

Learning CouchDB

A Non-Relational Alternative to Data Persistence
for Modern Software Applications

**Is CouchDB a good choice
for your application?**

Problem:

The Object-Relational Impedance Mismatch^[1]

How to persist data in an object-oriented software application?

1. <http://c2.com/cgi/wiki/ObjectRelationalImpedanceMismatch>
2. <http://martinfowler.com/eaCatalog/repository.html>
3. <http://c2.com/cgi/wiki/ObjectRelationalMapping>

Problem:

The Object-Relational Impedance Mismatch^[1]

How to persist data in an object-oriented software application?

separate domain
and data mapping
layers^[2]

object-relational
mapping (ORM)^[3]

non-relational
database

1. <http://c2.com/cgi/wiki/ObjectRelationalImpedanceMismatch>

2. <http://martinfowler.com/eaCatalog/repository.html>

3. <http://c2.com/cgi/wiki/ObjectRelationalMapping>

Problem:

Semi-Structured Data

How to persist data with flexible schemas?

Problem:

Semi-Structured Data

How to persist data with flexible schemas?

**leave
non-applicable
column values null**

**use the
entity-attribute-
value (EAV)
anti-pattern^[1]**

**use a
schema-less
database with a
self-describing
structure**

1. <http://pragprog.com/book/bksqla/sql-antipatterns>

Problem:

Persisting Graph Relationships

How to persist graphs, trees and hierarchical data?

Problem:

Persisting Graph Relationships

How to persist graphs, trees and hierarchical data?

model hierarchical data in SQL^[1]

graph database^[2]

document-oriented database

1. <http://www.percona.com/webinars/2011-02-28-models-for-hierarchical-data-in-sql-and-php/>

2. <http://www.graph-database.org/>

Problem:

Achieving High Concurrency

How to give up consistency in exchange for high availability? ^[1]

1. <http://www.julianbrowne.com/article/viewer/brewers-cap-theorem>

Problem:

Achieving High Concurrency

How to give up consistency in exchange for high availability? ^[1]

reduce
transaction
scope

database
replication

denormalization

multi-version
concurrency
control (MVCC)

1. <http://www.julianbrowne.com/article/viewer/brewers-cap-theorem>

Problem:

Building in Fault-Tolerance

How to make a fault-tolerant system?

Problem:

Building in Fault-Tolerance

How to make a fault-tolerant system?

**promote
slave database
to master on fault**

isolate faults

**multi-master
replication**

Problem:

Flexible Indexing

How to index data other than direct column values?

Problem:

Flexible Indexing

How to index data other than direct column values?

denormalize

user-defined types

Map/Reduce

Lesson

About CouchDB

Schema-Less

- **stores self-contained JSON documents**
- **related entities can be stored in a single document**
- **only store fields that are needed in each document**

Futon Web Administration

- built-in web administration console
- default location is: http://localhost:5984/_utils/
- create, read, update and delete databases and documents
- query databases
- configure CouchDB
- replicate between databases
- view task status
- run test suite
- set up server admins
- configure database security
- run compaction and cleanup maintenance tasks

CouchDB Views

- queries are run against indexed *views*
- views are generated through incremental *Map* functions
- aggregate results can be retrieved through *Reduce* functions

HTTP API

- every language and platform has an HTTP client
- uses existing semantics (e.g. 201 Created, 202 Accepted)
- distributed, scalable and cacheable

CouchDB and PHP

- can just use an HTTP client
- several client libraries specific to CouchDB are available

Flexible Querying Options

- all rows in a given view
- row(s) matching a specified *key*
- rows by start and end keys
- exact grouping
- group by *levels*
- *limit* and *skip* parameters
- output in descending order
- *include* original documents in result set

Lesson

Installation

CouchDB or Couchbase

CouchDB:

- project of the Apache Software Foundation
- available through package managers

Couchbase:

- superset of CouchDB
- includes geospatial indexing
- available as Couchbase Mobile for iOS and Android
- commercial support options available

CouchDB Installation

- Mac OS X
 - Homebrew
 - MacPorts
- Windows
 - [binary installer](http://wiki.apache.org/couchdb/Windows_binary_installer)^[1]
- Ubuntu
 - Aptitude
- Red Hat
 - Yum

1. http://wiki.apache.org/couchdb/Windows_binary_installer

Couchbase Single Server^[1]

Available for:

- Mac OS X
- Windows
- Ubuntu
- Red Hat

1. <http://www.couchbase.com/products-and-services/couchbase-single-server>

Lab

Install CouchDB or Couchbase

Installation and Startup



1. Install CouchDB

OR

Install Couchbase Single Server

2. If CouchDB, then at the command-line:

```
$ sudo couchdb
```

Apache CouchDB has started. Time to relax.

3. Optionally, test CouchDB:

```
$ curl 'http://localhost:5984/'
```

```
{"couchdb": "Welcome", "version": "1.1.0"}
```

Lesson

JSON Documents

Why JSON?

- **human-readable and simple data interchange format**
- **data structures from many programming languages can be easily converted to and from JSON**
- **lightweight—doesn't add too much to bandwidth overhead**

JSON Data Types

- **String**
- **Number**
- **Boolean** (**false** or **true**)
- **JSON Array** (e.g. ["a", "b", "c"])
- **JSON Object**: collection of name/value pairs where the name is a **String** and the value is any valid JSON data type
- **JSON NULL** (**null**)

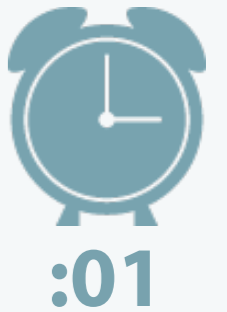
Example JSON Object

```
{  
  "_id": "978-0-596-15589-6",  
  "title": "CouchDB: The Definitive Guide",  
  "subtitle": "Time to Relax",  
  "authors": [  
    "J. Chris Anderson",  
    "Jan Lehnardt",  
    "Noah Slater"  
  ],  
  "publisher": "O'Reilly Media",  
  "released": "2010-01-19",  
  "pages": 272  
}
```

Lab

Futon

Access Futon



1. Visit Futon in your web browser:
http://localhost:5984/_utils/
2. Note the existing *_replicator* and *_users* databases
3. Note the *Tools* navigation section
4. Note the following message:
“Welcome to Admin Party!”


Access Futon

Overview

+ Create Database ...

Name	Size	Number of Documents	Update Seq
_replicator	8.1 KB	1	1
_users	4.1 KB	1	1

Showing 1-2 of 2 databases ← Previous Page | Rows per page: 10 | Next Page →



Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Create a Database



:01

1. Click *Create Database ...*
2. Enter “books” in the *Database Name* field
3. Click the *Create* button

Create a Database

Overview

+ Create Database ...

Name	Size	Number of Documents	Update Seq
_replicator	8.1 KB	1	1
_users	4.1 KB	1	1

Showing 1-2 of 2 databases

← Previous Page | Rows per page: 10 | Next Page →

Create New Database

Please enter the name of the database. Note that only lowercase characters (a-z), digits (0-9), or any of the characters `_`, `$`, `(`, `)`, `+`, `-`, and `/` are allowed.

Database Name:

[Create](#) [Cancel](#)

CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Create a Document



1. Click *New Document*
2. Enter "978-0-596-15589-6" as the value for the *_id* field, and then click apply ✓
3. Using the *Add Field* button, add a "title" field with a value of "CouchDB: The Definitive Guide"
4. Add a "subtitle" field with a value of "Time to Relax"
5. Add an "authors" field with a value of:
["J. Chris Anderson", "Jan Lehnardt", "Noah Slater"]
6. Add a "publisher" field with a value of "O'Reilly Media"
7. Add a "released" field with a value of "2010-01-19"
8. Add a "pages" field with a value of "272"
9. Click *Save Document*

Create a Document


Overview > books > 978-0-596-15589-6

Save Document Add Field Upload Attachment...

