

GETTING STARTED WITH MONGODB FOR CFML DEVELOPERS

Dan Skaggs, Co-Founder
Web-Meister Designs

<http://TNTechnoHermit.com>



Why you're here

- Hear about what MongoDB is
- See how easy it is to install for development
- Learn how to start putting data in and getting data out with Adobe ColdFusion and Railo CFML engines
- Hopefully not expecting to know everything there is to know about MongoDB when you leave here today
 - Session is designed to “whet your appetite”



Obligatory “About Me” slide

- Began experimenting with HTML in 1997 after the birth of our twins while on active duty with the U.S. Air Force
- Staff developer for a company headquartered in San Jose, CA for 8 1/2 years
- Began with Allaire ColdFusion 4.5 in 2000 as part of a Spectra 1.0 project
- Have developed on every Allaire/Macromedia/Adobe ColdFusion version since then *
- Have done several projects of varying sizes with Railo
- Independent consultant since 2006 working with clients of all sizes across the U.S. building applications on multiple CFML engines and versions
- Contributing partner and build master for the Model-Glue MVC framework for CFML
- Also enjoy shooting, coaching basketball, amateur (HAM) radio and video games

What is MongoDB?



Long, boring definition

[1] MongoDB is an open-source document database, and the leading NoSQL database. Written in C++, MongoDB features document-oriented storage, full index support, replication and high availability...

ZZZZZZZZZZZZZZ...

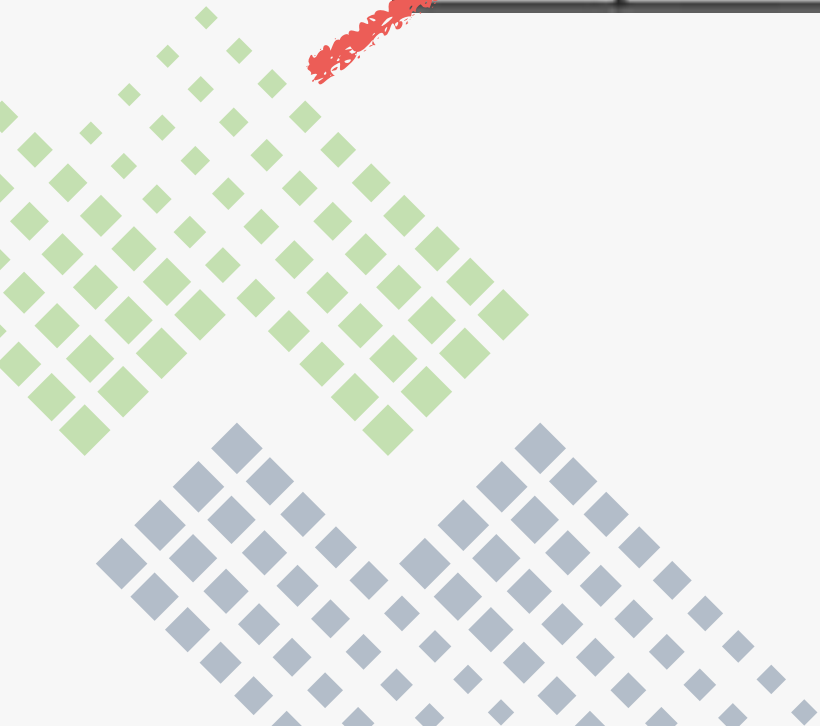
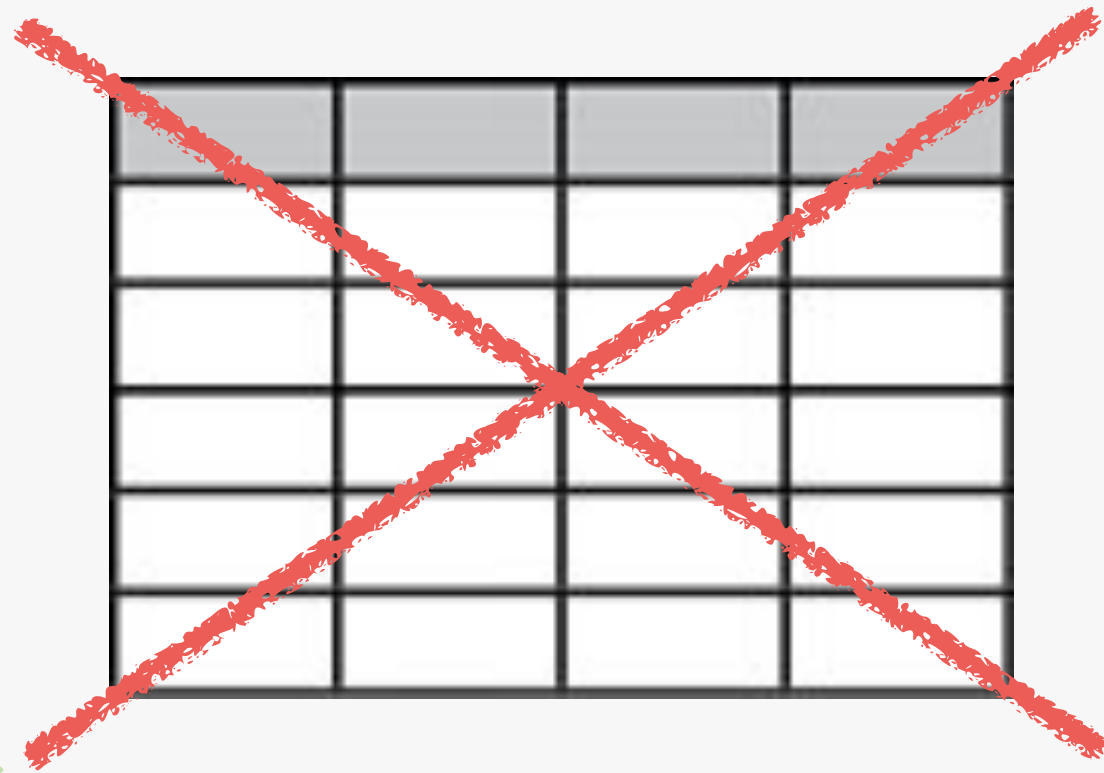
Usable definition

