Object Oriented Databases

www.softacad.bg

Who's talking?

- Nikolay Tomitov – Co-CEO at SoftAcad learning center.
- Danail Alexiev – Software Consultant at Axway, lector at SoftAcad learning center.

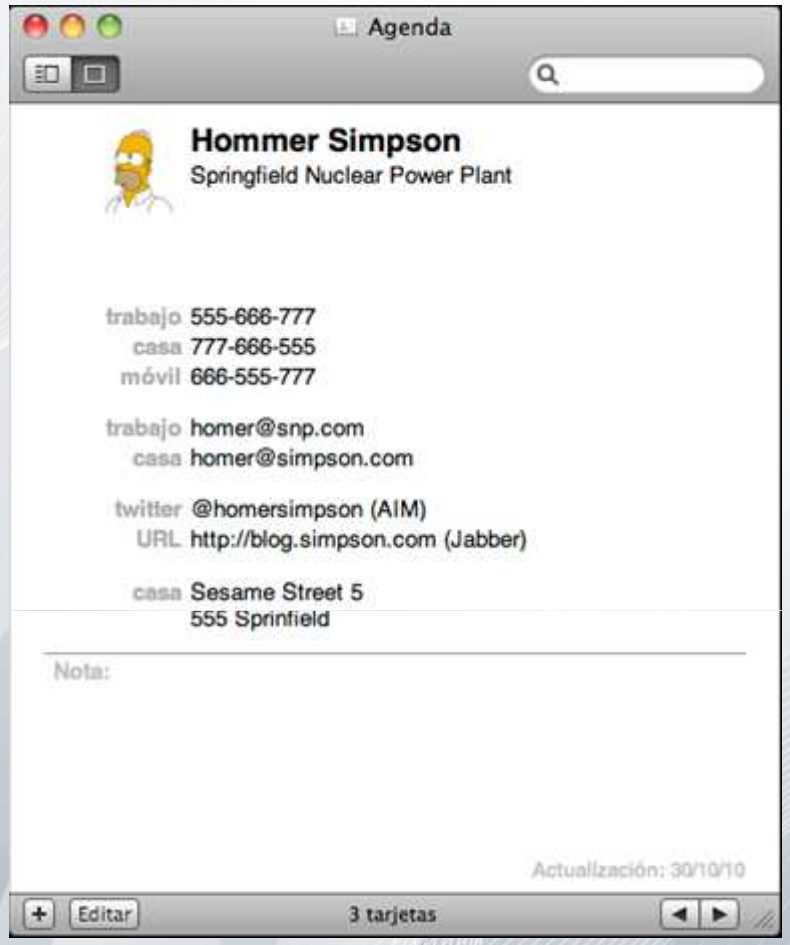
www.softacad.bg

Main Topics

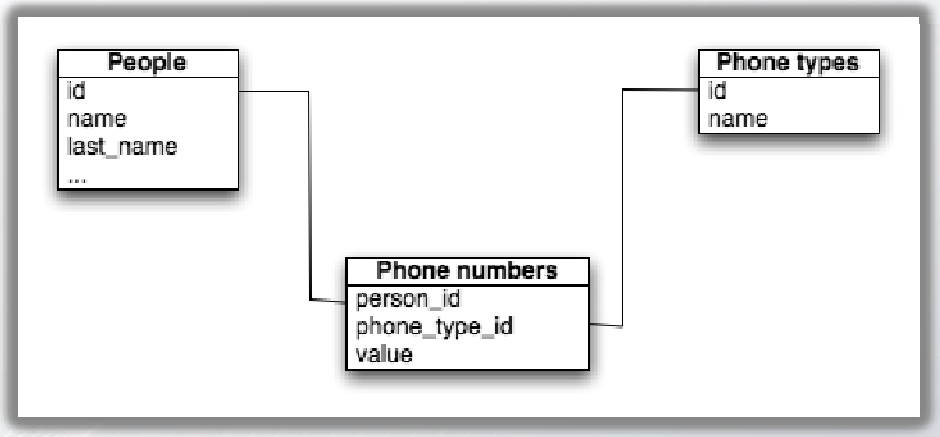
1. Document oriented databases
2. Introduction to JSON data format
3. Meet CouchDB
4. Meet MongoDB
5. CouchDB vs. MongoDB
6. Demo

Schema-free Document Oriented Database

Documental Databases



Relational Databases



www.softacad.bg

JSON Data Format

```
{
    "day": [ 2010, 01, 23 ],
    "products": {
        "apple": {
            "price": 10
            "quantity": 6
        },
        "kiwi": {
            "price": 20
            "quantity": 2
        }
    },
    "checkout": 100
}
```

Meet CouchDB

www.softacad.bg

Topics

1. Introduction to CouchDB
2. Create, Update, Remove, Query Operations
3. Views & Aggregation
4. Concurrency
5. Replication
6. Pros and cons
7. Resources

Main features

- Data is stored and returned in JSON format
- Queried via HTTP RESTful API
- Index building language: Javascript
- Simple and intuitive interface

Schema-less DB

```
{
        "_id": "debd7e7385464f4874dd2a38043f7825",
        "_rev": "3-839e43865b653a1ed73c3d21cc17c5dd",
        "name": "Homer",
        "last_names": ["Simpson", "Duff"],
        "phone_numbers":
        {
                "mobile": "555-666-777",
                “work": "666-777-555“,
                "bar": "777-666-555“
        },
        "interests": ["beer", "donuts", "couches"]
}
```

Key

```
{
    "day": [ 2010, 01, 23 ],
    "products": {
        "apple": {
            "price": 10
            "quantity": 6
        },
        "kiwi": {
            "price": 20
            "quantity": 2
        }
    },
    "checkout": 100
}
```