Fields Source

Field	Value
_id	"978-0-596-15589-6"
✕ title	"CouchDB: The Definitive Guide"
✕ subtitle	"Time to Relax"
✕ authors	0 "J. Chris Anderson" 1 "Jan Lehnardt" 2 "Noah Slater"
✕ publisher	"O'Reilly Media"
✕ released	"2010-01-19"
✕ pages	272

← Previous Version | Next Version →



Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Lesson

Map Functions

Map Functions

- used to transform documents into key/value pairs
- user-defined, typically written in JavaScript
- every document is incrementally passed through this function
- function is passed a JSON object representing a document
- function calls an *emit* function zero, one or more times
- *emit* function accepts two arguments:
 - a *key*
 - a *value* to be associated with the *key*
- an *id* field is implicitly emitted as well

Temporary Views

- contains a *Map* function and an optional *Reduce* function
- useful in development
- very slow on large data sets

Lab

One-To-One Mapping

Map Book Titles



1. Click on the *books* database in the breadcrumb navigation
2. Select *Temporary view...* from the *View* select menu
3. Enter the following in the *View Code* text area:

```
function(doc) {  
  if (doc.title) {  
    emit(doc.title);  
  }  
}
```

4. Click *Run*

Map Book Titles

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊖ Compact & Cleanup... ✕ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.title) { emit(doc.title); } }</pre>	

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Value
"CouchDB: The Definitive Guide" ID: 978-0-596-15589-6	null

Showing 1-1 of 1 row ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

**Map functions must be deterministic.
Given the same input, a Map function
must always return the same output.**

Add a Second Document



1. Click *New Document*
2. Enter "978-0-596-52926-0" as the value for the *_id* field, and then click apply ✓
3. Add a "title" field with a value of "RESTful Web Services"
4. Add a "subtitle" field with a value of "Web services for the real world"
5. Add an "authors" field with a value of:
["Leonard Richardson", "Sam Ruby"]
6. Add a "publisher" field with a value of "O'Reilly Media"
7. Add a "released" field with a value of "2007-05-08"
8. Add a "pages" field with a value of "448"
9. Click *Save Document*

Add a Second Document


Overview > books > 978-0-596-52926-0

Save Document Add Field Upload Attachment...

Fields Source

Field	Value
<input type="checkbox"/> _id	"978-0-596-52926-0"
<input type="checkbox"/> title	"RESTful Web Services"
<input type="checkbox"/> subtitle	"Web services for the real world"
<input type="checkbox"/> authors	0 "Leonard Richardson" 1 "Sam Ruby"
<input type="checkbox"/> publisher	"O'Reilly Media"
<input type="checkbox"/> released	"2007-05-08"
<input type="checkbox"/> pages	448

← Previous Version | Next Version →



Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

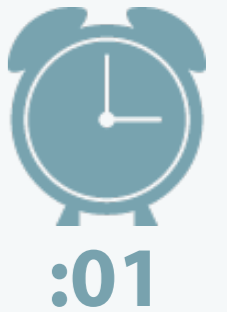
Recent Databases

books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Map Book Titles



1. Click on the *books* database in the breadcrumb navigation
2. If not already selected, select *Temporary view...* from the *View* select menu
3. If not already entered, enter the following in the *View Code* text area:

```
function(doc) {  
  if (doc.title) {  
    emit(doc.title);  
  }  
}
```

4. Click *Run*

Map Book Titles

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊖ Compact & Cleanup... ✕ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.title) { emit(doc.title); } }</pre>	

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Value
"CouchDB: The Definitive Guide" ID: 978-0-596-15589-6	null
"RESTful Web Services" ID: 978-0-596-52926-0	null

Showing 1-2 of 2 rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Lab

One-To-Many Mapping

Add a *formats* Field



1. Navigate to the *books* database, if not already there
2. Select *All documents* from the *View* select menu
3. Click the first document, 978-0-596-15589-6
4. Add a “formats” field with a value of:
["Print", "Ebook", "Safari Books Online"]
5. Click *Save Document*
6. Navigate back to the *books* database
7. Repeat steps 3 through 6 for the second document, 978-0-596-52926-0

✕ **formats**

0 "Print"

1 "Ebook"

2 "Safari Books Online"

When updating documents, you may have noticed the `_rev` field, an artifact of CouchDB's *Multi-Version Concurrency Control* (MVCC).

Add a Third Document



1. Click *New Document*
2. Enter "978-1-565-92580-9" as the value for the `_id` field, and then click apply ✓
3. Add a "title" field with a value of "DocBook: The Definitive Guide"
4. Add an "authors" field with a value of:
["Norman Walsh", "Leonard Muellner"]
5. Add a "publisher" field with a value of "O'Reilly Media"
6. Add a "formats" field with a value of:
["Print"]
7. Add a "released" field with a value of "1999-10-28"
8. Add a "pages" field with a value of "648"
9. Click *Save Document*

Add a Third Document


Overview > books > 978-1-565-92580-9

Save Document Add Field Upload Attachment...

Fields Source

Field	Value
_id	"978-1-565-92580-9"
title	"DocBook: The Definitive Guide"
authors	0 "Norman Walsh" 1 "Leonard Muellner"
publisher	"O'Reilly Media"
formats	0 "Print"
released	"1999-10-28"
pages	648

← Previous Version | Next Version →



Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Map Book Formats



1. Navigate to the *books* database
2. Select *Temporary view...* from the *View* select menu
3. Enter the following in the *View Code* text area:

```
function(doc) {  
  if (doc.formats) {  
    for (var i in doc.formats) {  
      emit(doc.formats[i]);  
    }  
  }  
}
```

4. Click *Run*

Map Book Formats

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊗ Compact & Cleanup... ⊗ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.formats) { for (var i in doc.formats) { emit(doc.formats[i]); } } }</pre>	

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Value
"Ebook" ID: 978-0-596-15589-6	null
"Ebook" ID: 978-0-596-52926-0	null
"Print" ID: 978-0-596-15589-6	null
"Print" ID: 978-0-596-52926-0	null
"Print" ID: 978-1-565-92580-9	null
"Safari Books Online" ID: 978-0-596-15589-6	null
"Safari Books Online" ID: 978-0-596-52926-0	null

Showing 1-7 of 7 rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Map Book Authors



1. Navigate to the *books* database
2. Select *Temporary view...* from the *View* select menu
3. Enter the following in the *View Code* text area:

```
function(doc) {  
  if (doc.authors) {  
    for (var i in doc.authors) {  
      emit(doc.authors[i]);  
    }  
  }  
}
```

4. Click *Run*

Map Book Authors

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊗ Compact & Cleanup... ⊗ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.authors) { for (var i in doc.authors) { emit(doc.authors[i]); } } }</pre>	

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Value
"J. Chris Anderson" ID: 978-0-596-15589-6	null
"Jan Lehnardt" ID: 978-0-596-15589-6	null
"Leonard Mueller" ID: 978-1-565-92580-9	null
"Leonard Richardson" ID: 978-0-596-52926-0	null
"Noah Slater" ID: 978-0-596-15589-6	null
"Norman Walsh" ID: 978-1-565-92580-9	null
"Sam Ruby" ID: 978-0-596-52926-0	null

Showing 1-7 of 7 rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Lesson

Reduce Functions

Reduce Functions

- optional, run against data set produced by a Map function
- typically used to *reduce* a set of values to a single, scalar value
- can reference built-in Reduce functions
- custom Reduce functions can be written in JavaScript
- set of already reduced data may be *rereduced*
- function is passed three arguments:
 - *keys*: array of mapped keys and associated document identifiers in the form of *[key, id]*
 - *values*: array of mapped values
 - *rereduce*: boolean value indicating whether or not the reduce function is being called recursively on its own output (in which case, *keys* will be null)

Built-In Reduce Functions

- written in CouchDB's native Erlang
- faster than user-defined JavaScript functions
- includes:
 - **_count**
 - **_sum**
 - **_stats**

If you think that you need to write your own custom Reduce function, you're probably doing it wrong.

Grouping

- values can be reduced by group
- grouping is controlled by a query, not by the Reduce function
- grouping is done by *key*
- *key levels*, or parts of a key, can also be grouped on

Lab

Built-In Reduce Functions

The built-in *_count* function can reduce arbitrary values, including null values.

