Getting Started (Linux)

These are the instructions to get Avalon up and running on a **clean RHEL machine**. Your mileage may vary for other Linux distributions.

Install Avalon

Install developer tools

Install packages required for dependencies to build properly.

```
yum install gcc-c++ zlib-devel readline-devel openssl-devel
java-1.7.0-openjdk
```

Install **RVM** to manage Ruby (http://rvm.io/) If it doesn't download right away it might ask you to download signatures.

```
curl -L https://get.rvm.io | bash -s stable --ruby
```

Reload shell environment using the instructions given in rvm's installation output.

Make sure ruby installed correctly

```
rvm list
```

Update **rubygem**

```
gem update --system
```

Install **bundler** for managing dependencies

```
gem install bundler
```

Install & setup Git

Install **Git**

```
yum install git
```

> The next step is for committers.

Create an account on github and set up SSH keys. For more help see the detailed instructions

```
cd ~/.ssh
ssh-keygen -t rsa -C "your_email@youremail.com"
cat id_rsa.pub #Add the output to your github account's list of keys
```

Install Matterhorn dependencies

You'll need a couple Matterhorn dependencies installed: ffmpeg and MediaInfo.

> This may take a **LONG** time to run so have something else to keep you busy!

FFMPEG

```
yum install ffmpeg
```

If installing ffmpeg with yum doesn't work, or you have dependency problems, you can try installing it from source.

```
yum install libfaac-devel
git clone git://source.ffmpeg.org/ffmpeg.git ffmpeg
cd ffmpeg
./configure --enable-gpl --enable-libfaac --enable-libmp3lame
--enable-libopencore-amrnb \
  --enable-libopencore-amrwb --enable-librtmp --enable-libtheora
--enable-libvorbis \
  --enable-libvpx --enable-libx264 --enable-nonfree --enable-version3
--enable-x11grab
make
```

That configure statement might need to change. It worked for me, but it seems puppet has more options enabled.

If you're still having trouble, check felix/log/opencast.log and see if there are any exceptions. Also, go to felix/etc/config.properties and make sure matterhorn is pointing to the correct ffmpeg.

Mediainfo

Go to MediaInfo's website to install MediaInfo. #Someone should fill more out here.

Setup Avalon

> The next step may be slightly different depending on your operating system if you are not using RHEL6. When running bundle install you may need to set: **QMAKE=/usr/bin/qmake-qt4**

Add the required dependencies so that all gems will install properly.

```
yum install ruby-devel libxml2-devel libxslt libxslt-devel libcurl-devel
sqlite-devel
```

Go to your workspace. This command will download avalon source from github. Run **bundle install** to install all of the necessary dependency gems.

```
git clone git@github.com:avalonmediasystem/avalon.git
cd avalon
bundle install --gemfile=Gemfile.first
QMAKE=/usr/bin/qmake-qt4 bundle install
```

Prepare the databases

```
rake db:migrate
rake db:test:prepare
```

Install the **hydra-jetty**, **avalon-felix**, and **avalon-red5** submodules. The hydra-jetty submodule contains Fedora and Solr and can be managed via jettywrapper rake tasks. The avalon-felix submodule is a custom instance of Matterhorn. avalon-red5 is a custom instance of Red5 with Matterhorn integration. These two servers can also be controlled with rake tasks. After installing them, force them to track the master branch for their respective repositories.

```
git submodule init
git submodule update
cd jetty/
git fetch --tags
git checkout 4.x
cd ..
cd felix/
git fetch --tags
git checkout 1.4.x
cd ..
```

Config avalon based upon the Configuration Files page. Note that Matterhorn config files can be put in the felix_conf directory and copied over with the felix:config rake task.

To configure Devise and Blacklight Secret Key, in your avalon directory run the following command:

```
rake secret
```

Copy the output, then create and open the file config/secrets.yml. Like the other yml files, you can have separate sections for development, test and production. Here is a template:

```
development:
    secret_key_base: <Paste the hash>
```

Start all services

```
rake jetty:config
rake felix:config
rake avalon:services:start
```

To make the startup of avalon easier after a git pull, you can add these functions to your .bashrc script.

Test Avalon

From your avalon installation execute

```
rails server
```

Navigate to http://localhost:3000