



# NoSQL

"Not Only SQL"

# Agenda

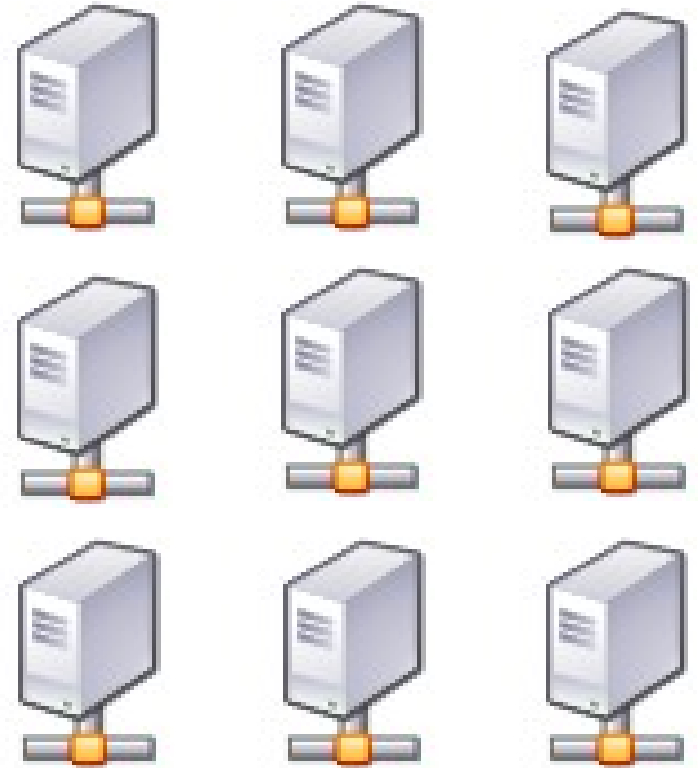
- Merkmale von NoSQL Systemen
- NoSQL System Typen
- Begriffe aus der NoSQL Welt
  - BASE, CAP, MapReduce
- Beispiele von NoSQL Systemen
  - Mit API und Live Demos
- Fazit
- Fragen



# Merkmale

- Schema frei
- Horizontal skalierbar
- Eventually consistent
- Nicht relational
- Verteilt
- Simple API

# Vertikale vs. Horizontale Skalierung





# Typen

- Document Store
- Wide Column Store
- Key Value / Tuple Store
- Graph Databases
- Object Databases
- XML Databases

# ACID vs. BASE

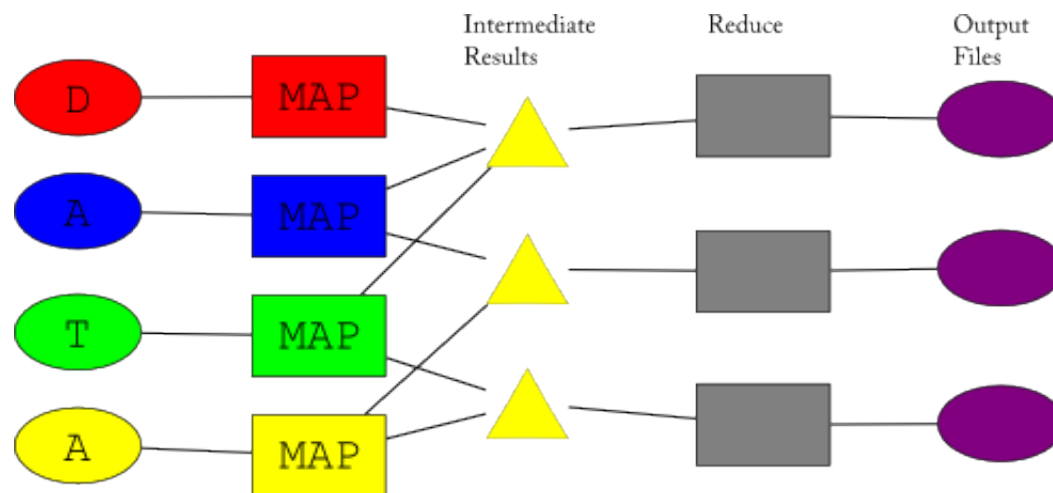
- Atomicity
    - Alles oder nichts
  - Consistency
    - Integrität
  - Isolation
    - Kapselung von Transaktionen
  - Durability
    - Persistenz aller Änderungen
- Basically Available
  - Soft-state
  - Eventual consistency
    - Zeitfenster mit Inkonsistenz

