

# Geodata and CouchDB

Volker Mische

Free and Open Source Software for Geospatial  
20–23. October 2009  
Sydney

# CouchDB

# Documents

# Document

---

```
{
  "_id": "950da89b4748cc6d08bc2f86fa2860c9",
  "_rev": "3-77f17a55f6ab11f7f6668e63a75f2281",

  "name": "Station-2942",
  "date": "2009-10-20",
  "location": [140.39583, -37.48272],
  "state": "SA",
  "temperature": 18,
  "rainfall": 3
}
```

# Schema Free

# Schema Free

---

```
{
  "_id": "950da89b4748cc6d08bc2f86fa2860c9",
  "_rev": "3-77f17a55f6ab11f7f6668e63a75f2281",

  "name": "Station-294",
  "date": "2009-10-20",
  "location": [140.39583, -37.48272],
  "state": "SA",
  "temperature": 18,
  "rainfall": 3,

}
```

# Schema Free

---

```
{
  "_id": "950da89b4748cc6d08bc2f86fa2860c9",
  "_rev": "3-77f17a55f6ab11f7f6668e63a75f2281",

  "name": "Station-294",
  "date": "2009-10-20",
  "location": [140.39583, -37.48272],
  "state": "SA",
  "temperature": 18,
  "rainfall": 3,
  "atmospheric_pressure": 1021
}
```

# RESTful HTTP API

---

- Create:
  - ▶ If you know the ID upfront:  
HTTP PUT /db/Station-2942\_001
  - ▶ else:  
HTTP POST /db
- Read:  
HTTP GET /db/Station-2942\_001
- Update:  
HTTP PUT /db/Station-2942\_001
- Delete:  
HTTP DELETE /db/Station-2942\_001



# Concurrency

---

- Highly concurrent
- Erlang
- MVCC (Multiversion concurrency control)