Count: No Grouping



1. Select *Temporary view...* from the *View* select menu
2. Enter the following in the *View Code* text area:

```
function(doc) {  
  if (doc.formats) {  
    for (var i in doc.formats) {  
      emit(doc.formats[i]);  
    }  
  }  
}
```

3. Enter the following in the *Reduce Function* text area:
 `_count`
4. Click *Run*
5. Check the *Reduce* checkbox
6. Select *none* from the *Grouping* select menu

Count: No Grouping

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊖ Compact & Cleanup... ✕ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.formats) { for (var i in doc.formats) { emit(doc.formats[i]); } } }</pre>	<pre>_count</pre>

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Grouping: <input type="text" value="none"/>	Value	<input checked="" type="checkbox"/> Reduce
null		7	

Showing 1-1 of unknown rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

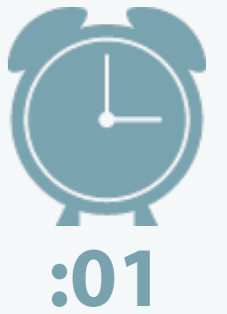
Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Count: Exact Grouping



1. Select *exact* from the *Grouping* select menu

Count: Exact Grouping

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊗ Compact & Cleanup... ⊗ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.formats) { for (var i in doc.formats) { emit(doc.formats[i]); } } }</pre>	<pre>_count</pre>

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Grouping: <input type="text" value="exact"/>	Value	<input checked="" type="checkbox"/> Reduce
"Ebook"		2	
"Print"		3	
"Safari Books Online"		2	

Showing 1-3 of unknown rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

**The *_sum* and *_stats* functions
will only reduce sets of numbers.**

Sum: Exact Grouping



1. Update the *View Code* text area with the following:

```
function(doc) {  
  if (doc.formats) {  
    for (var i in doc.formats) {  
      emit(doc.formats[i], doc.pages);  
    }  
  }  
}
```

2. Enter the following in the *Reduce Function* text area:

```
_sum
```

3. Click *Run*
4. The *Reduce* checkbox should be checked
5. *exact* should be selected from the *Grouping* select menu

Sum: Exact Grouping

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊖ Compact & Cleanup... ✕ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.formats) { for (var i in doc.formats) { emit(doc.formats[i], doc.pages); } } }</pre>	<pre>_sum</pre>

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Grouping: <input type="text" value="exact"/>	Value	<input checked="" type="checkbox"/> Reduce
"Ebook"		720	
"Print"		1368	
"Safari Books Online"		720	

Showing 1-3 of unknown rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

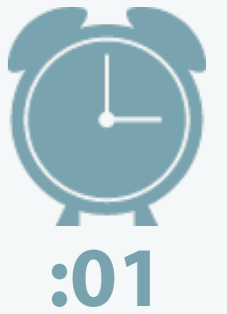
Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Stats: Exact Grouping



1. Enter the following in the *Reduce Function* text area:
_stats
2. Click *Run*
3. The *Reduce* checkbox should be checked
4. *exact* should be selected from the *Grouping* select menu

Stats: Exact Grouping

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊖ Compact & Cleanup... ✕ Delete Database...

▼ View Code


Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.formats) { for (var i in doc.formats) { emit(doc.formats[i], doc.pages); } } }</pre>	<pre>_stats</pre>

Run Language: = Revert Save As... Save

Warning: Please note that temporary views are not suitable for use in production, as they are really slow for any database with more than a few dozen documents. You can use a temporary view to experiment with view functions, but switch to a permanent view before using them in an application.

Key ▲	Grouping: <input type="text" value="exact"/>	Value	<input checked="" type="checkbox"/> Reduce
"Ebook"		{sum: 720, count: 2, min: 272, max: 448, sumsqr: 274688}	
"Print"		{sum: 1368, count: 3, min: 272, max: 648, sumsqr: 694592}	
"Safari Books Online"		{sum: 720, count: 2, min: 272, max: 448, sumsqr: 274688}	

Showing 1-3 of unknown rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Map/Reduce gives you tremendous flexibility, but has its limitations:

no ad-hoc queries*

index is one dimensional

only key or range queries are allowed†

*Temporary views are only useful in development, not in production.

†Multiple key queries are allowed, but result in multiple hits on the index.

Lesson

Design Documents

Non-Temporary Views

- multiple *views* can be defined within a *design document*
- a view contains a *Map* and an optional *Reduce* function
- a view is recomputed when its design document is updated
- views are incrementally re-indexed after documents are created, updated or deleted*

*Technically, views are incrementally re-indexed when queried.

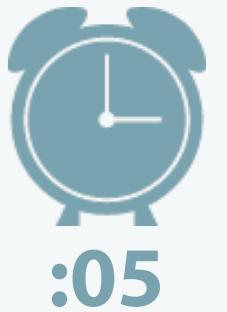
Contents of a Design Document

- **view definitions**
 - **Map functions**
 - **Reduce functions**
- **show functions**
- **list functions**
- **document update handlers**
- **document update validation functions**
- **rewrite definitions**
- **arbitrary fields**
- **arbitrary attachments**

Lab

Design Documents

Titles View



1. Update the *View Code* text area with the following:

```
function(doc) {  
  if (doc.title) {  
    emit(doc.title, doc.pages);  
  }  
}
```

2. Enter the following in the *Reduce Function* text area:

```
_stats
```

3. Click *Save As...*
4. Enter "default" as the *Design Document* name
5. Enter "titles" as the *View Name*
6. Click *Save*

Titles View

Overview > books

New Document Security... Jump to: Document ID View: Temporary view... Stale views

Compact & Cleanup... Delete Database...

View Code

Map Function:

```
function(doc) {  
  if (doc.title) {  
    emit(doc.title, doc.pages);  
  }  
}
```

Reduce Function (optional):

```
_stats
```

Run Language: javascript

Warning: Please note that temporary views are not saved to the database. You can use a temporary view for development, but not for production.

Key null

Showing 1-1 of unknown rows

Save View As...

You can save this function code as a permanent view in the database. Just enter or select the design document and the name of the view below. Note that if you choose an existing view, it will be overwritten!

Design Document:

View Name:

Save Cancel

CouchDB relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

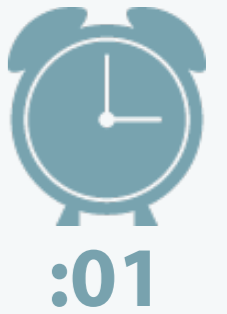
Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Query Titles View



1. Select *titles* from the *View* select menu, if not already selected
2. Click arrow next to *View Code*, if you'd like
3. Check the *Reduce* checkbox, if not already checked
4. Select *none* from the *Grouping* select menu

Query Titles View

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊗ Compact & Cleanup... ⊗ Delete Database...


▼ View Code _design/default

Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.title) { emit(doc.title, doc.pages); } }</pre>	<pre>_stats</pre>

Run Language: = Revert Save As... Save

Key ▲	Grouping: <input type="text" value="none"/>	Value	<input checked="" type="checkbox"/> Reduce
null		{sum: 1368, count: 3, min: 272, max: 648, sumsqr: 694592}	

Showing 1-1 of unknown rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Formats View



1. Update the *View Code* text area with the following:

```
function(doc) {  
  if (doc.formats) {  
    for (var i in doc.formats) {  
      emit(doc.formats[i], doc.pages);  
    }  
  }  
}
```

2. Enter the following in the *Reduce Function* text area:

`_stats`

3. Click *Save As...*
4. Enter "default" as the *Design Document* name
5. Enter "formats" as the *View Name*
6. Click *Save*

Formats View

The screenshot shows the CouchDB Admin interface for a database named 'books'. The 'View Code' section is active, displaying a JavaScript map function:

```
function(doc) {  
  if (doc.formats) {  
    for (var i in doc.formats) {  
      emit(doc.formats[i], doc.pages);  
    }  
  }  
}
```

The 'Reduce Function (optional):' section is empty. Below the code editor, there is a 'Run' button and a language dropdown set to 'javascript'. A warning message states: "Warning: Please note that temporary views are not suitable for production use with more than a few dozen documents. You can use a temporary view to experiment with new functions, but switch to a permanent view before using them in an application."