MongoDB is a different way of storing data than what we've been used to with relational databases.

DIFFERENT HOW!?!?



TABLE/ROW vs DOCUMENT



Document-Oriented Storage

```
{  
  "city" : "MINNEAPOLIS",  
  "latitude": 44.967345,  
  "longitude": -93.282841,  
  "pop" : 14098,  
  "state" : "MN",  
  "_id" : "55403"  
}
```



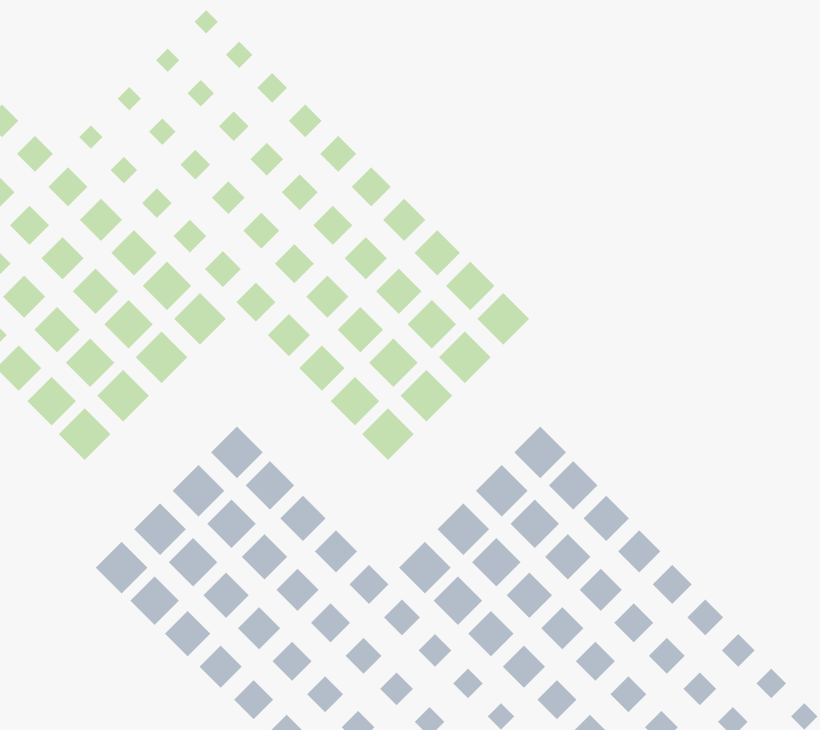
OTHER DIFFERENCES

“Schema-less”*

MongoDB doesn't care if not all documents in a collection have the same fields

Embed Data vs Foreign Key Linkage

MongoDB preference is to embed related data as a nested document



Document-Oriented Storage

```
{  
  "city" : "MINNEAPOLIS",  
  "loc" : [  
    -93.282841,  
    44.967345  
  ],  
  "postoffice": {  
    "address1": "100 S 1ST ST",  
    "phone": "612-349-4713",  
    "tty": "877-889-2457"  
  },  
  "pop" : 14098,  
  "state" : "MN",  
  "_id" : "55403"  
}
```

OTHER DIFFERENCES

“Schema-less”*

MongoDB doesn't care if not all documents in a collection have the same fields

Embed Data vs Foreign Key Linkage

MongoDB preference is to embed related data as a nested document

Easily Replicated

Easily replicate data across 3+ MongoDB instances for fault tolerance and performance improvements

WHY MONGODB?



THE GOOD, THE BAD, THE...

- It's **F**AST
- It's **F**LEXIBLE
- It's **F**ORGIVING
- **N**o transactions
- **N**o JOINS
- **N**o data write guarantee *
- **N**o documents over 16MB
- **N**o direct cross-over from SQL language most of us are used to



Installation



MAC / *NIX INSTALLATIONS

Using A Package Manager

```
brew install mongodb  
  
yum install mongo-10gen mongo-10gen-server  
  
apt-get install mongodb-10gen
```

Download & Unzip



<http://www.mongodb.org/downloads>

WINDOWS INSTALLATIONS

Download & Unzip



<http://www.mongodb.org/downloads>

Using Chocolatey Package Manager

```
cinst mongodb
```

Install as a Windows Service

```
