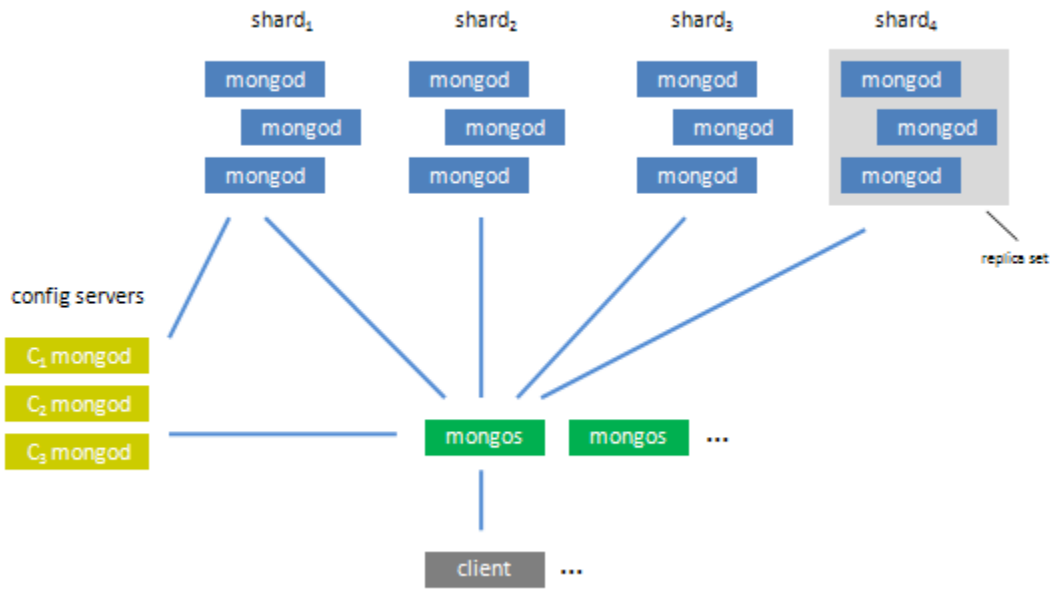
- MongoDB components:
  - mongod - core database process
  - mongos - sharding controller
  - mongo - database shell (uses interactive javascript)
  - import/export tools – mongoimport, mongoexport, mongodump, mongorestore, bsondump
  - mongofiles - the GridFS (specification for storing large files in MongoDB) utility
  - mongostat – various database statistics (e.g. number of connections)

# Popular NoSQL Database Engines

- Scalability and failover:
  - Mongo scales horizontally via an autosharding (partitioning) architecture that provides for:
    - Automatic balancing for changes in load and data distribution
    - Easy addition of new machines without down time
    - Scaling to one thousand nodes
    - No single points of failure
    - Automatic failover
  - Sharding is performed on a per-collection basis (small collections need no sharding)

# Popular NoSQL Database Engines

- Scalability and failover:




| collection | minkey | maxkey | location |
|---|---|---|---|
| users | { name : 'Miller' } | { name : 'Nessman' } | $shard_2$ |
| users | { name : 'Nessman' } | { name : 'Ogden' } | $shard_4$ |
| ... | | | |

# Popular NoSQL Database Engines

- Scalability and failover:
  - shards – each shard consists of one or more servers and stores data using mongod processes. In MongoDB sharding is the tool for scaling a system
  - replica sets - the set of servers/mongod process within the shard comprise a replica set. In MongoDB replication is the tool for data safety, high availability, and disaster recovery

# Popular NoSQL Database Engines

- Scalability and failover:
  - config servers - the config database stores all the metadata indicating the location (particular shard) of data by range. Config servers use their own replication model (not a replica set)
  - mongos servers - the mongos process can be thought of as a routing and coordination process that makes the various components of the cluster look like a single system

# Popular NoSQL Database Engines

- MongoDB common usage scenarios:
  - log data storage
  - real-time event statistics, analytics and reports
  - real-time aggregation of hierarchical data
  - storing product catalogs
  - storing comments
  - storing activity streams

# Popular NoSQL Database Engines

- Redis:
  - An open-source, networked, in-memory, key-value store
  - Data model of the database is a dictionary that maps keys to values
  - Supports high-level, atomic server-side operations like intersection, union, and difference between sets
  - Data is stored in-memory but additional facilities for data persistence are provided

# Popular NoSQL Database Engines

- Redis common usage scenarios:
  - show latest items
  - ordering of items
  - Implement expiration of items
  - Implementing real-time subscriptions
  - counting items
  - aggregating items
  - caching

# Popular NoSQL Database Engines

- Neo4j:
  - An embedded, disk-based, fully-transactional Java persistence engine that stores data in graphs rather than in tables
  - Licensed under GPL
  - Provides features for:
    - true ACID transactions
    - high availability
    - scaling to billions of nodes and relationships
    - high speed query through traversals

# Popular NoSQL Database Engines

- Neo4j use cases:
  - Network and IT Operations
  - Social Networking
  - Product-line management