A 'Save View As...' dialog box is overlaid on the interface. It contains the following text:

You can save this function code as a permanent view in the database. Just enter or select the design document and the name of the view below. Note that if you choose an existing view, it will be overwritten!

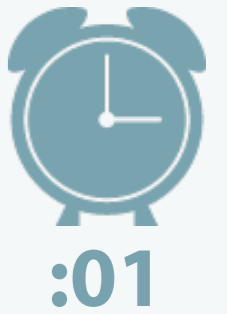
Design Document:

View Name:

Buttons: Save, Cancel

The right sidebar shows the CouchDB logo with the tagline 'relax', a 'Tools' menu with options: Overview, Configuration, Replicator, Status, Test Suite, and a 'Recent Databases' section listing 'books'. At the bottom, there is a footer: 'Welcome to Admin Party! Everyone is admin. Fix this' and 'Futon on Apache CouchDB 1.1.0'.

Query Formats View



1. Select *formats* from the *View* select menu, if not already selected
2. Click the arrow next to *View Code*, if you'd like
3. Check the *Reduce* checkbox, if not already checked
4. Select *none* from the *Grouping* select menu

Query Formats View

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊖ Compact & Cleanup... ✕ Delete Database...


▼ View Code _design/default

Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.formats) { for (var i in doc.formats) { emit(doc.formats[i], doc.pages); } } }</pre>	<pre>_stats</pre>

Run Language: = Revert Save As... Save

Key ▲	Grouping: <input type="text" value="none"/>	Value	<input checked="" type="checkbox"/> Reduce
null		{sum: 2808, count: 7, min: 272, max: 648, sumsqr: 1243968}	

Showing 1-1 of unknown rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Authors View



1. Update the *View Code* text area with the following:

```
function(doc) {  
  if (doc.authors) {  
    for (var i in doc.authors) {  
      emit(doc.authors[i], doc.pages);  
    }  
  }  
}
```

2. Enter the following in the *Reduce Function* text area:

`_stats`

3. Click *Save As...*
4. Enter "default" as the *Design Document* name
5. Enter "authors" as the *View Name*
6. Click *Save*

Authors View

The screenshot shows the Apache CouchDB Futon interface for a database named 'books'. The main area displays a JavaScript Map Function for a view:

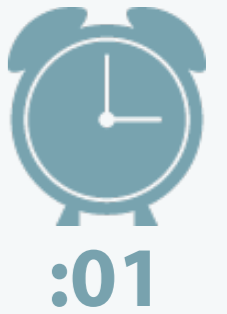
```
function(doc) {
  if (doc.authors) {
    for (var i in doc.authors) {
      emit(doc.authors[i], doc.pages);
    }
  }
}
```

The 'Reduce Function (optional):' field is empty, and the '_stats' checkbox is checked. Below the code editor, there is a 'Run' button and a 'Language' dropdown set to 'javascript'. A warning message states: "Warning: Please note that temporary views are not intended for use in production. They are only for experimentation with view functions, but switch to a permanent view before using them in an application."

A modal dialog titled "Save View As..." is open in the center. It contains the following text: "You can save this function code as a permanent view in the database. Just enter or select the design document and the name of the view below. Note that if you choose an existing view, it will be overwritten!". The dialog has two input fields: "Design Document:" with the value "_design/default" and "View Name:" with the value "authors". At the bottom of the dialog are "Save" and "Cancel" buttons.

The right sidebar shows the CouchDB logo with the tagline "relax", a "Tools" menu with options like Overview, Configuration, Replicator, Status, and Test Suite, and a "Recent Databases" section listing "books". At the bottom of the sidebar, there is a message: "Welcome to Admin Party! Everyone is admin. Fix this" and the version information "Futon on Apache CouchDB 1.1.0".

Query Authors View



1. Select *authors* from the *View* select menu, if not already selected
2. Click arrow next to *View Code*, if you'd like
3. Check the *Reduce* checkbox, if not already checked
4. Select *none* from the *Grouping* select menu

Query Authors View

Overview > books

+ New Document ? Security... Jump to: View: Stale views

⊖ Compact & Cleanup... ✕ Delete Database...


▼ View Code _design/default

Map Function:	Reduce Function (optional):
<pre>function(doc) { if (doc.authors) { for (var i in doc.authors) { emit(doc.authors[i], doc.pages); } } }</pre>	<pre>_stats</pre>

Run Language: = Revert Save As... Save

Key ▲	Grouping: <input type="text" value="none"/>	Value	<input checked="" type="checkbox"/> Reduce
null		{sum: 3008, count: 7, min: 272, max: 648, sumsqr: 1463168}	

Showing 1-1 of unknown rows ← Previous Page | Rows per page: | Next Page →



CouchDB
relax

Tools

- Overview
- Configuration
- Replicator
- Status
- Test Suite

Recent Databases

- books

Welcome to Admin Party!
Everyone is admin. [Fix this](#)

Futon on Apache CouchDB 1.1.0

Views use extra disk space in exchange for additional performance. Consider the design of your views accordingly.

Lesson

HTTP API

Resources

- **URI identifies a resource**

- **resource could be a:**

- **database**

`http://localhost:5984/books`

- **document**

`http://localhost:5984/books/978-0-596-15589-6`

- **attachment**

`http://localhost:5984/books/978-0-596-15589-6/cover.gif`

- **configuration section**

`http://localhost:5984/_config/uuids`

Methods

- HTTP methods define actions on resources
- HTTP methods used in CouchDB include:
 - GET
 - POST
 - PUT
 - DELETE

Self-Describing Messages

- HTTP request headers:
 - **Accept**
 - **Content-Type**
- HTTP response status codes:
 - **200 OK**
 - **201 Created**
 - **409 Conflict**
- HTTP response headers:
 - **Content-Type**
 - **Etag**

Conditional caching can be done with the **Etag** HTTP response header and the **If-None-Match** HTTP request header.

Hypermedia Controls

- not part of the core API*
- can be created with a combination of:
 - show functions
 - list functions
 - document update handlers

*One might argue that a document identifier could serve as a form of hypermedia control.

cURL^[1]

- tool for transferring data
- understands many protocols, including HTTP
- both a library and a command-line tool

Useful cURL Switches

- **-d** HTTP POST data
 - **--data-urlencode** HTTP POST data url encoded
- **-G** Send the -d data with a HTTP GET
- **-H** Custom header to pass to server
- **-i** Include protocol headers in the output
- **-X** Specify request command to use

Lab

cURL and CouchDB

Install cURL

1. Install cURL command-line tool using your package manager
OR

Download and install cURL executable^[1]

2. Test cURL and CouchDB:

```
$ curl -iX GET 'http://localhost:5984/'
```

```
HTTP/1.1 200 OK
```

```
Server: CouchDB/1.1.0 (Erlang OTP/R14B03)
```

```
Date: Tue, 11 Oct 2011 16:21:40 GMT
```

```
Content-Type: text/plain; charset=utf-8
```

```
Content-Length: 40
```

```
Cache-Control: must-revalidate
```

```
{"couchdb": "Welcome", "version": "1.1.0"}
```

Accept JSON

1. Tell CouchDB that you can accept JSON:

```
$ curl -iX GET 'http://localhost:5984/'  
-H 'Accept: application/json'
```

```
HTTP/1.1 200 OK  
Server: CouchDB/1.1.0 (Erlang OTP/R14B03)  
Date: Tue, 11 Oct 2011 16:51:40 GMT  
Content-Type: application/json  
Content-Length: 40  
Cache-Control: must-revalidate
```

```
{"couchdb": "Welcome", "version": "1.1.0"}
```

Create a Database

1. PUT a new database into CouchDB:

```
$ curl -iX PUT 'http://localhost:5984/mydb'
```

```
HTTP/1.1 201 Created
```

```
Location: http://localhost:5984/mydb
```

```
{"ok":true}
```


