# Setting Up CouchDB and Lucene

### This page provides the guidelines to install and run CouchDB server on your machine along with the Lucene search engine.

| | |
|---|---|
| Environment : | Debian |
| Environment : | Mac OS X 10.5 |

## Erlang and other dependencies:

To configure and build Couchdb you would need Erlang V. 5.6 or higher. If you are using Ubuntu 8.10 and higher the package with the proper version is already included in synaptic so you can simple apt-get it (howto: is described below). This is the complete list of dependencies that you need to have for Erlang/Couchdb:

```
sudo apt-get install automake autoconf libtool subversion-tools help2man
sudo apt-get install build-essential erlang libicu38 libicu-dev
sudo apt-get install libreadline5-dev checkinstall libmozjs-dev wget
sudo apt-get install libcurl4-openssl-dev
```

In case you are using earlier versions of Ubuntu/Debian or you would need the latest version of Erlang available. You have to build it yourself instead of installing it with apt-get:

- Download and unpack the latest version of Erlang from their website (current latest version is R13B01):

  ```
  wget http://erlang.org/download/otp_src_R13B01.tar.gz
  tar -xzf otp_src_R13B01.tar.gz
  ```

- Next go into the unpacked directory, configure and build Erlang:

  ```
  cd otp_src_R13B01
  ./configure
  make
  sudo make install
  ```

- The building step will take a while and once you are done you can check if Erlang is installed and you have a proper version (greater then or equal to 5.6):

  ```
  erl -V
  ```

  In case you get an error or wrong version (might happen if you still have other Erlang package installed) you might want to create a link to the new built:

  ```
  sudo ln -s /path/to/new-built-erlang/bin/erl /usr/bin/erl
  ```

  Check again and make sure that the version is updated.

## Couchdb:

Great! By now you should have all necessary packages and dependencies installed and you are ready to go on with installing Couchdb.
There are 2 options to reach the step when you are ready to configure and build Couchdb:

- First one is to download and bootstrap the source:

  ```
  svn co http://svn.apache.org/repos/asf/couchdb/trunk couchdb
  cd couchdb
  ./bootstrap
  ```

- Second (more stable) is to download and unpack the latest release of Couchdb (current latest version is 0.9.0):

  ```
  wget http://mirror.csclub.uwaterloo.ca/apache/couchdb/0.9.0/apache-couchdb-0.9.0.tar.gz
  tar -xzvf apache-couchdb-0.9.0.tar.gz
  ```

At this point regardless of the approach you took you should be able to proceed configuring and building Couchdb:

```
./configure
make
sudo make install
make clean
make distclean
sudo -i
adduser --system --home /usr/local/var/lib/couchdb --no-create-home --shell /bin/bash --group --gecos "CouchDB Administrator" couchdb
chown -R couchdb:couchdb /usr/local/var/lib/couchdb
chown -R couchdb:couchdb /usr/local/var/log/couchdb
chown -R couchdb:couchdb /usr/local/var/run
chown -R couchdb:couchdb /usr/local/etc/couchdb
chmod -R 0770 /usr/local/var/lib/couchdb
chmod -R 0770 /usr/local/var/log/couchdb
chmod -R 0770 /usr/local/var/run
chmod -R 0770 /usr/local/etc/couchdb
cp /usr/local/etc/init.d/couchdb /etc/init.d/
update-rc.d couchdb defaults
exit
```

Now the Couchdb should be installed and you should be able to run it by typing:

```
sudo /etc/init.d/couchdb start
```

To check if it is running open your browser and type in:

```
localhost:5984
```

By default CouchDB listens only for connections from the local host. To change that edit */usr/local/etc/couchdb/local.ini*. You should modify the following lines:

```
port = 5984
bind_address = 0.0.0.0
```

Great, so Couchdb is installed next step is to do the same if couchdb-lucene.

