

# Upgrading Avalon 6.4 to Avalon 6.5

See the [Avalon 6.5 Release Notes](#) for more information.

## Requirements

Avalon 6.5 and above requires Ruby 2.4 or newer. If you're using RVM, simply run

```
rvm list # Get current ruby version, eg 2.2.1
rvm upgrade 2.2.1 2.4.6 # Installs Ruby 2.4.6 and migrate gemsets, wrappers, aliases and environment files
```

## Upgrade Steps

### For Manual Installations

1. Download and deploy new code from the [avalon repository](#).
2. Upgrade to bundler 2, update gems, and DB

```
gem install bundler
bundle install
RAILS_ENV=production bundle exec rake db:migrate
```

3. [Install yarn](#) and update JS dependencies

```
curl --silent --location https://dl.yarnpkg.com/rpm/yarn.repo | sudo tee /etc/yum.repos.d/yarn.repo
sudo yum install nodejs # Tested with Node.js version 10
sudo yum install yarn
yarn install # in Avalon root dir
```

4. Restart passenger (if using)

```
passenger-config restart-app
```

5. Upgrade Redis and Restart Resque (background jobs processor)

```
sudo yum upgrade redis # Version 3.0+
sudo service redis restart
pkill -f resque
RAILS_ENV=production BACKGROUND=yes bundle exec rake resque:scheduler
RAILS_ENV=production BACKGROUND=yes QUEUE=* bundle exec rake resque:work
```

6. Upgrade ffmpeg (version 4+). You can either download a prebuilt binary

```
mkdir -p /tmp/ffmpeg && cd /tmp/ffmpeg
curl https://johnvansickle.com/ffmpeg/releases/ffmpeg-release-amd64-static.tar.xz | tar xJ
sudo cp `find . -type f -executable` /usr/bin
```

Or use <https://github.com/jrottenberg/ffmpeg> to build ffmpeg and then copy it to your machine. Be sure to update the ffmpeg path in `config/settings.yml` to match where it was installed.

7. Disable Matterhorn HLS prechunking (if not already) by commenting out the HLS directory in [etc/config.properties](#).

```
<matterhorn_root>/etc/config.properties
```

```
# org.opencastproject.hls.directory=../public
```

Then restart Matterhorn

```
sudo service matterhorn restart
```

This takes the HLS distribution module offline causing Matterhorn to appear in an error state on the about page. This does not indicate a real problem but you can optionally choose to fix it by manually deleting the module from the Matterhorn database.

```
delete from mh_service_registration where service_type = 'org.opencastproject.distribution.hls';
```

Note that if you do this you should also remove the HLS distribution steps from the avalon workflows in `/usr/local/matterhorn/etc/workflows`.

8. Update encoding profiles

```
wget -q https://raw.githubusercontent.com/avalonmediasystem/avalon-felix/e36aad51ee4a62795504f7a9e176e5f0ca7644b8/etc/encoding/avalon.properties -O /usr/local/matterhorn/etc/encoding/avalon.properties
```



More information about changes to the default encoding profile can be found in the [avalon-felix repository](#).

9. [Setup nginx streaming](#) if you had been previously depending on the rails server to serve the HLS content.

10. Run waveform backfill script

This script will kick off many background jobs to generate waveform data for existing content for use with the new structural metadata editor. (New content will have waveforms generated as part of the ingest process.) If you have a lot of files ingested into Avalon the background jobs may take a very long time to run. See [Waveform Backfill Script](#) for more information.

```
RAILS_ENV=production bundle exec rails r script/waveform_backfill.rb
```

11. Turn on the timeliner in settings.yml:

**config/settings.yml**

```
timeliner:  
  timeliner_url: https://<hostname>/timeliner
```

## For Docker Installations

Update new code from [avalon-docker](#)

```
git checkout master  
git pull
```

Rebuild and replace Avalon container.

```
docker-compose build --no-cache avalon  
docker-compose stop avalon  
docker-compose up avalon
```