# Updating a Document

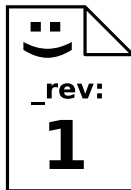
---

**\_rev**

# Updating a Document

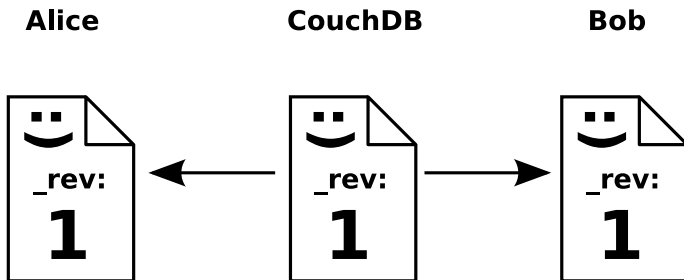
---

## CouchDB



# Updating a Document

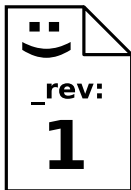
---



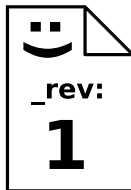
# Updating a Document

---

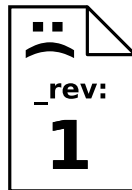
**Alice**



**CouchDB**



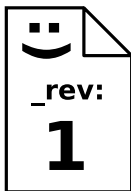
**Bob**



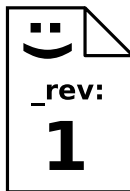
# Updating a Document

---

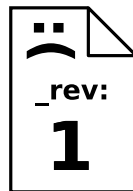
**Alice**



**CouchDB**



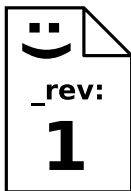
**Bob**



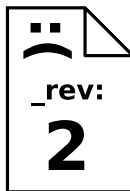
# Updating a Document

---

**Alice**



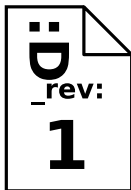
**CouchDB**



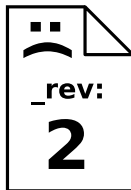
# Updating a Document

---

**Alice**



**CouchDB**



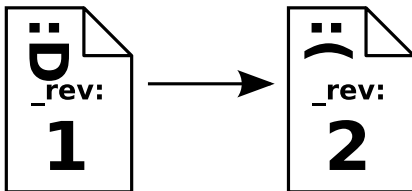


# Updating a Document

---

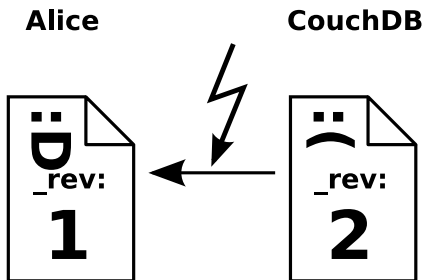
**Alice**

**CouchDB**



# Updating a Document

---

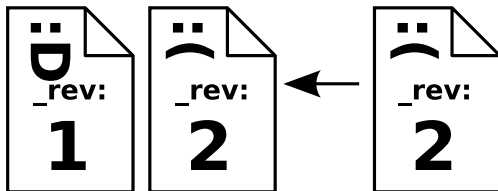


# Updating a Document

---

**Alice**

**CouchDB**



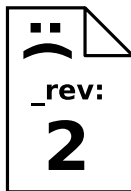
# Updating a Document

---

**Alice**



**CouchDB**

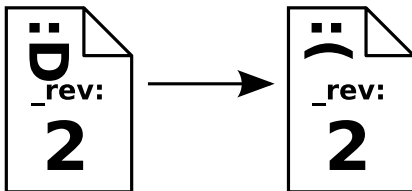


# Updating a Document

---

**Alice**

**CouchDB**



# Updating a Document

---

## CouchDB



# Views

# MapReduce



# Map function

---

```
function(doc) {  
  // emit(keys, value)  
  emit(...);  
}
```

# Map function

---

```
function(doc) {  
  // emit(keys, value)  
  emit([doc.name, doc.date], doc.rainfall);  
}
```

Keys	Value
...	...
Station-1873, 2009-10-19	2
Station-1873, 2009-10-20	0
Station-2942, 2009-07-08	13
Station-2942, 2009-07-09	14
Station-2942, 2009-07-10	17
...	...

# Accessing the View

---

[http://localhost:5984/db/\\_design/stations/\\_view/temperature](http://localhost:5984/db/_design/stations/_view/temperature)

Keys	Value
...	...
Station-1873, 2009-10-19	2
Station-1873, 2009-10-20	0
Station-2942, 2009-07-08	13
Station-2942, 2009-07-09	14
Station-2942, 2009-07-10	17
...	...

# Accessing the View

---

*“Get the temperatures from 8–9 July 2009 of Station-2942”*

Keys	Value
...	...
Station-1873, 2009-10-19	2
Station-1873, 2009-10-20	0
Station-2942, 2009-07-08	13
Station-2942, 2009-07-09	14
Station-2942, 2009-07-10	17
...	...

# Accessing the View

---

*“Get the temperatures from 8–9 July 2009 of Station-2942”*

`http://localhost:5984/db/_design/stations/_view/temperature?`

`startkey=["Station-2942", "2009-07-08"]&`

`endkey=["Station-2942", "2009-07-09"]`

Keys	Value
...	...
Station-1873, 2009-10-19	2
Station-1873, 2009-10-20	0
<b>Station-2942, 2009-07-08</b>	<b>13</b>
<b>Station-2942, 2009-07-09</b>	<b>14</b>
Station-2942, 2009-07-10	17
...	...

# Reduce

# Reduce

---

```
function(keys, values) {  
  var sum = 0;  
  for(var idx in values) {  
    sum = sum + values[idx];  
  }  
  return sum;  
}
```

# Reduce

---

```
function(keys, values) {  
  var sum = 0;  
  for(var idx in values) {  
    sum = sum + values[idx];  
  }  
  return sum;  
}
```



# Reduce result

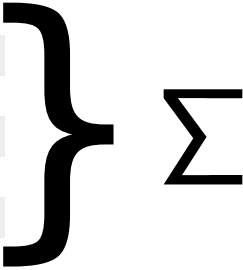
---

Keys	Value
...	...
Station-1873, 2009-10-19	2
Station-1873, 2009-10-20	0
Station-2942, 2009-07-08	13
Station-2942, 2009-07-09	14
Station-2942, 2009-07-10	17
...	...

# Reduce result

---

Keys	Value
...	...
Station-1873, 2009-10-19	2
Station-1873, 2009-10-20	0
Station-2942, 2009-07-08	13
Station-2942, 2009-07-09	14
Station-2942, 2009-07-10	17
...	...



# Reduce result

---

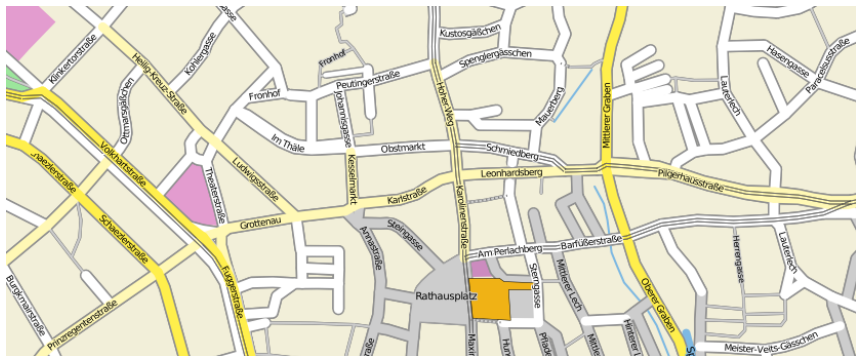
Keys	Value
...	...
Station-1873, 2009-10-19	2
Station-1873, 2009-10-20	0
<b>Station-2942, 2009-07-08</b>	<b>13</b>
<b>Station-2942, 2009-07-09</b>	<b>14</b>
Station-2942, 2009-07-10	17
...	...

