Jihočeská univerzita
v Českých Budějovicích
University of South Bohemia
in České Budějovice

ústav
aplikované
informatiky

Přírodovědecká
fakulta
Faculty
of Science

ADVANCED DATABASE SYSTEMS
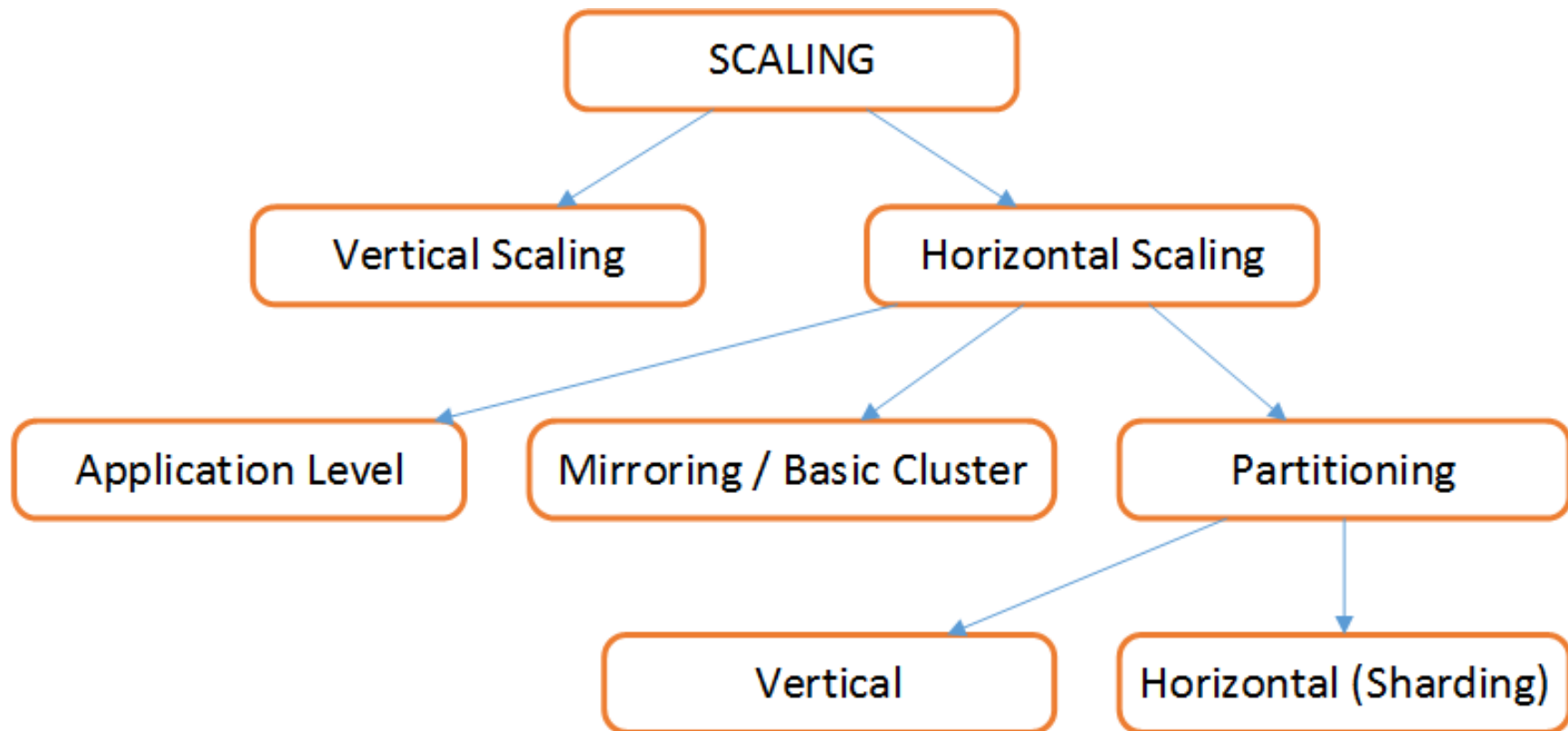
Distributed database systems and NoSQL

Mgr. Jakub Geyer

Content

> ## Part 2
>> ### Revision
>> ### CAP (Brewer's Theorem)
>>> #### ACID x BASE
>>> #### Transactions
> ### Practice 3: Indexes
> ### Practice 4: Document-Oriented database (RavenDB)

Revision

Revision

- [ ] What are disadvantages/limits of RDBMS?

- [ ] What are the scaling options for databases?

- [ ] What are the horizontal scaling options for database systems?

- [ ] How does the sharding work? What are the common data sharding strategies?

- [ ] What are the characteristics of NoSQL databases?