Create a Document

1. POST a new document into a CouchDB database:

```
$ curl -iX POST 'http://localhost:5984/mydb'  
-H 'Content-Type: application/json'  
-d '{"_id": "mydoc"}'
```

```
HTTP/1.1 201 Created  
Location: http://localhost:5984/mydb/mydoc
```

```
{  
  "ok": true,  
  "id": "mydoc",  
  "rev": "1-967a00dff5e02add41819138abb3284d"  
}
```

Read the Document

1. GET a document:

```
$ curl -iX GET 'http://localhost:5984/mydb/mydoc'
```

```
HTTP/1.1 200 OK
```

```
Etag: "1-967a00dff5e02add41819138abb3284d"
```

```
{  
  "_id": "mydoc",  
  "_rev": "1-967a00dff5e02add41819138abb3284d"  
}
```

Update the Document

1. PUT an updated document:

```
$ curl -iX PUT 'http://localhost:5984/mydb/mydoc'  
-H 'If-Match: "1-967a00dff5e02add41819138abb3284d"'  
-H 'Content-Type: application/json'  
-d '{"title": "Learning CouchDB"}'
```

```
HTTP/1.1 201 Created  
Location: http://localhost:5984/mydb/mydoc  
Etag: "2-516027e3179a22a22e06874c374e8ef0"
```

```
{  
  "ok": true,  
  "id": "mydoc",  
  "rev": "2-516027e3179a22a22e06874c374e8ef0"  
}
```

Read the Document

1. GET the document again:

```
$ curl -iX GET 'http://localhost:5984/mydb/mydoc'
```

```
HTTP/1.1 200 OK
```

```
Etag: "2-516027e3179a22a22e06874c374e8ef0"
```

```
{
```

```
  "_id": "mydoc",
```

```
  "_rev": "2-516027e3179a22a22e06874c374e8ef0",
```

```
  "title": "Learning CouchDB"
```

```
}
```

Delete the Document

1. DELETE the document:

```
$ curl -iX DELETE 'http://localhost:5984/mydb/mydoc'  
-H 'If-Match: "2-516027e3179a22a22e06874c374e8ef0"'
```

```
HTTP/1.1 200 OK  
Etag: "3-e9a5aa1c486eee23c84fa028bc904991"
```

```
{  
  "ok": true,  
  "id": "mydoc",  
  "rev": "3-e9a5aa1c486eee23c84fa028bc904991"  
}
```

Delete the Database

1. DELETE the database:

```
$ curl -iX DELETE 'http://localhost:5984/mydb'
```

```
HTTP/1.1 200 OK
```

```
{"ok":true}
```

Lesson

CouchDB Libraries for PHP

HTTP Clients

- **Zend_Http_Client^[1]**
 - a Zend Framework component
 - relatively easy-to-use API
- **PHP's cURL extension^[2]**
 - commonly installed extension
 - less intuitive API
- **HTTP_Request2^[3]**
 - a PEAR component

1. <http://framework.zend.com/manual/en/zend.http.html>

2. <http://php.net/manual/en/book.curl.php>

3. http://pear.php.net/package/HTTP_Request2

Sag for CouchDB^[1]

- accepts PHP data structures (e.g. objects, strings)
- can return raw JSON
- can return a PHP object

PHP Object_Freezer^[1]

- created and maintained by Sebastian Bergmann
- “freezes” and “thaws” PHP objects
- extensible storage options
- includes CouchDB object storage
- installable through pear:

```
$ pear channel-discover pear.phpunit.de  
$ pear install phpunit/Object_Freezer
```

PHP CouchDB Extension^[1]

- supports a large part of the CouchDB API
- supports both cookie and basic authentication
- automatically encodes and decodes JSON to and from PHP objects and arrays
- available as a PECL extension

Doctrine2 CouchDB ODM^[1]

- a mapper between PHP objects and CouchDB documents
- uses metadata mapping pattern to map documents to plain old PHP objects
- potentially good fit if you're already using Doctrine

Others

- **PHP On Couch**^[1]
- **Chill**^[2]
- **Settee**^[3]
- **PHPillow**^[4]*

1. <https://github.com/dready92/PHP-on-Couch>

2. <https://github.com/dancryer/Chill>

3. <https://github.com/inadarei/settee>

4. <http://arbitracker.org/phpillow.html>

*PHPillow does not appear to be maintained.

Lesson

Querying Views

To Reduce or Not To Reduce

- results will not be reduced if no Reduce function is defined
- by default, results will be reduced
- to not reduce, set the *reduce* parameter to *false*

Key Matching

- by default, all rows will be returned
- results can be filtered with a *key* parameter
- the *key* parameter can be set to any valid JSON value
- valid JSON values include a:
 - **String**
 - **Number**
 - **Boolean**
 - **JSON Array**
 - **JSON Object**
- must be URL encoded in a query string
- zero, one or more rows matching the key will be returned

Start and End Key Range

- *startkey* parameter can define the beginning of a range
- *endkey* parameter can define the end of a range
- *startkey* and *endkey* parameters can be any valid JSON value
- *startkey* and *endkey* parameters must be URL encoded
- *startkey_docid* parameter further refines the start of a range
- *endkey_docid* parameter further refines the end of a range
- since they reference document identifiers, the *startkey_docid* and *endkey_docid* parameters must be strings

Limiting, Skipping and Reversing

- *limit* parameter specifies a maximum number of rows
- *skip* parameter specifies a number of rows to skip over
- to reverse results, set the *descending* parameter to *true*
- when reversing results, swap your *startkey/startkey_docid* parameters with your *endkey/endkey_docid* parameters

Grouping

- to group results by exact key, set the *group* parameter to *true*
- *group_level* parameter is useful if view's keys are JSON Arrays, otherwise known as *compound keys*
 - a *group_level* of 1 will only group on the first items in the arrays, ignoring any subsequent array elements
 - a *group_level* of 2 will only group on the first and second items in the arrays, ignoring any subsequent array elements
 - a *group_level* of 3 will only group on the first, second and third items in the arrays, ignoring any subsequent array elements
- ...

Including Documents

- original document from which each row was emitted can be included by setting the *include_docs* parameter to *true*
- this will add a *doc* field to your output rows
- if the *value* emitted from the Map function is a JSON Object containing an *_id* field, then CouchDB will instead include the document with an identifier matching the value of this field

Stale Views

- by default, views are incrementally updated when queried
- set the *stale* parameter's value to *ok* to avoid a view update
- if any documents have been created, updated or deleted then this will return results faster
- however, it will return possibly outdated results
- set the *stale* parameter's value to *update_after* to get stale results, but trigger a view update after results are returned

Lab

Querying Views using Zend_Http_Client

Setting up Zend_Http_Client

Install Zend Framework



1. If not already in your include path, install Zend Framework:

```
$ pear channel-discover pear.zfcampus.org  
$ pear install zfcampus/zf
```


Interactive PHP Shell



:01

1. Open an interactive PHP shell:

```
$ php -a
```

Set Up Autoloader



```
require_once 'Zend/Loader/Autoloader.php';  
Zend_Loader_Autoloader::getInstance();
```

All Rows

Reduced



```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/authors');  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

Reduced

key	value
NULL	Array ([sum] => 3008 [count] => 7 [min] => 272 [max] => 648 [sumsqr] => 1463168)

Not Reduced



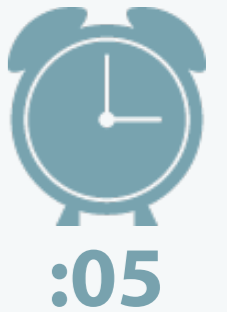
```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/authors');  
$client->setParameterGet('reduce', 'false');  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

Not Reduced

id	key	value
"978-0-596-15589-6"	"J. Chris Anderson"	272
"978-0-596-15589-6"	"Jan Lehnardt"	272
"978-1-565-92580-9"	"Leonard Muellner"	648
"978-0-596-52926-0"	"Leonard Richardson"	448
"978-0-596-15589-6"	"Noah Slater"	272
"978-1-565-92580-9"	"Norman Walsh"	648
"978-0-596-52926-0"	"Sam Ruby"	448