# Popular NoSQL Database Engines

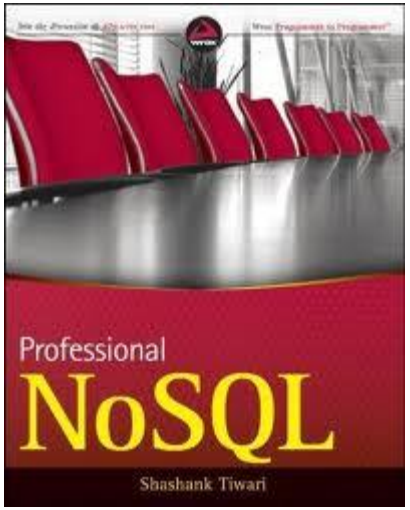
- Hadoop:
  - An open-source software framework for storage and large-scale processing of data-sets on clusters and commodity hardware
  - Provides a Hadoop file system (HDFS) for storing data in the cluster
  - Provides a Hadoop YARN - resource management system for resources in the Hadoop cluster
  - Provides Hadoop MapReduce - a programming model for large-scale data processing

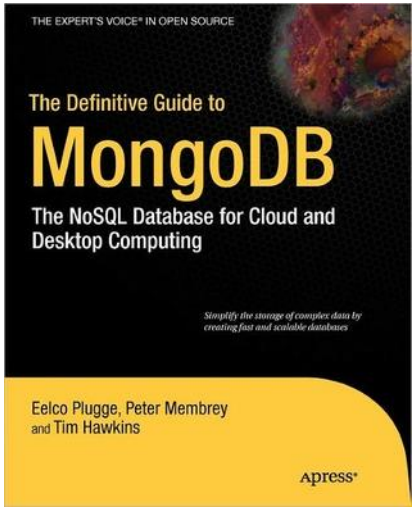
# Popular NoSQL Database Engines

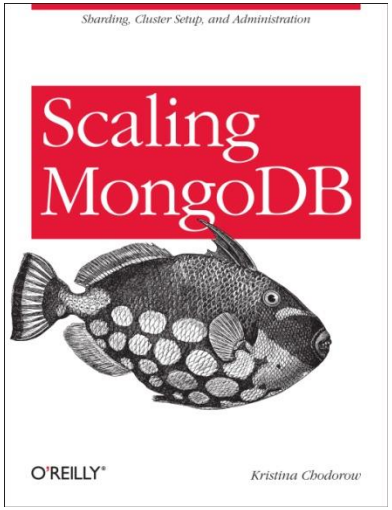
- Hadoop use cases:
  - Batch aggregations
  - Data Warehousing
  - Extract-Transform-Load (ETL)

Questions ?

# Some books


Professional
NoSQL
Shashank Tiwari


THE EXPERT'S VOICE® IN OPEN SOURCE
The Definitive Guide to
MongoDB
The NoSQL Database for Cloud and Desktop Computing
Simplify the storage of complex data by creating fast and scalable databases
Eelco Plugge, Peter Membrey and Tim Hawkins
Apress®


Sharding, Cluster Setup, and Administration
Scaling MongoDB
O'REILLY®
Kristina Chodorow

# Resources

Wikipedia's entry on NoSQL
http://en.wikipedia.org/wiki/NoSQL

Document-oriented databases
http://en.wikipedia.org/wiki/Document-oriented_database

XML database
http://en.wikipedia.org/wiki/XML_database

MongoDB official documentation
http://www.mongodb.org/display/DOCS/Home

MongoDB backups (official documentation)
http://www.mongodb.org/display/DOCS/Backups

MongoDB backup and recovery
http://jonathanhui.com/mongodb-backup-and-recovery

# Resources

MongoDB backup tools
http://www.mongodb.org/display/DOCS/Import+Export+Tools

MongoExplorer
http://www.mongodb.org/display/DOCS/Admin+UIs

Email Integration - Inbound Emails
http://ecp-alpha.cisco.com/html/index.html#url=/web/view-post/post/-/posts?postId=25327222

MongoDB architecture
http://johanlouwers.blogspot.com/2011/06/mongodb-architecture.html

# Resources

SQL to Mongo Mapping Chart
http://www.mongodb.org/display/DOCS/SQL+to+Mongo+Mapping+Chart

SQL vs NoSQL presentation
http://www.slideshare.net/skillsmatter/4alaric-snell-pym

NoSQL and MongoDB
http://www.slideshare.net/csixty4/nosql-mongodb

CouchDB - the Definitive Guide
http://guide.couchdb.org/

Introduction to MongoDB
http://www.slideshare.net/drumwurzel/intro-to-mongodb

# Resources

How graph databases can make you a superstar
http://www.slideshare.net/andres_taylor/graph-database-super-star-8079303

Graph databases, NoSQL and Neo4j
http://www.infoq.com/articles/graph-nosql-neo4j

NoSQL Patterns
http://horicky.blogspot.com/2009/11/nosql-patterns.html

Wikipedia's entry on UnQL (Unstructured Query Language)
http://en.wikipedia.org/wiki/UnQL

# Resources

'Ugly' MongoDB defies NoSQL death rumour
http://www.theregister.co.uk/2012/03/27/picking_a_no_sql_winner/

MongoDB - production deployments (contains a detailed list of concrete applications)
http://www.mongodb.org/display/DOCS/Production+Deployments

MongoDB manual
http://www.mongodb.org/display/DOCS/Manual

Overview – MongoDB interactive shell
http://www.mongodb.org/display/DOCS/Overview+-+The+MongoDB+Interactive+Shell

# Resources

MongoDB – Java tutorial
http://www.mongodb.org/display/DOCS/Java+Tutorial

MongoDB query language
http://www.mongodb.org/display/DOCS/Mongo+Query+Language

Bigtable: A Distributed Storage System for Structured Data
http://research.google.com/archive/bigtable.html