C:\mongodb\bin\mongod.exe  
--config C:\mongodb\mongod.cfg --install
```


OK, NOW WHAT?

Hint: Run It!

Mac / *nix

```
mongod
```

Windows

```
c:\mongodb\bin\mongod.exe
```

SUCCESS! Or, wait a minute....



Whoops

```
Macintosh HD — bash — 120x28
Dans-Macbook-Pro:/ dskaggs$ mongod
mongod --help for help and startup options
Sun Apr  6 19:46:02.059 [initandlisten] MongoDB starting : pid=23449 port=27017 dbpath=/data/db/ 64-bit host=Dans-Macbook-Pro.local
Sun Apr  6 19:46:02.060 [initandlisten] db version v2.4.9
Sun Apr  6 19:46:02.060 [initandlisten] git version: 52fe0d21959e32a5bdbecdc62057db386e4e029c
Sun Apr  6 19:46:02.060 [initandlisten] build info: Darwin bs-osx-106-x86-64-2.10gen.cc 10.8.0 Darwin Kernel Version 10.8.0: Tue Jun  7 16:32:41 PDT 2011; root:xnu-1504.15.3~1/RELEASE_X86_64 x86_64 BOOST_LIB_VERSION=1_49
Sun Apr  6 19:46:02.060 [initandlisten] allocator: system
Sun Apr  6 19:46:02.060 [initandlisten] options: {}
Sun Apr  6 19:46:02.060 [initandlisten] exception in initAndlisten: 10296
*****
ERROR: dbpath (/data/db/) does not exist.
Create this directory or give existing directory in --dbpath.
See http://dochub.mongodb.org/core/startingandstoppingmongo
*****
, terminating
Sun Apr  6 19:46:02.060 dbexit:
Sun Apr  6 19:46:02.060 [initandlisten] shutdown: going to close listening sockets...
Sun Apr  6 19:46:02.060 [initandlisten] shutdown: going to flush diaglog...
Sun Apr  6 19:46:02.060 [initandlisten] shutdown: going to close sockets...
Sun Apr  6 19:46:02.060 [initandlisten] shutdown: waiting for fs preallocator...
Sun Apr  6 19:46:02.060 [initandlisten] shutdown: lock for final commit...
Sun Apr  6 19:46:02.060 [initandlisten] shutdown: final commit...
Sun Apr  6 19:46:02.060 [initandlisten] shutdown: closing all files...
Sun Apr  6 19:46:02.060 [initandlisten] closeAllFiles() finished
Sun Apr  6 19:46:02.060 dbexit: really exiting now
Dans-Macbook-Pro:/ dskaggs$
```