Restful HTTP API

- Create:
  - HTTP PUT /db/test
- Read:
  - HTTP GET /db/test
- Update:
  - HTTP PUT /db/test
- Delete:
  - HTTP DELETE /db/test

Querying the database

- Views: the way to arrange data to answer our questions

- Method to build views:
  Incremental MapReduce using Javascript

DEMO Map Reduce

www.softacad.bg

Example

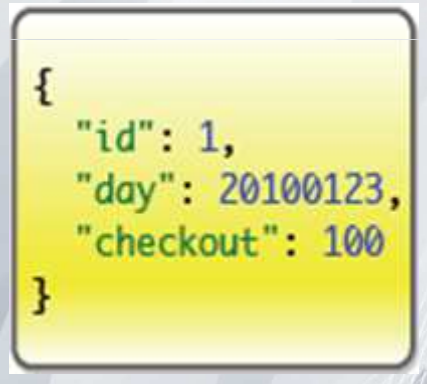
```
{
  "id": 1,
  "day": 20100123,
  "checkout": 100
}
```

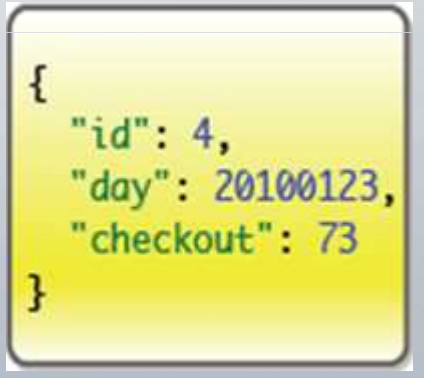
```
{
  "id": 2,
  "day": 20100123,
  "checkout": 42
}
```

```
{
  "id": 3,
  "day": 20100123,
  "checkout": 215
}
```

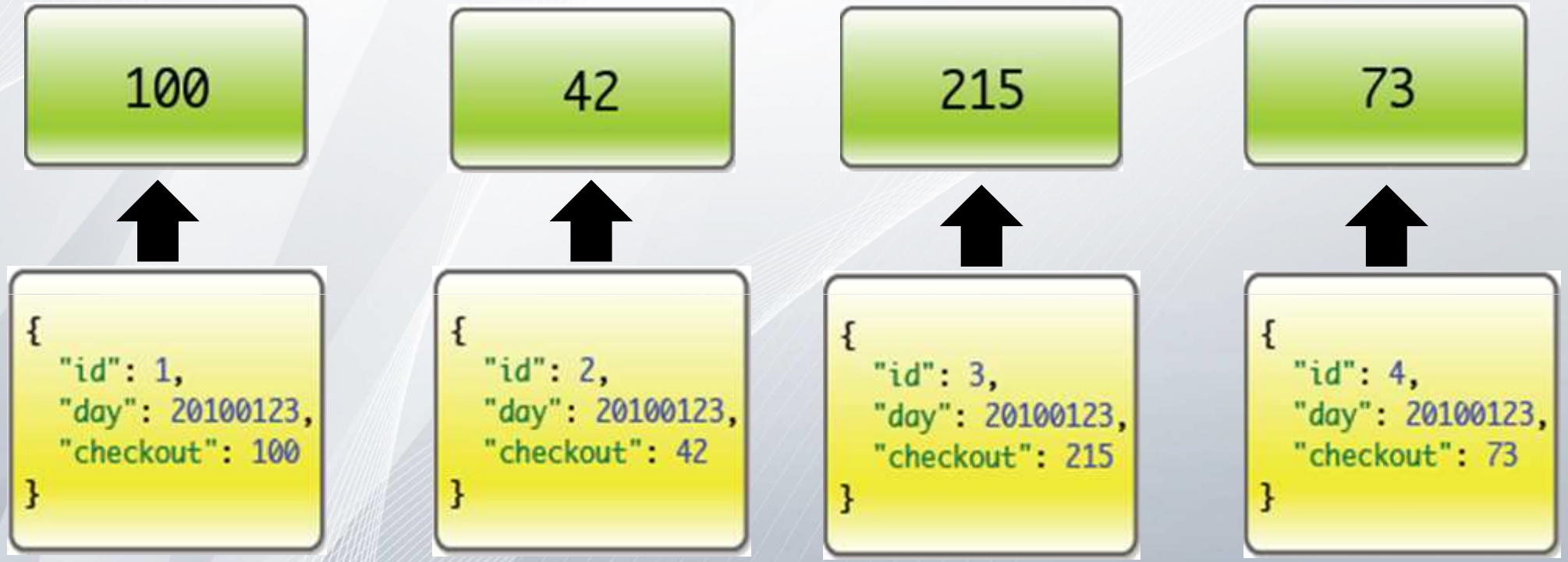
```
{
  "id": 4,
  "day": 20100123,
  "checkout": 73
}
```

How to find sum(checkouts) ?