} 27

# Web Mapping Applications

# Serving up data

- Base map (e.g. Google, OpenStreetMap)



Source: <http://www.openstreetbrowser.org/>

# Serving up data

- Base map (e.g. Google, OpenStreetMap)
- Some overlay (your data)



Source: <http://www.openstreetbrowser.org/>

# Typical 3-Tier architecture

---



## Client

JavaScript  
(e.g. OpenLayers)

---



## Server

Web Map/Feature Server  
(e.g. GeoServer, MapServer)

---



## Database

Geospatial database  
(e.g. PostGIS, SpatiaLite)

# Typical 3-Tier architecture

---



## Client

JavaScript  
(e.g. OpenLayers)

---



## Server

Web Map/Feature Server  
(e.g. GeoServer, MapServer)

---



## Database

Geospatial database  
(e.g. PostGIS, SpatiaLite)

- Problem: Much effort needed for a high-performance/scalable database



## 2-Tier architecture with CouchDB

---



### Client

JavaScript  
(e.g. OpenLayers,  
code to access CouchDB)

---



### CouchDB

# GeoCouch

# GeoCouch

---

- Spatial extension for CouchDB

# GeoCouch

---

- Spatial extension for CouchDB
- Currently SpatiaLite backend (PostGIS is easily possible)

# GeoCouch

---

- Spatial extension for CouchDB
- Currently SpatiaLite backend (PostGIS is easily possible)
- Seamless data storage integration with CouchDB

# GeoCouch Features

---

- Currently supported geometries (OGC):
  - ▶ (Multi-)Points
  - ▶ (Multi-)LineStrings
  - ▶ (Multi-)Polygons

# GeoCouch Features

---

- Currently supported geometries (OGC):
  - ▶ (Multi-)Points
  - ▶ (Multi-)LineStrings
  - ▶ (Multi-)Polygons
  
- Currently supported queries (OpenSearch Geo):
  - ▶ Bounding box search
  - ▶ Polygon search
  - ▶ Radius search

# Spatially enabled CouchDB

---



## Client

JavaScript  
(e.g. OpenLayers,  
code to access CouchDB)

---



CouchDB



GeoCouch

(with spatial index)



# Replication

# Replication

---

- CouchDB supports n-master replication

# Replication

---

- CouchDB supports n-master replication
- Easy to work “offline”, synchronize later

# Replication

---

- CouchDB supports n-master replication
- Easy to work “offline”, synchronize later
- CouchDB is integral part of Ubuntu Karmic for synchronising contacts/bookmarks.

# Robust Storage

# Robust Storage

---

- Always in a consistent state

# Robust Storage

---

- Always in a consistent state
- No shutdown command

# Robust Storage

---

- Always in a consistent state
- No shutdown command
- Append-only B-Tree



# Robust Storage

---

- Always in a consistent state
- No shutdown command
- Append-only B-Tree
- Compaction

# CouchDB as massive storage for geodata

---

- Attachments

# CouchDB as massive storage for geodata

---

- Attachments
  - ▶ As tilecache

# CouchDB as massive storage for geodata

---

- Attachments
  - ▶ As tilecache
  - ▶ As metadata storage/server

# Conclusion

# Conclusion

---

- For developers
  - ▶ Robust, highly concurrent
  - ▶ Eases development
  - ▶ Easy replication

# Conclusion

---

- For developers
  - ▶ Robust, highly concurrent
  - ▶ Eases development
  - ▶ Easy replication
- Won't vanish:
  - ▶ Used in production systems (e.g. BBC)
  - ▶ Growing user base (as part of Ubuntu Karmic)

# Conclusion

---

- For developers
  - ▶ Robust, highly concurrent
  - ▶ Eases development
  - ▶ Easy replication
- Won't vanish:
  - ▶ Used in production systems (e.g. BBC)
  - ▶ Growing user base (as part of Ubuntu Karmic)

**CouchDB as next-generation geodata storage.**



# Thanks!

# Get in touch with me

---

- Webiste: <http://vmx.cx/>
- IRC: vmx @ freenode
- Email: [volker.mische@gmail.com](mailto:volker.mische@gmail.com)
- Jabber: [volker@vmx.cx](xmpp:volker@vmx.cx)