

# Documentation Guide



## Become a contributor

You need to be added as a contributor to edit the wiki. But don't worry! Just email any [Mailing List](#) or grab us on [IRC](#) and let us know your user name.

## Introduction to CouchDB Documentation

Contributing to the CouchDB Documentation is one of the most valuable things you can do - every bit helps. This document will also guide you on how you can contribute.

CouchDB Documentation is bit scattered, however, authoritative documentation is now in the git repository and available for browsing on-line. Unfortunately, some things are still available only on this wiki or in mailing lists/commit messages/github gists/etc, which ideally should have been in the official documentation; contributions are welcome and greatly appreciated.

- The [Official Documentation](#)
- The [CouchDB Wiki](#) (you're here)
- The [Definitive Guide](#), which you can edit/fork/contribute to.
- The documentation [repository](#) is separated out from CouchDB [sourcecode](#). Go ahead and fork the documentation repository and start contributing.
- A [Search Read The Docs](#) page and packaged [document downloads](#) are generated using the awesome Read The Docs site.

We welcome your contributions to any of these - if you need help, just ask in IRC, or on the dev email list. Thanks!!

## The Quick Start Guide

- Read the sphinx tutorials and style guide (pointers below).
- Clone the [couchdb repository](#)
- Set up the python tool-chain as described below.
- Edit .rst files as needed in share/doc
- Build using this command in share/doc, replacing version, release, year as required, to generate HTML pages(select a command based on per page HTML or single HTML),

### Command - Separate HTML per page

```
sphinx-build -a -E -W -n -b html \  
-D version='$ver' \  
-D release='$ver-git-sha' \  
-D project='Apache CouchDB' \  
-D copyright='$year, Apache Software Foundation' \  
-d $TMPDIR/couchdocs/ \  
src/ \  
`pwd`/manual
```

### Command - Single HTML for whole docs

```
sphinx-build -a -E -W -n -b singlehtml \  
-D version='$ver' \  
-D release='$ver-git-sha' \  
-D project='Apache CouchDB' \  
-D copyright='$year, Apache Software Foundation' \  
-d $TMPDIR/couchdocs/ \  
src/ \  
`pwd`/manual
```

- Double check all the changes, make a patch & attach to a JIRA ticket, or send a pull request via github.
- Whew! Congratulations, you are contributing now.

## Restructured Text(.rst) and Sphinx Tutorials

The official source contains .rst (Restructured Text) files used for API and release notes. These will be published with every CouchDB release, and made available [online](#) eventually. There are plenty of helpful tutorials out there:

- The [Sphinx site](#) has a great introduction
- The official [quick start guide](#)
- A [cheatsheet](#) for both Sphinx and rst

- If you want to regenerate the documentation into HTML, the [Sphinx](#) guide may be of interest.

## Adding Version Added/Changed or Deprecated Information

Sphinx supports [paragraph-level annotations for tracking when functionality was modified](#). By adding this information directly to the .rst documentation during committing any functionality, future NEWS and CHANGES updates can be directly extracted from Sphinx.

## Editing Documentation via GitHub

GitHub supports displaying .rst files with correct formatting directly in their web UI. This makes quick updates very easy, using the mirrored ASF [CouchDB documentation git repository](#).

## Get your own Fork

Before editing the documentation, get your own editable copy of the repository:

1. Fork: as you will not have write permission to the ASF CouchDB-documentation git mirror, you'll need to make your own fork, or copy. Just click on the fork button, and you're done.
2. If you are stuck with identifying the correct branch etc, drop an email to [dev@couchdb.apache.org](mailto:dev@couchdb.apache.org) or catch folks on IRC (#couchdb-dev).

## Driving GitHub UI

Now that you've got a suitable branch selected:

- Browse to the required rst file.
- Identify the file you want to edit/alter and click on edit button on github menu.
- Switch to raw view and start editing! Note that you can switch back between source and edit view without losing your content.
- Once you're happy with the changes, add a suitable brief commit message and follow the contributor guide to send your patch for developer code review.
- Once done, the committers will pick up your documentation changes, and your Karma will be increased!

## Generating Docs

Documentation should be freshly generated for each pull request, to ensure the formatting and linking in particular is still correct, prior to commit or merge.

## Setting up Python and Sphinx for the first time

Assuming you have python 2.7 or a close relative, the following instructions will set up a python virtual environment containing the required tools:

### Setting up Python and Sphinx-build

```
$-> cd $MY_COUCH_SOURCE_REPO
$-> virtualenv env
$-> source env/bin/activate
$-> which python pip
$-> pip install sphinx docutils pygments
```

In future, you'll only need to run `source $MY_COUCH_SOURCE_REPO/env/bin/activate` in each shell / prompt you run in.

If you want to build the PDF, info files, etc, you will need a working TeX installation. Mac OS X users can use [MacTeX](#) for this.

*Can someone explain why a virtual environment is needed? Why can't, or shouldn't, I just install these packages normally? ~ Noah S.*

*Virtualenv is good common practice to not mix system wide installed Python packages with project required ones. While in most cases you will not face any problems with installing sphinx and all his dependencies system wide, there are some edge cases that could make things broken:*

*- Your system package manager doesn't have sphinx or some of his depended package available to install. Manual system wide installation is like make && make install - one more step closer to unmanaged system;*

*- You already has sphinx installed, but of version that has breaking changes against required one, and you're needed this different version - updating could break something for you;*

*Since Python 3.3 release there wouldn't be any requirement for virtualenv since it goes to be in [stdlib](#) and sphinx is ready for Python 3.x. ~ Alexander Sh.*

## Generation via ReadTheDocs

To regenerate [the online docs](#), push your changes to the /docs branch, and then POST to a build-specific URL `curl -vX POST http://readthedocs.org/build/4251`. You can track the [build progress](#) and also grab the various built [formats](#).

The <https://github.com/dch/couchdb/tree/docs> repo has a github post-commit hook already set up, so for testing you can use that repo, and view your [results](#) directly, or see the [error log](#). You will be added as a contributor to it on your first pull request.

## ToDoS

Check out the [documentation component](#) in JIRA for our current TODO list.

@ @ *Replace this with a shared filter, once you have permissions.* ~ Noah S.

## DocBook

The original Couchbase-provided API documentation came in DocBook form. XML is feared and loathed by the Couch Community, who prefer to Relax with JSON. Much wailing and gnashing of teeth was done, and eventually the foul beast was slain by a thousand IRC cuts and ten thousand reply-to-alls. Those who have no fear may find the following lightweight information of use. Others may use it to scare small children and remind teenagers that the world was much harder in days gone by.

The DocBook source was exported using [pandoc](#) which is available as a binary on most common platforms, as follows:

### Using pandoc

```
for xml in *.xml; \
do echo ${xml}; \
pandoc --smart --preserve-tabs --normalize --reference-links --chapters \
--number-sections --standalone --from docbook --to rst \
--output ../dist/${xml} ${xml};
done
```