## Couchdb-Lucene

To install couchdb-lucene make sure you have git as well as maven2 installed on your maching first. If not install it by typing:

```
sudo apt-get install git-core maven2
```

Once you are done download the source:

```
git clone git://github.com/rnewson/couchdb-lucene.git
```

Next step is to build everything:

```
cd couchdb-lucene
mvn
```

After finishing building you should have an assembled jar file in the target sub-directory called *couchdb-lucene-*-jar-with-dependencies.jar*.

## Setting up Couchdb-Lucene

Great we are getting closer and next steps will let us set up lucene search engine with our Couch database.
The file we are going to modify contains various configuration options of the database and it is located in */usr/local/etc/couchdb/local.ini* (the same file where we changed the ip address before). These are the options that need to be added or modified:

```
[couchdb]
os_process_timeout=60000

[external]
fti=/usr/bin/java -jar /path/to/couchdb-lucene-*-jar-with-dependencies.jar -search

[update_notification]
indexer=/usr/bin/java -jar /path/to/couchdb-lucene-*-jar-with-dependencies.jar -index

[httpd_db_handlers]
_fti = {couch_httpd_external, handle_external_req, <<"fti">>}
```

**NOTE:** There was a serious issue that I faced during the further steps in the process that is probably the best to address here. Couchdb-Lucene needs to have write access to the directory where it saves the indexes. However the path to it is relative to Couchdb. I found that the best way to make sure the path is consistent is to pass it as a system property value in the same local.ini file:

```
[external]
fti=/usr/bin/java -Dcouchdb.lucene.dir=/path/to/indexing/dir -jar /path/to/couchdb-lucene-*-jar-with-
dependencies.jar -search

[update_notification]
indexer=/usr/bin/java -Dcouchdb.lucene.dir=/path/to/indexing/dir -jar /path/to/couchdb-lucene-*-jar-with-
dependencies.jar -index
```

Again, make sure that couchdb has write access to that directory:

```
chown -R couchdb:couchdb /path/to/indexes/lucene
chmod -R 0770 /path/to/indexes/lucene
```

Next step is to add a design document to the database that couchdb-lucene hooks up to and indexes according to. Here we assume that there is already a database with a number of documents saved in it. The easiest way to add a design document is to do it in futon. Go to *"your_database_ip":5984/_utils*, proceed to your database and select *"Design documents"* in the dropdown. Next is to click on *"Create document ..."* and name it _"*design/lucene"* (the prefix in the name identifies the design document). The last thing is to add the new *"fulltext"* field to the document that can contain one or more views used in searching/indexing. For example if you want to index all elements in the document the value for that field will look like that:

```
{
"all": {
   "defaults": {
       "store": "no"
    },
   "index": "function(doc) {var ret = new Document();function idx(obj) {for (var key in obj) {switch (typeof
obj[key]) {case 'object':idx(obj[key]);break;case 'function':break;default:ret.add(obj[key]);break;}}};
idx(doc);if (doc._attachments) {for (var i in doc._attachments) {ret.attachment("attachment", i);}}return ret;}"
}
}
```

Make sure you save the document.
You might also need to restart the database simply by typing:

```
sudo /etc/init.d/couchdb restart
```

Now you are finished setting up Couchdb with Lucene, you can try querying the database like this:

```
curl http://"your_database_ip":5984/database_name/_fti/design_doc/view_name?q=Query
```

## couchdb-python

### Download all dependencies:

```
sudo aptitude install python-simplejson
sudo aptitude install python-httplib2
```

Download the latest version of couchdb-python project (.egg).

Run the following scripts to install couchdb-python:

```
wget http://peak.telecommunity.com/dist/ez_setup.py
sudo python ez_setup.py
wget "path to downloaded .egg file"
sudo easy_install "path to downloaded .egg file"
```

### pouch

### couchdbkit

## Installing and running CouchDB on Windows

CouchDB Wiki

## Installing and running CouchDB on Mac OS X

Setting up CouchDB on Mac OS X