Range Queries

By Key



```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/formats');  
$client->setParameterGet('reduce', 'false');  
$client->setParameterGet('key', '"Print"');  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

By Key

id	key	value
"978-0-596-15589-6"	"Print"	272
"978-0-596-52926-0"	"Print"	448
"978-1-565-92580-9"	"Print"	648

By Start and End Keys



```
$db = 'http://127.0.0.1:5984/books';
$client = new Zend_Http_Client();
$client->setUri($db . '/_design/default/_view/authors');
$client->setParameterGet('reduce', 'false');
$client->setParameterGet('startkey', '"j"');
$client->setParameterGet('endkey', '"j\u0000"');
$client->setHeaders('Accept', 'application/json');
$response = $client->request(Zend_Http_Client::GET);
$body = $response->getBody();
$results = Zend_JSON::decode($body);
print_r($results);
```

By Start and End Keys

id	key	value
"978-0-596-15589-6"	"J. Chris Anderson"	272
"978-0-596-15589-6"	"Jan Lehnardt"	272

Limiting, Skipping and Reversing

Limit Results

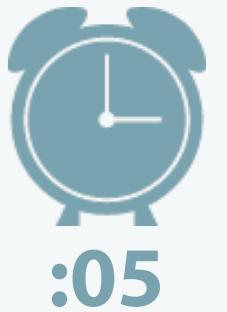


```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/formats');  
$client->setParameterGet('reduce', 'false');  
$client->setParameterGet('limit', 5);  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

Limit Results

id	key	value
"978-0-596-15589-6"	"Ebook"	272
"978-0-596-52926-0"	"Ebook"	448
"978-0-596-15589-6"	"Print"	272
"978-0-596-52926-0"	"Print"	448
"978-1-565-92580-9"	"Print"	648

Skipping Results



```
$db = 'http://127.0.0.1:5984/books';
$client = new Zend_Http_Client();
$client->setUri($db . '/_design/default/_view/formats');
$client->setParameterGet('reduce', 'false');
$client->setParameterGet('limit', 5);
$client->setParameterGet('skip', 5);
$client->setHeaders('Accept', 'application/json');
$response = $client->request(Zend_Http_Client::GET);
$body = $response->getBody();
$results = Zend_Json::decode($body);
print_r($results);
```


Skipping Results

id	key	value
"978-0-596-15589-6"	"Safari Books Online"	272
"978-0-596-52926-0"	"Safari Books Online"	448

Although the *skip* and *limit* parameters can be used together to implement pagination, this can be inefficient on large data sets.

First Page

id	key	value
"978-0-596-15589-6"	"Ebook"	272
"978-0-596-52926-0"	"Ebook"	448
"978-0-596-15589-6"	"Print"	272
"978-0-596-52926-0"	"Print"	448
"978-1-565-92580-9"	"Print"	648

Next Page



```
$db = 'http://127.0.0.1:5984/books';
$client = new Zend_Http_Client();
$client->setUri($db . '/_design/default/_view/formats');
$client->setParameterGet('reduce', 'false');
$client->setParameterGet('limit', 5);
$client->setParameterGet('startkey', '"Print"');
$client->setParameterGet('startkey_docid', '978-1-565-92580-9');
$client->setParameterGet('skip', 1);
$client->setHeaders('Accept', 'application/json');
$response = $client->request(Zend_Http_Client::GET);
$body = $response->getBody();
$results = Zend_Json::decode($body);
print_r($results);
```

Next Page

id	key	value
"978-0-596-15589-6"	"Safari Books Online"	272
"978-0-596-52926-0"	"Safari Books Online"	448

Reversing Results

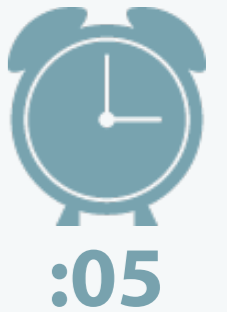


```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/titles');  
$client->setParameterGet('reduce', 'false');  
$client->setParameterGet('descending', 'true');  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

Reversing Results

id	key	value
"978-0-596-52926-0"	"RESTful Web Services"	448
"978-1-565-92580-9"	"DocBook: The Definitive Guide"	648
"978-0-596-15589-6"	"CouchDB: The Definitive Guide"	272

Exact Grouping



```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/formats');  
$client->setParameterGet('group', 'true');  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

Exact Grouping

key	value
"Ebook"	<pre>Array ([sum] => 720 [count] => 2 [min] => 272 [max] => 448 [sumsqr] => 274688)</pre>
"Print"	<pre>Array ([sum] => 1368 [count] => 3 [min] => 272 [max] => 648 [sumsqr] => 694592)</pre>
"Safari Books Online"	<pre>Array ([sum] => 720 [count] => 2 [min] => 272 [max] => 448 [sumsqr] => 274688)</pre>

What We're Doing Next

- adding a fourth book document with a *released* date of *2010-01-08*
- this is the same year and month as another book's *released* date
- creating a new *releases* view that indexes books by their *released* dates
- *releases* view will use a compound key—a three-level JSON Array of *year, month* and *day*

Add a Fourth Document



```
$book = array(  
    '_id'        => '978-0-596-80579-1',  
    'title'      => 'Building iPhone Apps with HTML, CSS, and JavaScript',  
    'subtitle'   => 'Making App Store Apps Without Objective-C or Cocoa',  
    'authors'    => array('Jonathan Stark'),  
    'publisher' => 'O\Reilly Media',  
    'formats'    => array('Print', 'Ebook', 'Safari Books Online'),  
    'released'  => '2010-01-08',  
    'pages'     => 192  
);  
$client = new Zend_Http_Client();  
$client->setUri('http://127.0.0.1:5984/books');  
$client->setHeaders('Content-Type', 'application/json');  
$client->setRawData(Zend_Json::encode($book));  
$response = $client->request(Zend_Http_Client::POST);  
print_r($response->getStatus()); // 201
```

Releases View



1. In the books database in Futon, select *Temporary view...* from the *View* select menu
2. Enter the following into the *View Code* text area:

```
function(doc) {  
  if (doc.released) {  
    emit(doc.released.split("-"), doc.pages);  
  }  
}
```
3. Enter the following in the *Reduce Function* text area:

```
_stats
```
4. Click *Save As...*
5. Enter "default" as the *Design Document* name
6. Enter "releases" as the *View Name*
7. Click *Save*

Exact Grouping

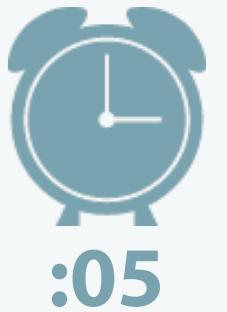


```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/releases');  
$client->setParameterGet('group', 'true');  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

Exact Grouping

key	value
<pre>Array ([0] => 1999 [1] => 10 [2] => 28)</pre>	<pre>Array ([sum] => 648 [count] => 1 [min] => 648 [max] => 648 [sumsqr] => 419904)</pre>
<pre>Array ([0] => 2007 [1] => 05 [2] => 08)</pre>	<pre>Array ([sum] => 448 [count] => 1 [min] => 448 [max] => 448 [sumsqr] => 200704)</pre>
<pre>Array ([0] => 2010 [1] => 01 [2] => 08)</pre>	<pre>Array ([sum] => 192 [count] => 1 [min] => 192 [max] => 192 [sumsqr] => 36864)</pre>
<pre>Array ([0] => 2010 [1] => 01 [2] => 19)</pre>	<pre>Array ([sum] => 272 [count] => 1 [min] => 272 [max] => 272 [sumsqr] => 73984)</pre>

Group Level 1

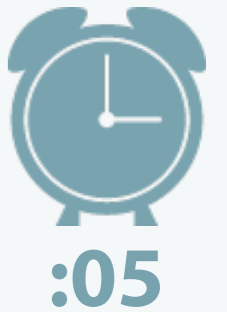