- [ ] What are 4 types of NoSQL databases? How they work?
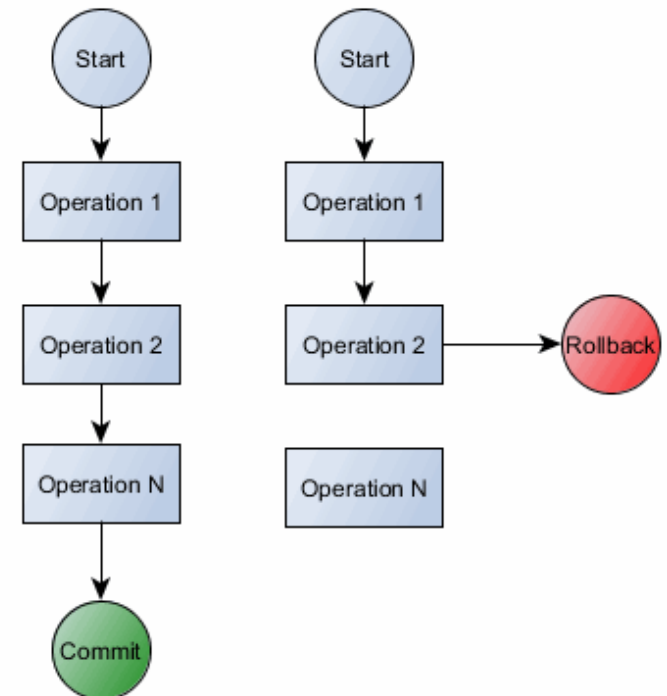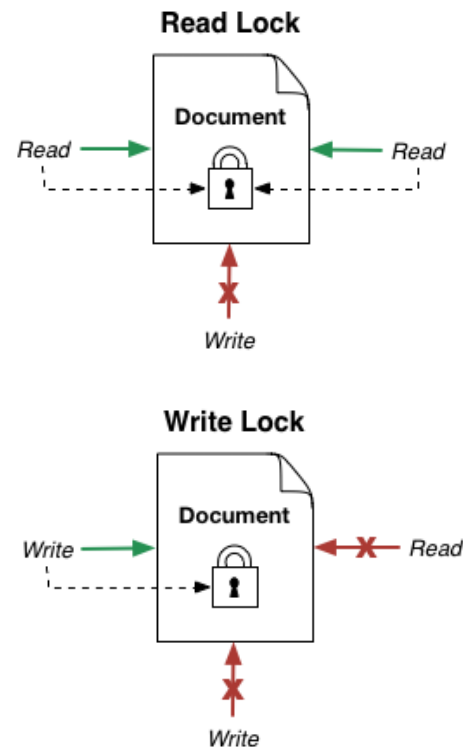 Key-Value; Column-Oriented; Document-Oriented; Graph Database

DB Transactions

ACID model

Atomicity

Consistency

Isolation

Durability

Which of these features most often do not meet NoSQL databases?

DB transactions

BASE model

ACID   BASE

Basic **A**vailability

Soft State

Eventual Consistency

It is <u>not true</u> that NoSQL databases cannot meet ACID!
However, be careful about claims that specific NoSQL database meets ACID. They often only meets the requirement for data, not for indexes.
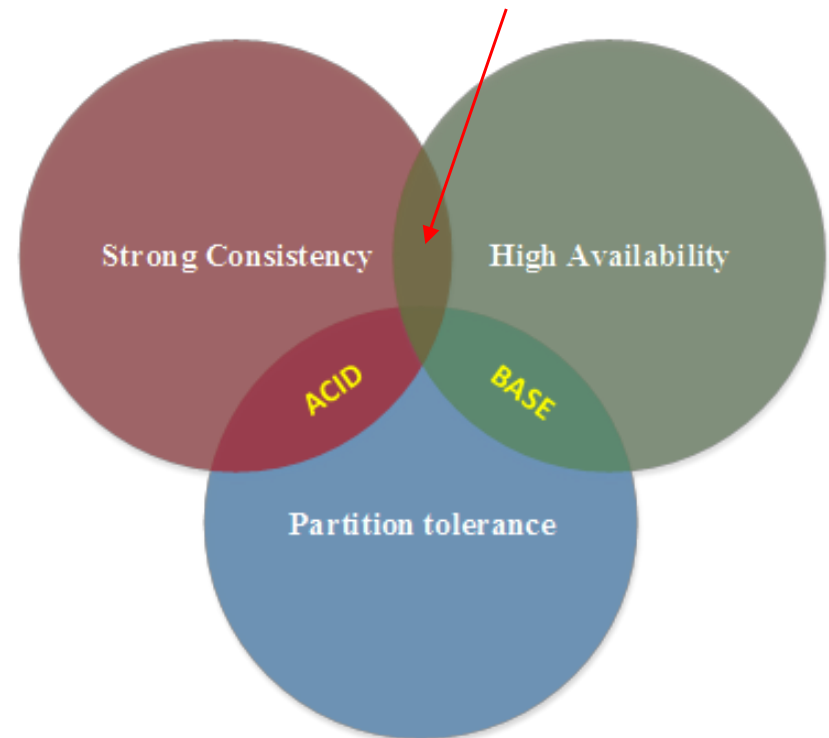
CAP Theorem

= Brewer's Theorem

CA / CP / AP

Only 2 of 3 are possible at once:

Strong **C**onsistency

High **A**vailability

Partition Tolerance

What belongs here?