DATA DIRECTORIES

MongoDB does not auto create your data directories

Mac / *nix

```
/data/db
```

Windows

```
c:\data\db
```

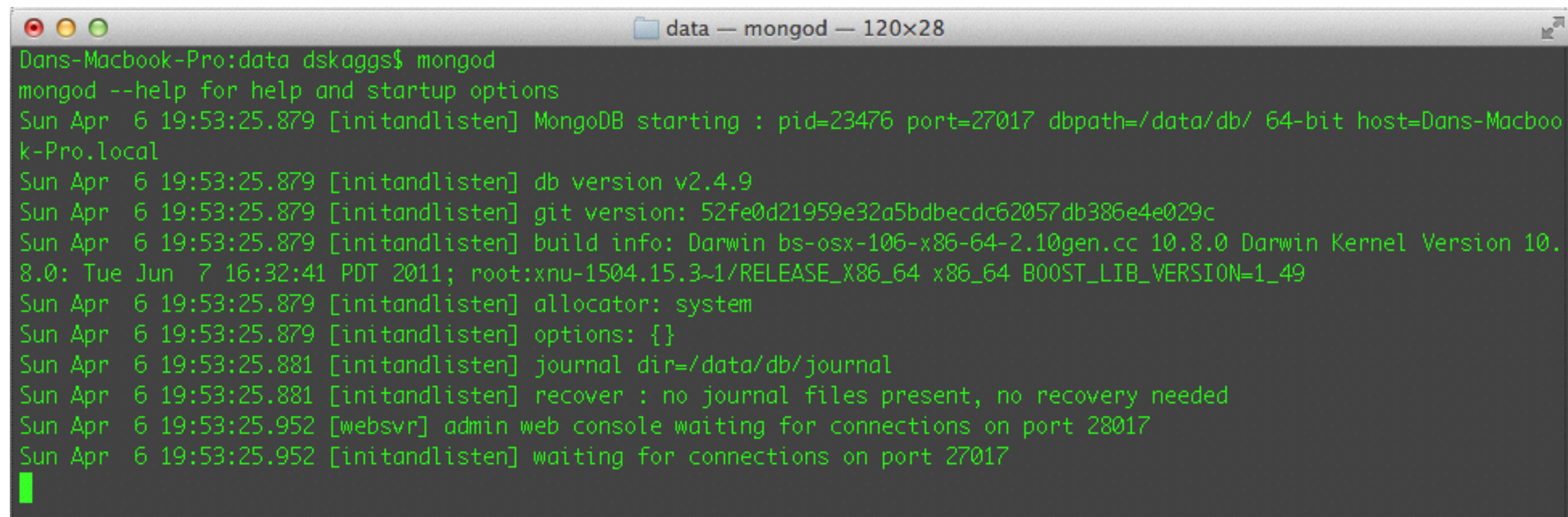
Use a Custom Data Directory

```
mongod --dbpath /path/to/my/folder  
mongod.exe --dbpath "d:\path\to\my\folder"
```

Maybe this time.....



SUCCESS!