# CAP Theorem

- C – Consistency
- A – Availability
- P – Partition Tolerance
- You can only have 2 at a time
- **Make your choice!**

# Map Reduce

- Map the list of key-value pairs from one domain to another
- Group them by key
- Reduce to another list of key-value pairs





# HBase

- Unterprojekt von Apache Hadoop
- Aktuelle Version: 0.20.6
- Typ: Wide Column Store
  - „HBase uses a data model very similar to that of Bigtable. Users store data rows in labelled tables. A data row has a sortable key and an arbitrary number of columns. The table is stored sparsely, so that rows in the same table can have crazily-varying columns, if the user likes. „ - Hbase Wiki
- Geschrieben in Java
- <http://hbase.apache.org/>



# neo4j

- Typ: Graphendatenbank
- Aktuelle Version: 1.0
- Lizenz: Apache 2.0
- Geschrieben in Java
- <http://neo4j.org/>

# neo4j API

```
package org.neo4j.example;

import org.neo4j.graphdb.*;
import org.neo4j.graphdb.factory.GraphDatabaseFactory;

public class Neo4jExample {

    private static GraphDatabaseService db;

    static {
        db = GraphDatabaseFactory.getInstance().newDatabase("neo4j");
    }

    public static void main(String[] args) {
        createProperties();
        createRelationships();
        printRelationships();
    }

    private static void createProperties() {
        Node node = db.createNode(Label.withName("Person"), "name", "Neo");
        node.setProperty("age", 30);
        Index index = db.indexFor(Indexable.STRING, "name");
        Index index2 = db.indexFor(Indexable.STRING, "age");
    }

    private static void createRelationships() {
        Node node = db.createNode(Label.withName("Person"), "name", "Neo");
        Node node2 = db.createNode(Label.withName("Person"), "name", "Neo");
        Relationship relationship = db.createRelationship(node, node2, RelationshipTypes.ATTENDED);
    }

    private static void printRelationships() {
        Relationship relationship = db.relationships().first();
        System.out.println("Relationship: " + relationship);
    }
}
```

**DEMO**

void java.io.PrintStream.println(String s)

Prints a String and then terminate the line. This method behaves as though it invokes `println(String s, "r\n")`.

Parameters:  
s The String to be printed.

Errors	Warnings	Others
0	3	0
neo4j-example	neo4j-example	neo4j-example
neo4j-example	neo4j-example	neo4j-example
neo4j-example	neo4j-example	neo4j-example



# MongoDB

- Typ: Dokumentenorientiert
- Speicherformat: BSON (Binary JSON)
- Aktuelle Version: 1.4.4
- Lizenz: GNU AGPL
- Geschrieben in C
- <http://www.mongodb.org/>

# MongoDB API

The image shows a screenshot of an IDE (Eclipse) with a Java project named 'MongoDBExample'. The main editor displays the following code:

```
Wave waveProperty = (Wave) mongoClient.getDatabase("waves").getCollection("waves").findOne();
Index indexWave = mongoClient.getDatabase("waves").getCollection("waves").getIndexes().get(0);

// Create a new relationship
MongoClient mongoClient = MongoClientFactory.createClient("mongodb://localhost:27020");
MongoDatabase database = mongoClient.getDatabase("waves");
MongoCollection collection = database.getCollection("relationships");

// Create a new relationship
MongoDocument document = new BasicDBObject("wave_id", waveProperty.getId(), "attendee_id", attendee.getId());
collection.insertOne(document);

// Print the relationship
System.out.println("Created relationship: " + document);
System.out.println("Attendee: " + attendee);
System.out.println("Wave: " + waveProperty);

// Print the relationship
System.out.println("Created relationship: " + document);
System.out.println("Attendee: " + attendee);
System.out.println("Wave: " + waveProperty);

// Print the relationship
System.out.println("Created relationship: " + document);
System.out.println("Attendee: " + attendee);
System.out.println("Wave: " + waveProperty);
```

A large, semi-transparent "DEMO" watermark is overlaid on the code. A tooltip for the `void java.io.PrintStream.println(String s)` method is visible over the `println` call.