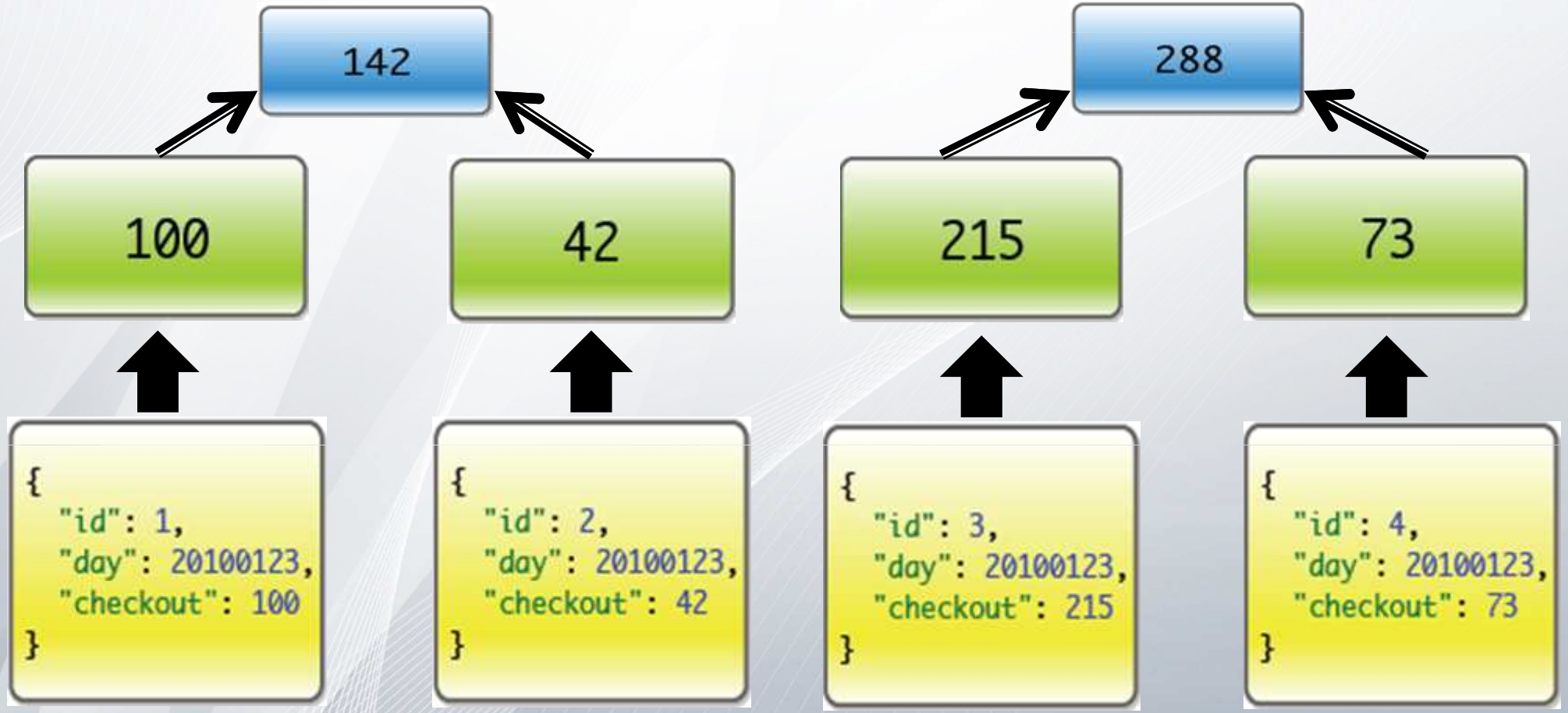




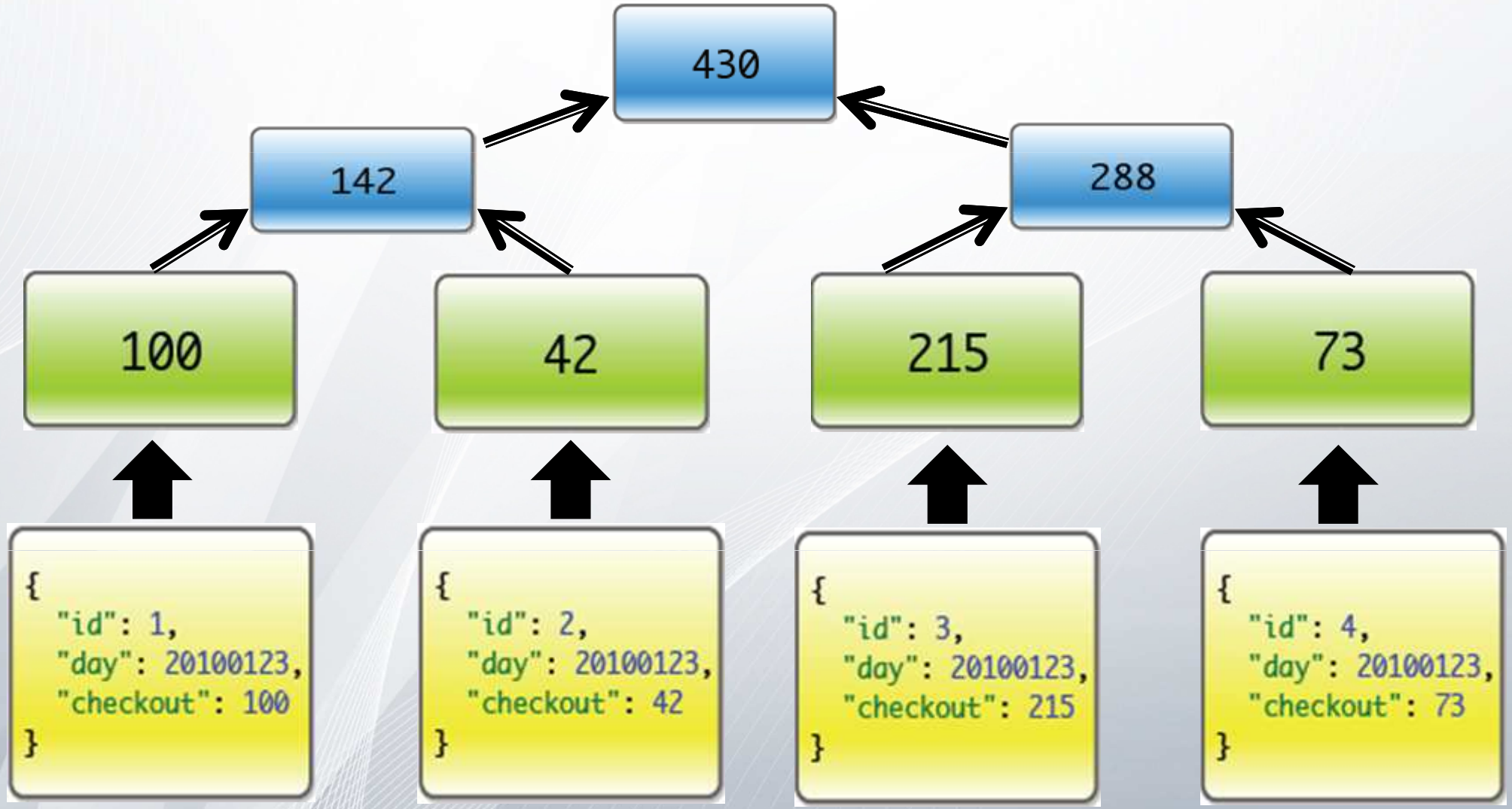
Simple Map : emit(checkout)