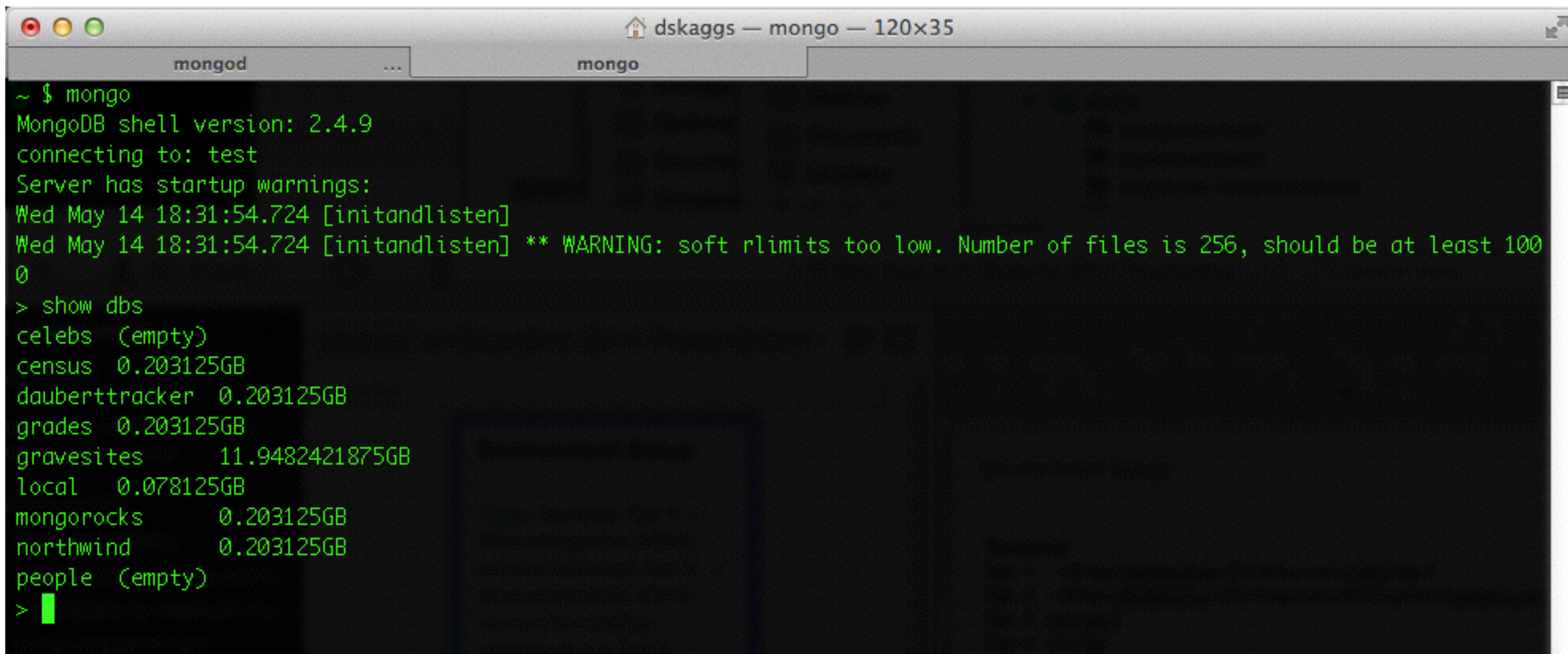
A screenshot of a macOS terminal window titled "data — mongod — 120x28". The terminal shows the output of the "mongod" command. The logs indicate that MongoDB is starting successfully on a Macbook Pro. The logs include the following information:

```
Dans-Macbook-Pro:data dskaggs$ mongod
mongod --help for help and startup options
Sun Apr  6 19:53:25.879 [initandlisten] MongoDB starting : pid=23476 port=27017 dbpath=/data/db/ 64-bit host=Dans-Macbook-Pro.local
Sun Apr  6 19:53:25.879 [initandlisten] db version v2.4.9
Sun Apr  6 19:53:25.879 [initandlisten] git version: 52fe0d21959e32a5bdbecdc62057db386e4e029c
Sun Apr  6 19:53:25.879 [initandlisten] build info: Darwin bs-osx-106-x86-64-2.10gen.cc 10.8.0 Darwin Kernel Version 10.8.0: Tue Jun  7 16:32:41 PDT 2011; root:xnu-1504.15.3~1/RELEASE_X86_64 x86_64 B00ST_LIB_VERSION=1_49
Sun Apr  6 19:53:25.879 [initandlisten] allocator: system
Sun Apr  6 19:53:25.879 [initandlisten] options: {}
Sun Apr  6 19:53:25.881 [initandlisten] journal dir=/data/db/journal
Sun Apr  6 19:53:25.881 [initandlisten] recover : no journal files present, no recovery needed
Sun Apr  6 19:53:25.952 [websvr] admin web console waiting for connections on port 28017
Sun Apr  6 19:53:25.952 [initandlisten] waiting for connections on port 27017
```