Strong Consistency · High Availability · Partition tolerance · ACID · BASE

This point of view is obsolete!

CAP Theorem

CAP pro distributed systems:
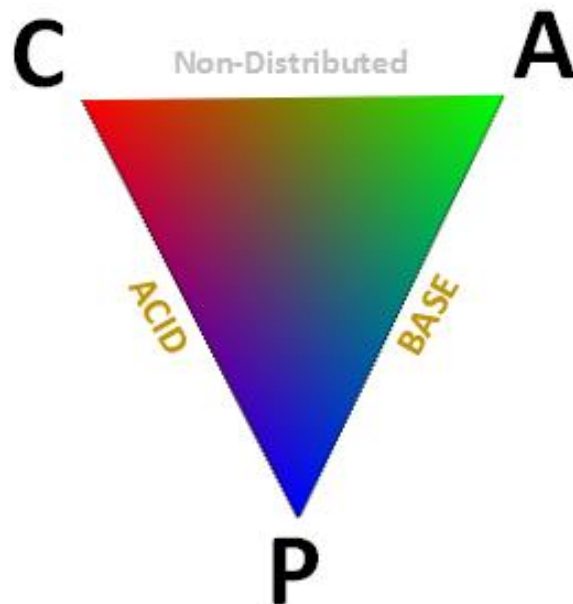
~~What is the best database? Relational, object-oriented, key-value based, graph?~~

What are the specific requirements?
- Data reliability?
- Working with a large amount of data?
- Read queries speed?
- Lots of writes?
- Complex relationships in data?
- Great data diversity?
- Different views of data? (reporting)
- Transactions?

Indexes

- Significantly affect performance in both RDBMS and NoSQL!!!

- NoSQL databases can often be queried <u>only</u> by indexed values.

- Enables other advanced features (e.g. Map-Reduce, text analyzers, scoring, etc.).

What can usually be indexed in each database type?
(Relational; Key-Value; Column-Oriented; Document-Oriented; Graph Database)

Indexes

Practice 3:

Advanced indexing and querying using **Apache Lucene**.

https://lucene.apache.org/core/

http://lucene.apache.org/core/2_9_4/queryparsersyntax.html

Be careful with *Range* - can work with both numeric values and String (alphabetically).

NoSQL databases

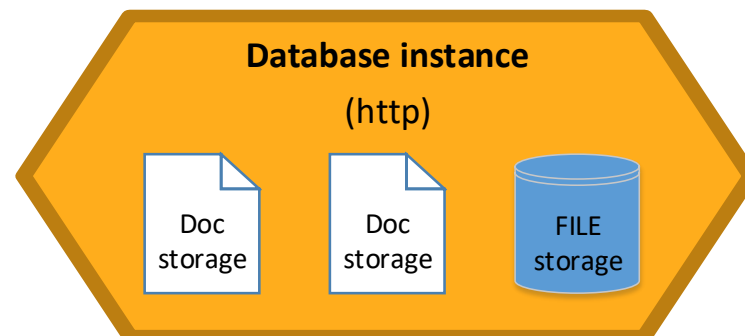## Practice 4:

Implementation of simple database using Document-Oriented database **RavenDB**.

https://ravendb.net/

https://ravendb.net/docs/article-page/3.0/csharp

https://azure.microsoft.com/en-us/blog/ravenhq-now-available-in-the-azure-store/

RavenDB

- Open-Source document-oriented database written in C#.

- Objects (documents) stored as JSON.

- 2 storage types: document storage, file storage

- Storage can be used both separately or as part of a sharding cluster (created and maintained by client application -> no management server required).


```
<Document>
Key
{
  "customerid": "fc986e48ca6"
  "customer":
  {
  "firstname": "Pramod",
  "lastname": "Sadalage",
  "company": "ThoughtWorks"
  "likes": [ "Biking","Photography" ]
  }
  "billingaddress":
  { "state": "AK",
      "city": "DILLINGHAM",
      "type": "R"
  }
}
```

Database instance

(http)

Doc storage | Doc storage | FILE storage

Thank you for your attention!