

Upgrading Avalon 3.2 to Avalon 3.3

Preparing for the Upgrade

Before doing anything to prepare for the upgrade, backup your Avalon 3.2 installation.

Workflow updates to support multiple derivative batch ingest

Part of the changes made to get multiple derivatives ingest to work was updating the Matterhorn workflows for skip transcoding. Instructions for updating these workflows are found below.

Default Protocol configuration

To help Avalon automatically generate URLs for use in system emails and other places, it is now possible to configure a default protocol (http or https). An example configuration can be found in `config/avalon.yml.example` on [github](#).

Validating Media Object metadata

Avalon release 3.3 includes a new requirement that metadata values Creation Date, Copyright Date and Date Issued conform to the [Extended Date Time Format](#). Dates that do not conform will not be indexed and therefore will not be searchable. To assist in identifying existing records affected by this change, a tool has been implemented to identify those records. In addition to identifying invalid dates it identifies all records with ANY invalid metadata. Any records identified as having validation errors should be fixed so that they will be indexed. After the upgrade, run the tool by executing the following command:

```
$ su - avalon -c "cd /var/www/avalon/current; RAILS_ENV='production' bundle exec rake avalon:validate"
```

Matterhorn disk cleanup

In previous Avalon versions, there was a potential for Matterhorn disk space to fill with workflow artifacts. The default error handler workflow updated as above reduces this potential significantly. And a tool is provided in Release 3.3 that helps clean up Matterhorn workspace further. More can be learned [here](#)

See [What's New in Avalon 3.3](#) for more details.

Instructions

The following instructions will work for any Avalon 3.2 installation that used Puppet as its source. This includes the [portable VM Image Installation](#), the [Vagrant Virtual Machine Installation](#), and the [Manual Installation of Avalon Using Puppet](#).

Upgrading is fairly simple, and must be run as the root user from a bash prompt on the Avalon system. The process for this depends on which kind of installation you have.

Portable VM image:

- Open a terminal window from within the VM console, and type `su root` to become the root user. This will require the root password you provided the first time the system was booted.
- Vagrant virtual machine:
From within the original installer directory (from step 4 of the [Vagrant Virtual Machine Image Installation Instructions](#)), type `vagrant ssh` to log in as the vagrant user, then type `sudo -i` to become the root user.
- Manual Puppet installed machine:
Log in and become root as you did in steps 1 and 2 of the [Manual Installation of Avalon Using Puppet](#) instructions.

Once you have a root prompt, proceed with the upgrade as follows:

Upgrade Avalon

```
# 1. if starting from Portable VM image, download the installer package, unpack, and link it
$ wget http://www.avalonmediasystem.org/downloads/avalon-installer-flat.tar.gz -O flat.tar.gz
$ tar xzf flat.tar.gz
$ export INSTALL_DIR=`pwd`
$ ln -s $INSTALL_DIR/avalon-installer-flat/files /etc/puppet/avalon_files

# 2. navigate to the installer
$ cd avalon-installer-flat

# 3. remove the Avalon deploy tag
$ rm -f /var/www/avalon/bare-deploy
$ rm -rf /opt/staging/avalon/avalon-bare-deploy.tar.gz /opt/staging/avalon/avalon-bare-deploy

# 4. apply the puppet manifest
$ puppet apply --fileserversconfig=fileserver.conf --modulepath=modules \
  --hiera_config=hiera/hiera.yaml --templatedir=templates manifests/init.pp

# 5. reindex
$ su - avalon -c "cd /var/www/avalon/current; RAILS_ENV='production' bundle exec rake avalon:reindex"
```

Upgrade Matterhorn

```
# 1. Switch to matterhorn user
$ su - matterhorn
$ cd /usr/local/matterhorn

# 2. Install updated workflows
$ cd etc/workflows
$ wget --no-check-certificate https://raw.githubusercontent.com/avalonmediasystem/avalon-felix/1.4.x/etc/workflows/avalon-skip-transcoding.xml -O avalon-skip-transcoding.xml
$ wget --no-check-certificate https://raw.githubusercontent.com/avalonmediasystem/avalon-felix/1.4.x/etc/workflows/avalon-skip-transcoding-audio.xml -O avalon-skip-transcoding-audio.xml
$ wget --no-check-certificate https://raw.githubusercontent.com/avalonmediasystem/avalon-felix/1.4.x/etc/workflows/default-error-handler.xml -O default-error-handler.xml

# 3. Restart Matterhorn
$ exit
$ service matterhorn stop
$ service matterhorn start
```

If you didn't load the sample content before, [try it out](#) to test your avalon 3.3 upgrade.