GETTING CONNECTED



Mongo Shell



The screenshot shows a macOS terminal window titled "dskaggs — mongo — 120x35". The terminal has two tabs: "mongod" and "mongo". The "mongo" tab is active, displaying the MongoDB shell version 2.4.9 and connection status to the "test" database. It shows startup warnings, including a warning about soft rlimits. The user has entered the command "show dbs", which lists the databases and their sizes. The databases listed are celebs (empty), census (0.203125GB), dauberttracker (0.203125GB), grades (0.203125GB), gravesites (11.9482421875GB), local (0.078125GB), mongorocks (0.203125GB), northwind (0.203125GB), and people (empty). The prompt ">" is visible at the bottom.

```
~ $ mongo
MongoDB shell version: 2.4.9
connecting to: test
Server has startup warnings:
Wed May 14 18:31:54.724 [initandlisten]
Wed May 14 18:31:54.724 [initandlisten] ** WARNING: soft rlimits too low. Number of files is 256, should be at least 1000
> show dbs
celebs (empty)
census 0.203125GB
dauberttracker 0.203125GB
grades 0.203125GB
gravesites 11.9482421875GB
local 0.078125GB
mongorocks 0.203125GB
northwind 0.203125GB
people (empty)
>
```


Railo MongoDB Extension

The screenshot displays the Railo Server Administrator interface. The top navigation bar includes the Railo logo and tabs for 'Server Administrator' and 'Web Administrator'. The left sidebar contains a search bar and a menu with categories: Settings, Services, Extension, Remote, and Archives & Resources. The 'Extension' category is expanded, showing 'Applications' and 'Providers'. The main content area is titled 'Extension - Applications' and includes a search bar and a 'filter' button. Below this, a grid of application tiles is shown, each with an icon, name, and status. The 'MongoDB (Beta)' tile is highlighted with a red box, and a red arrow points to it from the 'Extension' menu in the sidebar.

Extension - Applications [Favorites](#) [Log out](#)

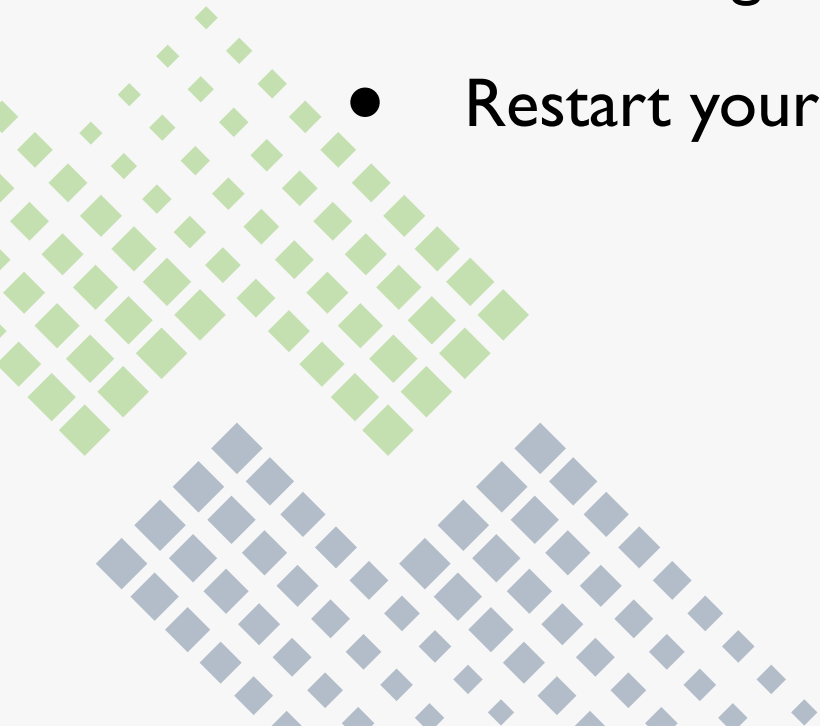
Not installed
These applications are not yet installed on the system.