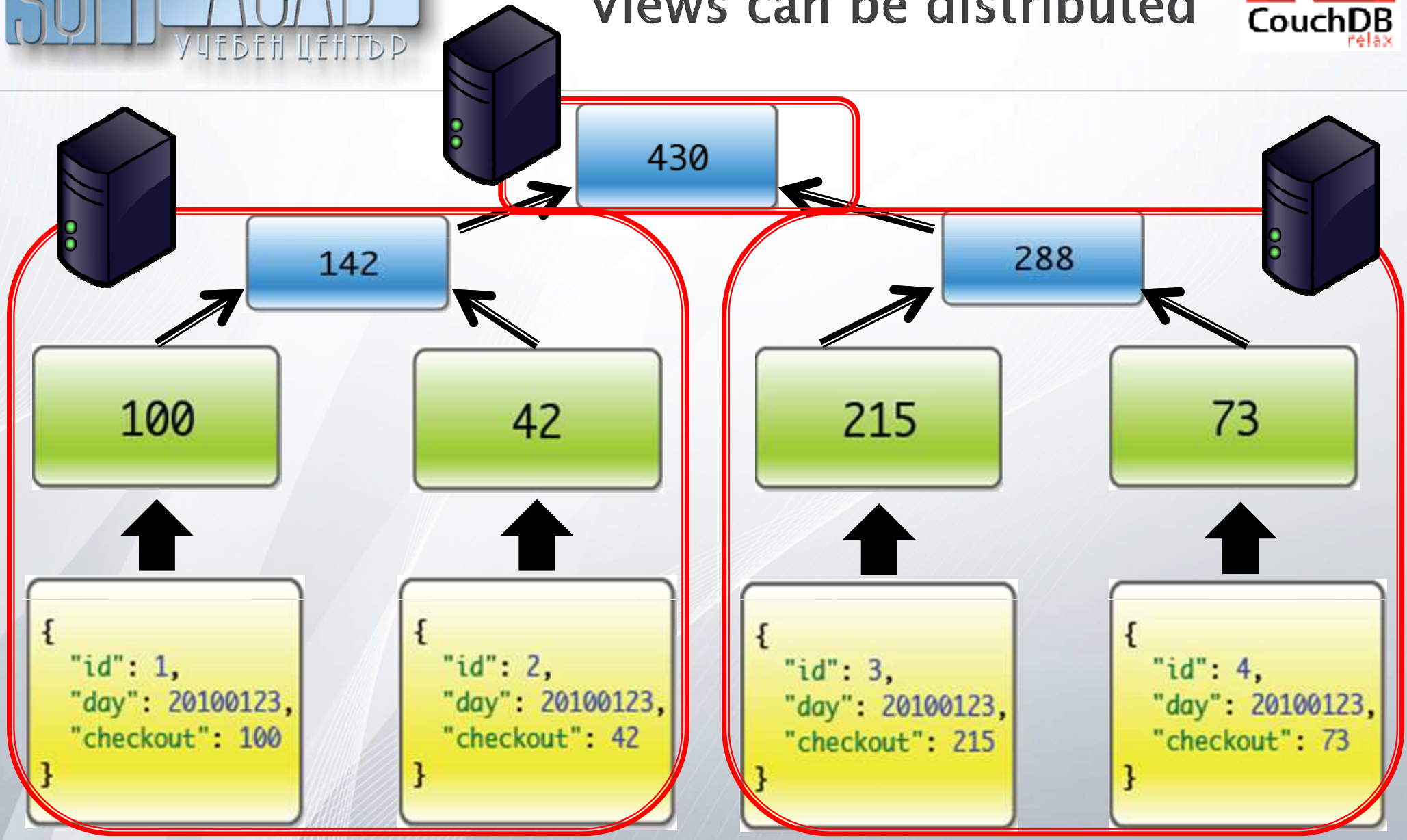
Simple Reduce : sum(checkouts)



Simple Reduce : sum(checkouts)



Views can be distributed



Views

- Views are useful for many purposes:
  - Filtering the documents in your database to find those relevant to a particular process.
  - Extracting data from your documents and presenting it in a specific order.
  - Building efficient indexes to find documents by any value.
  - Use these indexes to represent relationships among documents.
  - With views you can make all sorts of calculations on the data in your documents.

