

# The rebuild process - Recovering from my Macbook drive failure

☒ You never see it coming, yet there it is staring you in the face. A perfectly frozen screen with no animation, no mouse movement, no keyboard response. The first thing you think is that it is just some errant process that ate up all of the resources and went off to chew on it in a tree somewhere.

I took the first logical step. Hard power down and boot back up. We've all done it a number of times before. This time was different though and I immediately had that sickening feeling. The Apple logo screen and it's little cutesy spinner were on the screen for around 5 minutes. My fears were realized soon after as the OS loaded after another 5 minutes only to hear the constant whirring of the hard drive and glacially slow response to mouse clicks in the UI.

Within 15 minutes the system had ground to a halt and multiple attempts to revive the poor overworked hard drive proved to be fruitless. I have been using Time Machine backups for

There are some points that I know will be troublesome:

- MySQL and Rails on Snow Leopard - 64 bit configuration was a pain last time, so it won't be any different this time
- Recovering and rebuilding iTunes library

Conversely, there are some things made very easy because of simple things:

- Rails and Ruby code - Not a problem because I use Unfuddle to keep all my code in a repository on the web
- Pictures - I learned from other's mistakes. I have a Time Machine backup from 6 weeks ago
- Documents and other tools - Thank you Google! I store a lot of items in Gmail and Google Docs for ease of use and ease of access. This has a doubling effect of saving me from this type of situation

Here is my build and how it all went together. It is more for my own personal notes, but if you ever wanted to setup a Macbook as a Ruby on Rails development environment as well as a pretty solid Garage Band recording studio, this is how you can do it.

1. Install Snow Leopard - my machine came with Leopard so I only have the core upgrade/install DVD
2. Install Xcode from the Snow Leopard DVD
3. Install Applications (iLife bundle) - I have to use the Leopard edition and update after the fact using Software Update
4. Install Google Chrome
5. Update all code to latest and greatest using Software Update
6. Install TextMate and configure as the default editor for the shell and for SVN (don't judge)

```
export EDITOR='mate -w'  
export SVN_EDITOR='mate -w'
```

7. Run the Rubygems updates

```
sudo gem update -system
```

8. Setup my code folder and get my active project code from Unfuddle

9. Go into each folder and run Bundle Install to get all the necessary components (Dear Rails 3.x, Thank you!)

10. Install MySQL (file: `mysql-5.1.53-osx10.6-x86_64.dmg`) including the package, Pref Pane and Startup Item

11. Configure MySQL (follow the MySQL section here:

<http://weblog.rubyonrails.org/2009/8/30/upgrading-to-snow-leopard> )

Additionally I set the path include the MySQL bin folder by adding this line to my `~/.bash_profile` file

```
export PATH=/usr/local/mysql/bin:$PATH
```

12. Install and update MacPorts

13. Install uTorrent

14. Install my Line6 software. This is easy because I install a product called Line6 Monkey, plug in the TonePort UX2 and download and install the related software using Monkey

And that, as they say, is that. One fully functional Ruby on Rails environment and one fully operational recording studio in a neat and tiny little aluminum case.