Search term: [filter](#)

| | | | | |
|-----------------------------------|---|---|---|--|
| CFBEER Free | cfmarkdown Free | cfmediawiki Free | ColdFunctional Free | CouchDB Cache (Beta) Free |
| EHCACHE Free | FusionReactor Server Monitor Free | JBoss Cache (Beta) 49 USD | JMS Gateway (Preview) 299 USD | Log analyzer global installer Free |
| MongoDB (Beta) Free | MongoDB Cache (Beta) Free | Railo Memory Monitor 49 USD | Railo Scope Monitor 49 USD | RedisCache Free |

CFMongoDB Installation

- Download CFMongoDB from GitHub (<https://github.com/marcesher/cfmongodb>)
- Extract to your web root or to an external location and set up a mapping
- Copy the following files from the /lib folder to a folder that's in the class path for your CFML server (location varies by engine / version)
 - mongodb[version].jar
 - cfmongodb.jar
- Restart your CFML server



ABOUT THE DATA

MongoDB version of Northwind database

customers

orders

orderdetails

products

suppliers

....more....



Connecting Your Code

Railo Extension

```
application.mongo=MongoDBConnect("northwind","localhost", 27017);
```

CFMongoDB

```
mongoConfig = createObject(  
    'component',  
    'cfmongodb.core.MongoConfig'  
).init(dbName="northwind");  
  
application.mongo = createObject(  
    'component',  
    'cfmongodb.core.MongoClient'  
).init(mongoConfig);
```

TO THE DATA!



findOne()

```
db.customers.findOne(  
{  
  "_id" : ObjectId("51ba0970ae4ad8cc43bb95e3"),  
  "CustomerID" : "ALFKI",  
  "CompanyName" : "Alfreds Futterkiste",  
  "ContactName" : "Maria Anders",  
  "ContactTitle" : "Sales Representative",  
  "Address" : "Obere Str. 57",  
  "City" : "Berlin",  
  "Region" : "NULL",  
  "PostalCode" : 12209,  
  "Country" : "Germany",  
  "Phone" : "030-0074321",  
  "Fax" : "030-0076545"  
})
```

find()

```
> db.customers.find({'Country':'Germany'}).pretty()
{
  "_id" : ObjectId("51ba0970ae4ad8cc43bb9638"),
  "CustomerID" : "WANDK",
  "CompanyName" : "Die Wandernde Kuh",
  "ContactName" : "Rita Müller",
  "ContactTitle" : "Sales Representative",
  "Address" : "Adenauerallee 900",
  "City" : "Stuttgart",
  "Region" : "NULL",
  "PostalCode" : 70563,
  "Country" : "Germany",
  "Phone" : "0711-020361",
  "Fax" : "0711-035428"
}
```

+ 10 more records

CFMongoDB

Railo Extension

CFMongoDB Search DSL



Limiting Returned Data

```
> db.customers.find(  
    {'Country':'Germany'},  
    {'ContactName': 1, 'Phone': 1, 'Fax': 1}  
).pretty();  
  
{  
  "_id" : ObjectId("51ba0970ae4ad8cc43bb95e3"),  
  "ContactName" : "Maria Anders",  
  "Phone" : "030-0074321",  
  "Fax" : "030-0076545"  
}  
  
+ 10 more records
```

CFMongoDB

Railo Extension

Adding A Record

```
> newCustomer = {  
    "CustomerID" : "TEST123",  
    "CompanyName" : "Kevin's Klow'n Kostumes",  
    "ContactName" : "Kevin Kennedy",  
    "ContactTitle" : "Owner",  
    "Address" : "Jagdhaus 77",  
    "City" : "Kaiserslautern",  
    "Region" : "NULL",  
    "PostalCode" : 67661,  
    "Country" : "Germany",  
    "Phone" : "0251-037483",  
    "Fax" : "0251-035112"  
};  
> db.customers.save( newCustomer );
```


Deleting A Record

MongoDB Shell

```
> db.customers.remove({"CustomerID" : "TEST123"});
```

CFMongoDB

```
collection = application.mongo.getDBCollection( "customers" );  
collection.remove({'CustomerID': 'TEST123'});
```

Railo Extension

```
application.mongo.customers.remove({'CustomerID': 'TEST123'});
```



REPORTING!?!?!?