Typical document(2)

```
{
        "_id": "debd7e7385464f4874dd2a38043f7825",
        "_rev": "3-839e43865b653a1ed73c3d21cc17c5dd",
        "name": "Homer",
        "last_names": ["Simpson", "Duff"],
        "phone_numbers":
        {
                "mobile": "555-666-777",
                "work": "666-777-555",
                "bar": "777-666-555"
        },
        "interests": ["beer", "donuts", "couches"]
}
```

Concurrency

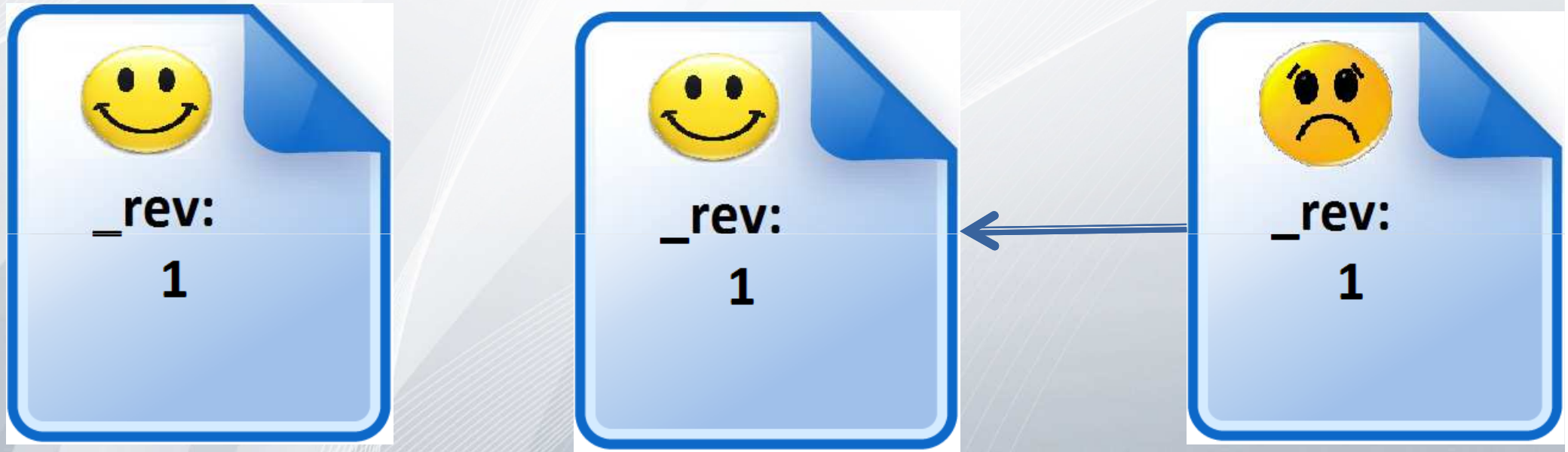
Temenujka

CouchDB

Sofroniy

Concurrency

Temenujka CouchDB Sofroniy



Concurrency

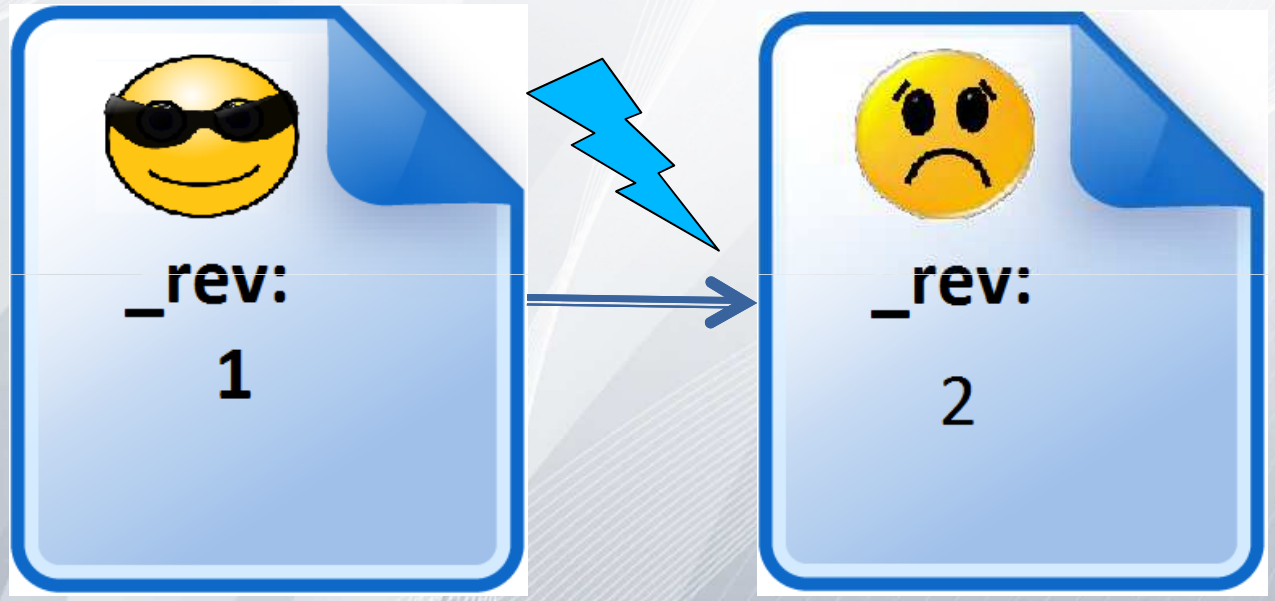
Temenujka

CouchDB

Concurrency

Temenujka

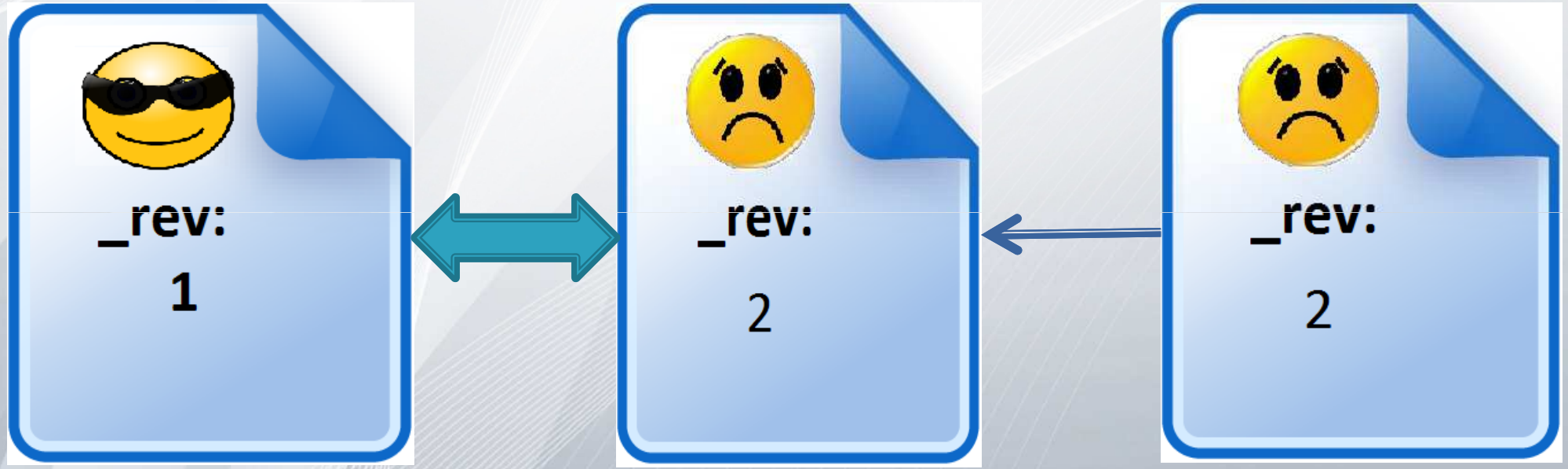
CouchDB



Concurrency

Temenujka

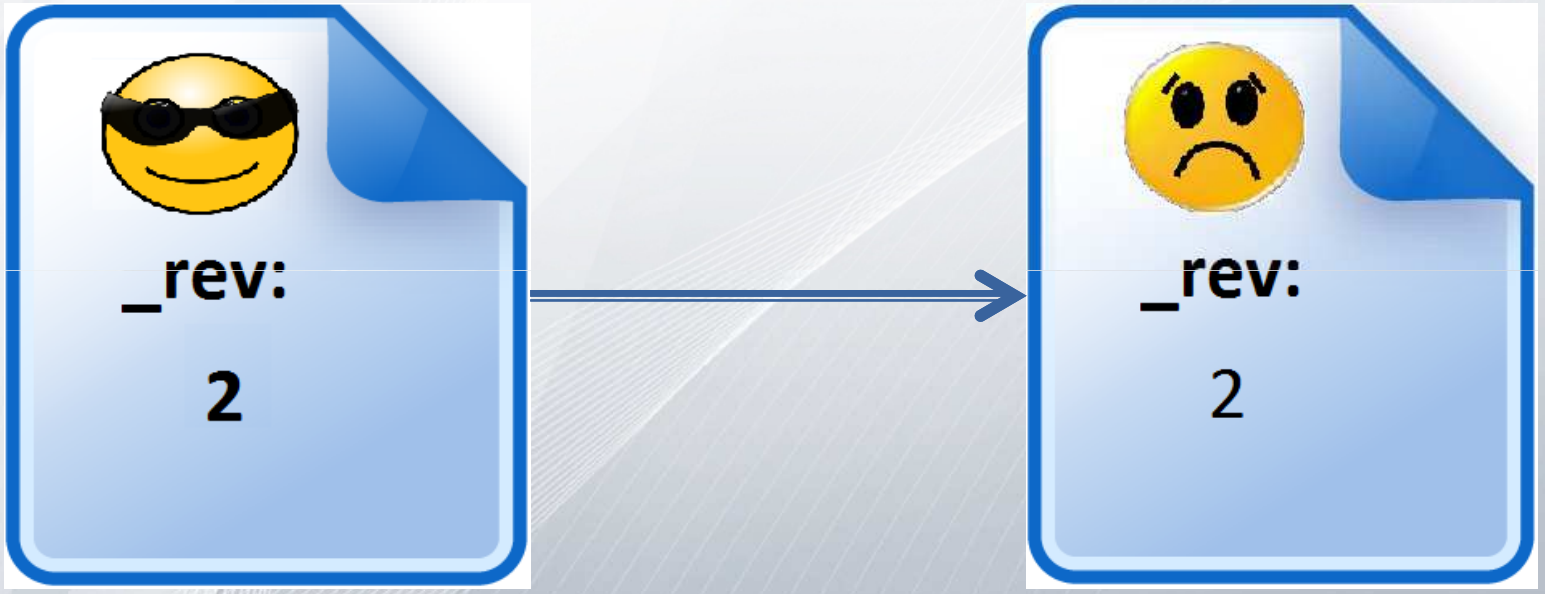
CouchDB