The IDE interface includes a Project Explorer on the left, an Outline view on the right, and a Problems view at the bottom showing a list of errors:

Errors	Severity	Details	Location	Type
1	Error	The project org.mongodb.hadoop.conf.Configuration is never used	MongoExample\src\main\java\org\mongodb\hadoop\conf\...	Java Problem
2	Error	Could not load class org.mongodb.hadoop.conf.Configuration	MongoExample\src\main\java\org\mongodb\hadoop\conf\...	Java Problem
3	Error	Could not load class org.mongodb.hadoop.conf.Configuration	MongoExample\src\main\java\org\mongodb\hadoop\conf\...	Java Problem

# Voldemort

- Typ: Key Value Store
- Aktuelle Version: 0.81
- Lizenz: Apache License 2.0
- Geschrieben in Java
- <http://project-voldemort.com/>

# Voldemort API

The screenshot shows an IDE window with the following code:

```
Wave waveProperty (KEY_TITLE, "Google Wave");
Wave waveProperty ("maxAttendees", 30);
Index index(wave, KEY_TITLE, "Google Wave");

job(wave, wave.createRelationShipTo (android, MyRelationShipTypes.ATTENDEE);
job(wave, wave.createRelationShipTo (facebook, MyRelationShipTypes.ATTENDEE);
job(wave, wave.createRelationShipTo (google, MyRelationShipTypes.ATTENDEE);
job(wave, wave.createRelationShipTo (yahoo, MyRelationShipTypes.ATTENDEE);

myCustomRelationShipTo (facebook, MyRelationShipTypes.ATTENDEE);
myCustomRelationShipTo (google, MyRelationShipTypes.ATTENDEE);
myCustomRelationShipTo (yahoo, MyRelationShipTypes.ATTENDEE);

myAttendee ();

// ... more code ...

System.out.println ("Found for " + wave.getKey());
System.out.println ("void java.io.PrintStream.println(String s)");
Index index (Prints a String and then terminate the line. This method behaves as though it invokes write
System.out.println ("Printing out the print()");
// ... more code ...
```

A large, semi-transparent "DEMO" watermark is overlaid on the code. A tooltip for the `void java.io.PrintStream.println(String s)` method is visible, showing its parameters and description.

The IDE interface includes a Project Explorer on the left, an Outline on the right, and a Problems window at the bottom showing a list of errors.

Errors	Severity	Details	Row	Col	Location	Type
Warning (3 items)						
The import org.apache.hadoop.conf.Configuration is never used	Warning	File: example...	line 6		java problem	
Could not load configuration element: Job 1.1. There are no Attachments	Warning	File: example...	line 10		java problem	
Could not load configuration element: HADOOP_CLASSPATH_DIRECTORY	Warning	File: example...	line 11		java problem	





# Weitere NoSQL Systeme

- Riak
- CouchDB
- Cassandra
- Redis
- Berkley DB
- ...

# Fazit

- Keine einheitliche API
  - Teilweise Plugins für JPA
- Blick über den Tellerrand
  - Denken jenseits von relationalem Datenmodell
- Das richtige Werkzeug für den richtigen Zweck

# Nächste Termine

- 24.08.2010 um 20:00 Uhr:
  - Java Champion Adam Bien "Stop Talking - Start Hacking ...with Java EE 6"
  
- 06.10.2010 um 18:30 Uhr:
  - Developer Evangelist Jochen Hiller "Android-Apps selbst schreiben: Techniken, Tools und Tipps"





Fragen?