Products Document

```
> db.products.find({'ProductID': 77}).pretty()
{
  "_id" : ObjectId("51ba0971ae4ad8cc43bba270"),
  "ProductID" : 77,
  "ProductName" : "Original Frankfurter grüne Soße",
  "SupplierID" : 12,
  "CategoryID" : 2,
  "QuantityPerUnit" : "12 boxes",
  "UnitPrice" : 13,
  "UnitsInStock" : 12,
  "UnitsOnOrder" : 0,
  "ReorderLevel" : 25,
  "Discontinued" : 0
}
```

Products On Order

```
> db.products.aggregate([
  { "$match": { Discontinued: 0, UnitsOnOrder: {'$gt': 0} } }
]);

> {"result" : [
  {
    "_id" : ObjectId("51ba0971ae4ad8cc43bba225"),
    "ProductID" : 2,
    "ProductName" : "Chang",
    "SupplierID" : 1,
    "CategoryID" : 1,
    "QuantityPerUnit" : "24 - 12 oz bottles",
    "UnitPrice" : 19,
    "UnitsInStock" : 17,
    "UnitsOnOrder" : 40,
    "ReorderLevel" : 25,
    "Discontinued" : 0
  },
  ... 16 other products ....
]}
```

Products On Order v2

```
> db.products.aggregate([
  { "$match": { Discontinued: 0, UnitsOnOrder: {'$gt': 0} } }
, { "$project": { _id: 0, SupplierID: 1, ProductID: 1
                  , TotalOrder: { $multiply:["$UnitPrice", "$UnitsOnOrder"] } } }
]);

> {"result" : [
  {
    "ProductID" : 2,
    "SupplierID" : 1,
    "TotalOrder" : 760
  },
  {
    "ProductID" : 3,
    "SupplierID" : 1,
    "TotalOrder" : 700
  }
  ... 15 other products ....
]}
```

Products On Order v3

```
> db.products.aggregate([
  { $match: {Discontinued: 0, UnitsOnOrder: {'$gt': 0}} },
  { $project: { _id: 0, SupplierID: 1, ProductID: 1
                , TotalOrder: { $multiply:['$UnitPrice', '$UnitsOnOrder']} } }
  , { $group : { _id : "$SupplierID", SupplierTotal : { $sum : '$TotalOrder' } } }
  , { $sort : { _id: 1 } }
]);
> {"result" : [
  {
    "_id" : 1,
    "SupplierTotal" : 1460
  },
  {
    "_id" : 2,
    "SupplierTotal" : 1700
  },
  ... 12 other suppliers ....
]}
```

Product Reorders Needed

```
> db.products.aggregate([
  { "$project": {
    _id: 0,
    SupplierID: 1,
    ProductID: 1,
    Discontinued: 1,
    UnitsOnOrder: 1,
    StockRatio: { $subtract:["$UnitsInStock", "$ReorderLevel"] }
  }
  , { "$match": { Discontinued: 0
    , UnitsOnOrder: 0
    , StockRatio: {'$lte': 0}
  }
}]
```

WHAT'S NEXT?



Offerings

M101J: MongoDB for Java Developers



M101JS: MongoDB for Node.js Developers



M101P: MongoDB for Developers



M102: MongoDB for DBAs



M202: MongoDB Advanced Deployment and Operations



Resources

- MongoDB University
- CFMongoDB GitHub Repo Wiki
- Railo Google Group
- Stack Overflow
<http://stackoverflow.com/questions/tagged/mongodb>
- Free Support Forum
<http://groups.google.com/group/mongodb-user>
- IRC Chat and Support
<irc://irc.freenode.net/#mongodb>
- Commercial Support

Questions?

Dan Skaggs, Co-Founder
Web-Meister Designs

Twitter

@dskaggs

Email/GTalk

dan.skaggs@web-meister.com

Blog

<http://TNTechnoHermit.com>

Skype

dan.skaggs

GitHub

<http://www.github.com/dskaggs>