Concurrency

Temenujka

CouchDB

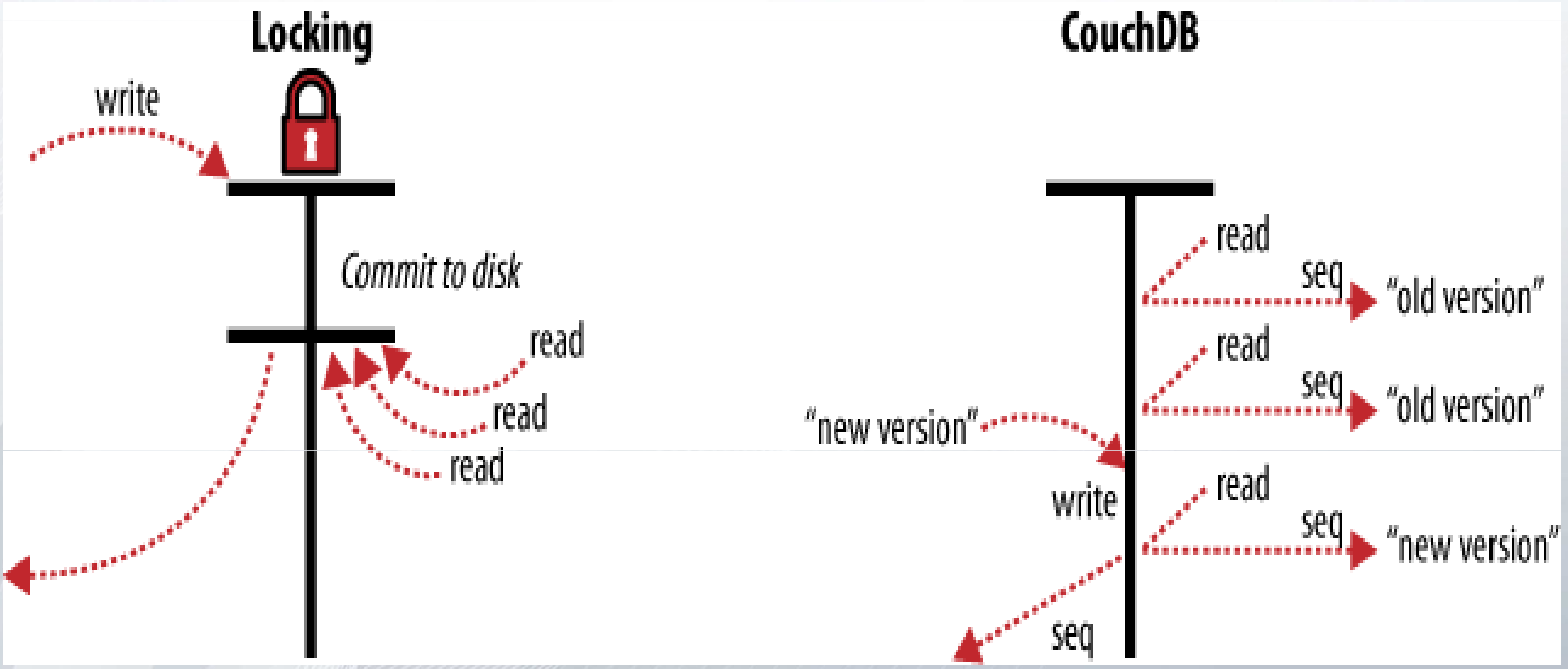


Concurrency

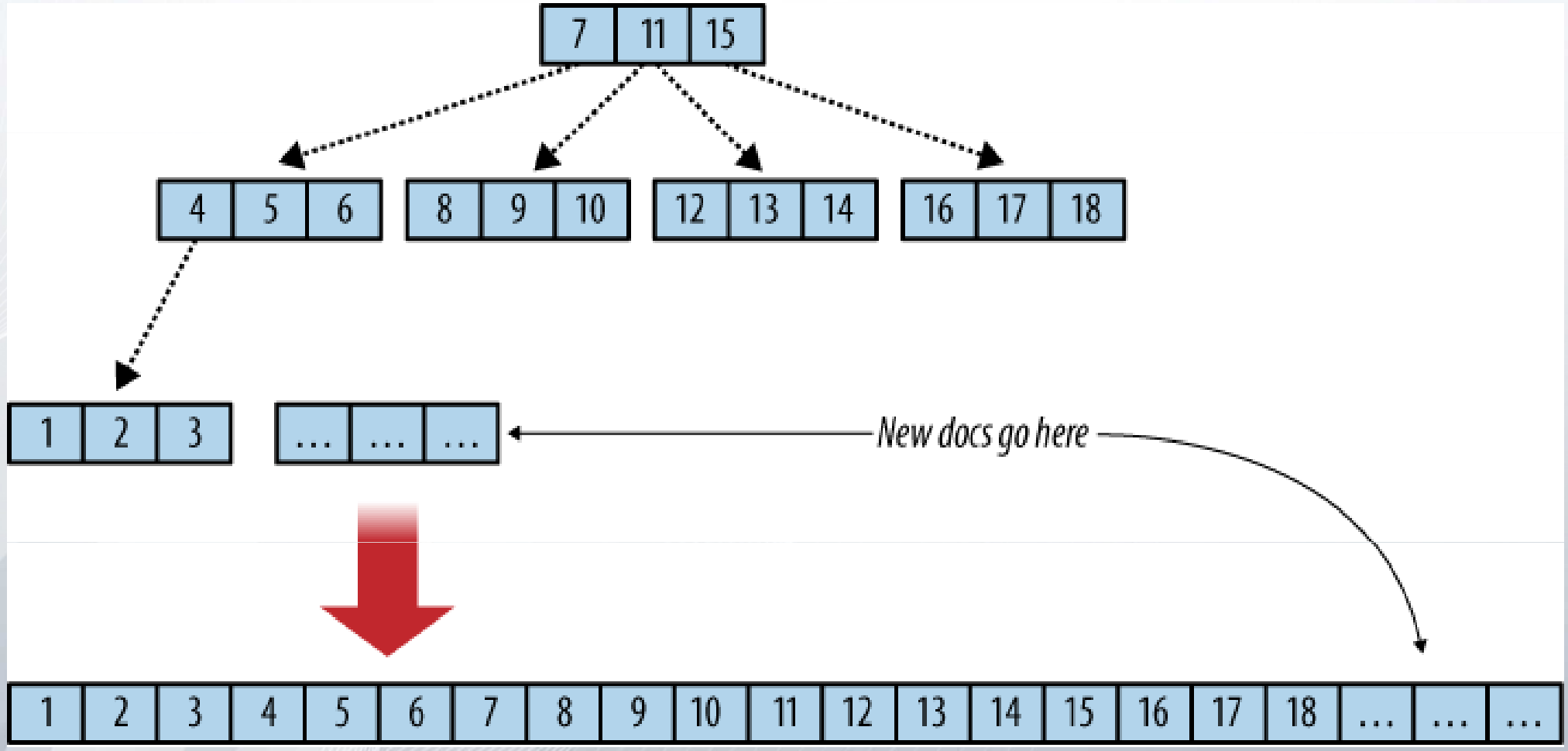
CouchDB

MVCC

- MVCC means no locking !



Under the hood



Replication

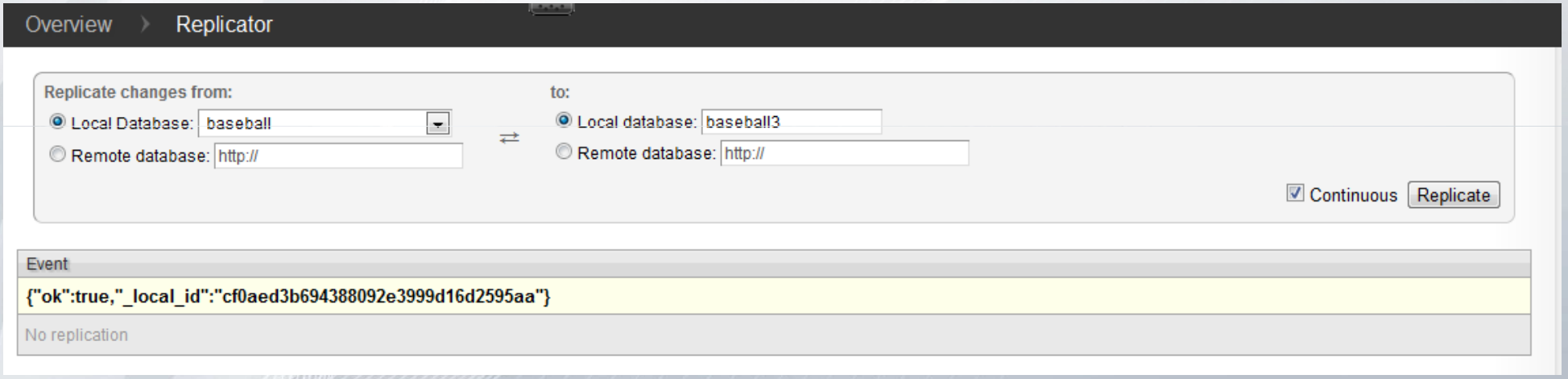
- What is replication ?
- It could be bidirectional
- Offline by default
- Continious replication

How to start replication?

- POST /_replicate

{"source":"database",
"target":"http://example.org/database"}

- By using Admin UI



Why CouchDB ?

- Fault tolerant (it's written in Erlang)
- Intuitive REST/HTTP interface
- Easy replication
- Deep structures and no schemas
- Stable and scalable, so it is successful in production
- The only database solution optimized for mobile

Why not CouchDB ?

- It's too new