```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/releases');  
$client->setParameterGet('group_level', 1);  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```


Group Level 1

key	value
Array ([0] => 1999)	Array ([sum] => 648 [count] => 1 [min] => 648 [max] => 648 [sumsqr] => 419904)
Array ([0] => 2007)	Array ([sum] => 448 [count] => 1 [min] => 448 [max] => 448 [sumsqr] => 200704)
Array ([0] => 2010)	Array ([sum] => 464 [count] => 2 [min] => 192 [max] => 272 [sumsqr] => 110848)

Group Level 2



```
$db = 'http://127.0.0.1:5984/books';
$client = new Zend_Http_Client();
$client->setUri($db . '/_design/default/_view/releases');
$client->setParameterGet('group_level', 2);
$client->setHeaders('Accept', 'application/json');
$response = $client->request(Zend_Http_Client::GET);
$body = $response->getBody();
$results = Zend_Json::decode($body);
print_r($results);
```

Group Level 2

key	value
<pre>Array ([0] => 1999 [1] => 10)</pre>	<pre>Array ([sum] => 648 [count] => 1 [min] => 648 [max] => 648 [sumsqr] => 419904)</pre>
<pre>Array ([0] => 2007 [1] => 05)</pre>	<pre>Array ([sum] => 448 [count] => 1 [min] => 448 [max] => 448 [sumsqr] => 200704)</pre>
<pre>Array ([0] => 2010 [1] => 01)</pre>	<pre>Array ([sum] => 464 [count] => 2 [min] => 192 [max] => 272 [sumsqr] => 110848)</pre>

Including Documents

Including Docs



```
$db = 'http://127.0.0.1:5984/books';  
$client = new Zend_Http_Client();  
$client->setUri($db . '/_design/default/_view/titles');  
$client->setParameterGet('reduce', 'false');  
$client->setParameterGet('include_docs', 'true');  
$client->setHeaders('Accept', 'application/json');  
$response = $client->request(Zend_Http_Client::GET);  
$body = $response->getBody();  
$results = Zend_Json::decode($body);  
print_r($results);
```

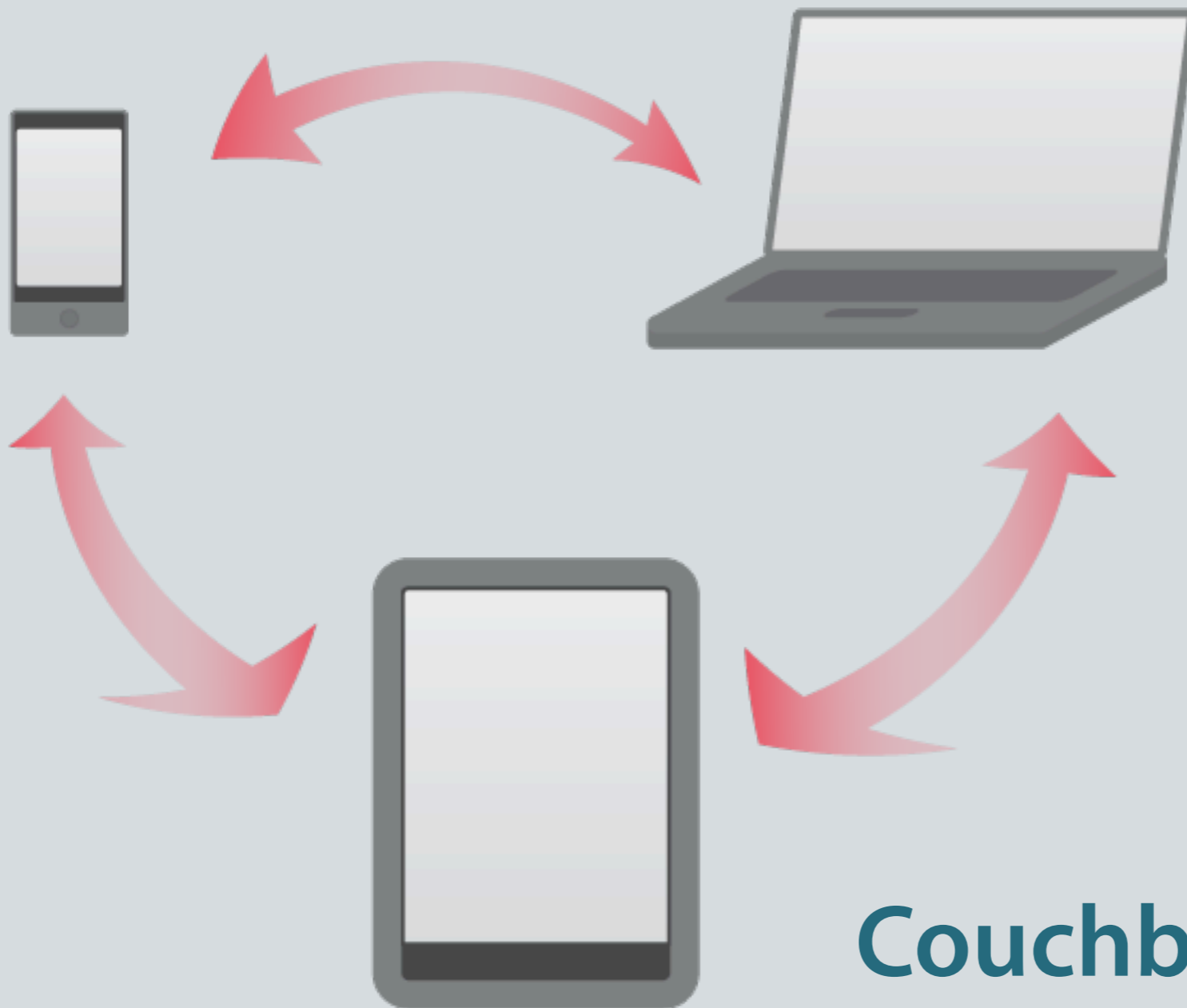
Including Docs

id	key	value	doc
"978-0-596-80579-1"	"Building iPhone Apps with HTML, CSS, and JavaScript"	192	Array ([_id] => 978-0-596-80579-1 [_rev] => 1-09ce09fef75068834da99957c7b14cf2 ...)
"978-0-596-15589-6"	"CouchDB: The Definitive Guide"	272	Array ([_id] => 978-0-596-15589-6 [_rev] => 2-099d205cbb59d989700ad7692cbb3e66 ...)
"978-1-565-92580-9"	"DocBook: The Definitive Guide"	648	Array ([_id] => 978-1-565-92580-9 [_rev] => 1-b945cb4799a1ccdd1689eae0e44124f1 ...)
"978-0-596-52926-0"	"RESTful Web Services"	448	Array ([_id] => 978-0-596-52926-0 [_rev] => 2-de467b329baf6259e791b830cc950ece ...)

More CouchDB Features

Multi-Master Replication

- peer-based and bi-directional
- no explicit cluster setup required
- synchronization is incremental
- changes feed allows for replication to other systems



Couchbase Mobile^[1] for iOS and Android

1. <http://www.couchbase.com/products-and-services/couchbase-mobile>

Horizontally Scalable

- **shared nothing architecture**
- **non-blocking reads and writes for high concurrency**
- **fault tolerant, isolated messages**

ACID Compliant

Atomicity: individual document updates are “all or nothing”

Consistency: queries will be “point in time” consistent

Isolation: updates isolated to document revision number

Durability: success code is returned only after data is on disk

Authentication

- **special *_users* database**
- **basic access authentication**
- **cookie authentication**
- **OAuth support**

Security Model

- **server admins**
- **database admins**
- **database readers**
- **document-level authorization with validation functions**

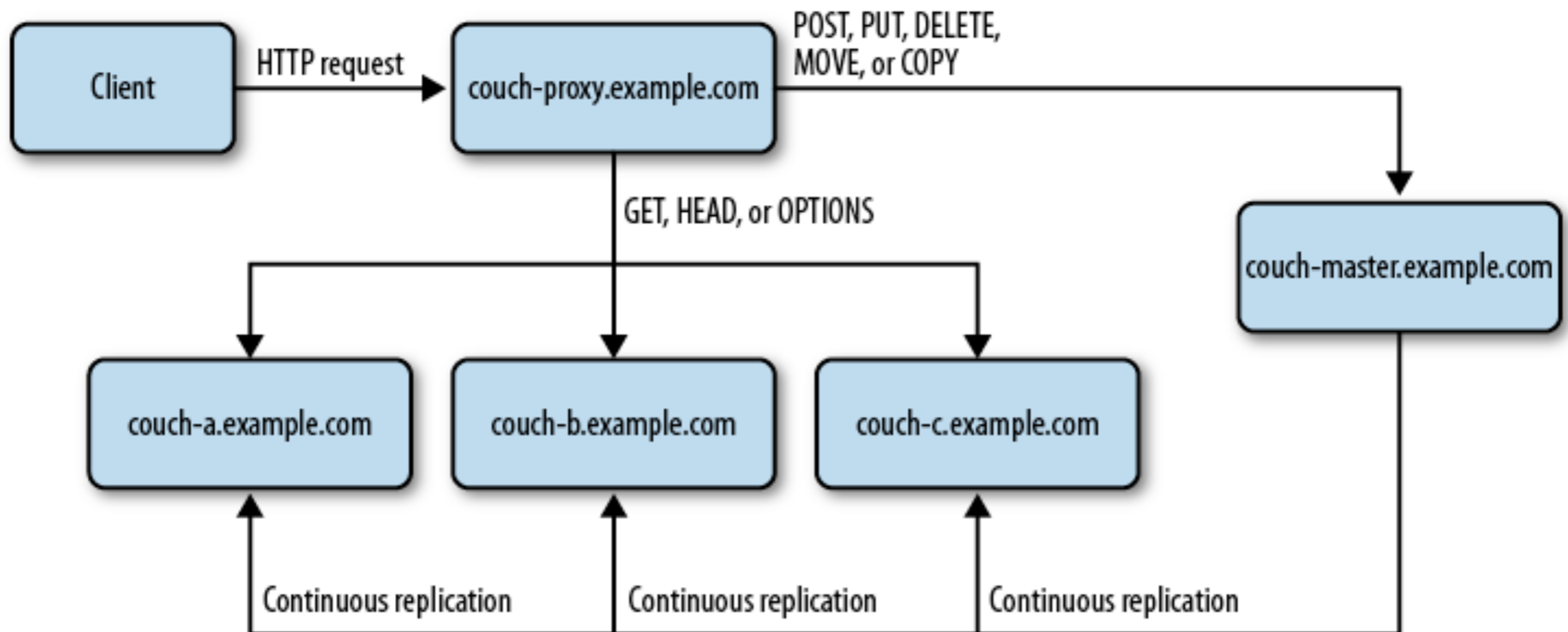
Web Server

- **static files as attachments (e.g. HTML, Javascript, CSS)**
- **URL rewriting**
- **HTTP proxying**
- **HTTP range requests**
- **SSL support**
- **virtual hosts**

Application Server

- *show* functions
- *list* functions
- document update handlers

Load Balancing



Clustering

- Lounge^[1]
- Pillow^[2]
- BigCouch^[3]

1. <http://tilgovi.github.com/couchdb-lounge/>
2. <https://github.com/khellan/Pillow>
3. <https://github.com/cloudant/bigcouch>

Extensible

- **GeoCouch^[1]** for geospatial indexing
- **couchdb-lucene^[2]** for full text indexing
- **ElasticSearch^[3]** for full text indexing

1. <https://github.com/couchbase/geocouch>

2. <https://github.com/rnewson/couchdb-lucene>

3. <http://www.elasticsearch.org/tutorials/2010/08/01/couchb-integration.html>

Resources

Hosting

Iris Couch

- Free (for now)
- Couchbase server
- GeoCouch geospatial indexing
- <http://www.iriscouch.com>

Cloudant

- Various pricing levels
- Full-text searching
- Ad-hoc queries
- Elastic scaling, clustering and sharding using BigCouch
- <https://cloudant.com>

Documentation and Books

CouchDB Wiki

<http://wiki.apache.org/couchdb/>

CouchDB: The Definitive Guide

**by J. Chris Anderson, Jan
Lehnardt and Noah Slater
(O'Reilly)**

978-0-596-15589-6

***Writing and Querying MapReduce
Views in CouchDB***

by Bradley Holt (O'Reilly)

978-1-449-30312-9

Scaling CouchDB

by Bradley Holt (O'Reilly)

978-1-4493-0343-3

Getting Started with CouchDB

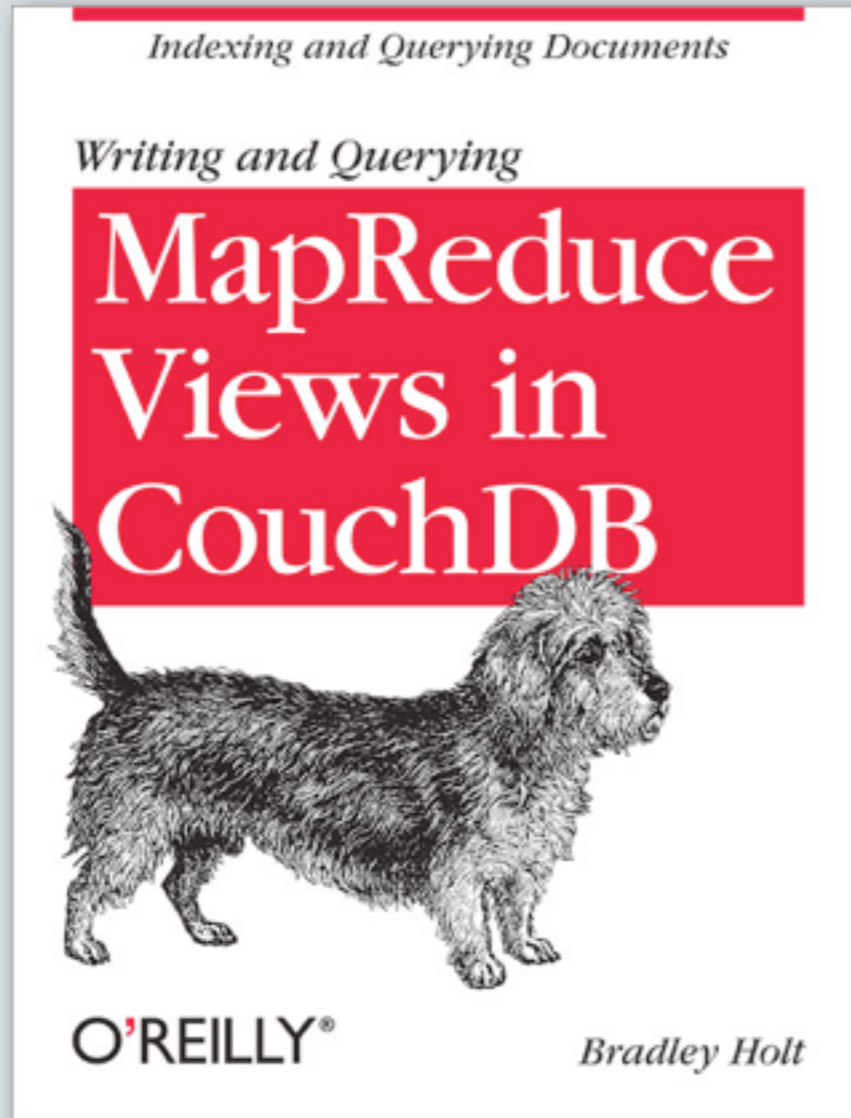
by MC Brown (O'Reilly)

978-1-4493-0755-4

Beginning CouchDB

by Joe Lennon (Apress)

978-1-430-27237-3



<http://oreilly.com/catalog/9781449303129/>



<http://oreilly.com/catalog/9781449303433/>

Questions?

Thank You

[@BradleyHolt](#)

<http://bradley-holt.com>

bradley.holt@foundline.com