- There is no security model
- Too slow for building views
- Doesn't support fancy SQL queries

I want more!

- Get it from here:

http://couchdb.apache.org/downloads.html

- Case studies :

http://www.couchbase.com/customers/case-studies

- Practice Map Reduce:

http://blog.mudynamics.com/wp-content/uploads/2009/04/icouch.html

Meet MongoDB

www.softacad.bg

Topics

1. Introduction to MongoDB
2. Create, Update, Remove Operations
3. Querying MongoDB
4. Concurrency
5. Indexes
6. Aggregation
7. Replication (Sharding)
8. Pros and cons

Intro

- Scale out rather than scale down
- Replace the row with the more flexible document
- Built for speed
- Drivers available for large range of programming languages
- Almost no administration needed

CRUD

- Create
- Update
- Delete

Querying MongoDB

- `find` and `findOne`
- Query Criteria and Conditionals
- Type – Specific Queries
- `$where`

Concurrency

- Database level locking
- Yielding

Indexes

- Almost identical to relational database indexes
- Geospatial indexes

Aggregation

- count
- distinct
- group
- Map/Reduce

Replication

- Master-Slave Replication
- Sharding

Pros and Cons

- Pros: scalable, fast, flexible, almost administration free
- Cons: no atomic operations, no validation

CouchDB vs. MongoDB

- MVCC (Multiversion concurrency control)
- Horizontal scalability
- Query Expression
- Atomicity
- Durability
- Map Reduce
- Javascript
- REST
- Performance
- Use cases

Questions?

www.softacad.bg

Thank you

nikolay.tomitov
@softacad.bg

danail.alexiev
@softacad.bg

www.softacad.bg