

Configuration Files



This documentation is for Release 6.3.x and above.
For older releases:

- Release 6.2.x, see [v.51](#).
- Release 5.x, see [v.47](#).
- Release 4.0, see [v.40](#).
- Release 3.3, see [v.36](#).
- Release 3.2, see [v.32](#).
- Release 3.0 and 3.1 version of this page, see [v.31](#).
- Release 2 version of this page, see [v.21](#).
- Release 1 version of this page, see [v.9](#).



Note

Changes to configuration files often require restarting Avalon to take affect. They also require Resque (the background jobs processor) to be restarted.

- [Avalon Web Application](#)
 - [config/settings.yml](#)
 - [config/database.yml](#)
 - [config/fedora.yml](#)
 - [config/intercom.yml](#)
 - [config/matterhorn.yml](#)
 - [config/solr.yml](#)
 - [config/role_map_<environment>.yml](#)
 - [config/controlled_vocabulary.yml](#)
 - [config/secrets.yml](#)
- [Opencast Matterhorn](#)
 - [etc/config.properties](#)
 - [etc/encoding/avalon.properties](#) (ADVANCED)
 - [etc/workflows/avalon-video.xml](#) (ADVANCED)
 - [etc/workflows/avalon-audio.xml](#) (ADVANCED)

Avalon Web Application

config /settings. yml	
Contents	Environment-specific Avalon configuration options
Format	YAML
Example	<pre>production: name: 'Avalon' # used in page titles domain: host: localhost port: 3000 protocol: http bib_retriever: protocol: sru url: http://zgate.library.example.edu:9000/catdb query: rec.id=%{bib_id} controlled_vocabulary: path: 'config/controlled_vocabulary.yml' dropbox: path: '/usr/local/masterfiles/dropbox/' upload_uri: 'sftp://localhost/srv/avalon/dropbox'</pre>

```
email:
  comments: 'digitalcollections@northwestern.edu'
  notification: 'digitalcollections@northwestern.edu'
  support: 'digitalcollections@northwestern.edu'
  mailer:
    smtp:
      address: 'mail-relay.iu.edu'
      port: 587
      enable_starttls_auto: false

ffmpeg:
  path: '/usr/bin/ffmpeg'

fedora:
  namespace: 'numedia'

groups:
  system_groups: ['administrator', 'group_manager', 'manager']

matterhorn:
  root: 'http://avalon-mhorn-dev.library.northwestern.edu:8080/'
  media_path: '/usr/local/masterfiles'
mediainfo:
  path: '/usr/bin/mediainfo'
streaming:
  server: :generic # or :adobe
  content_path: /var/avalon/rtmp_streams
  rtmp_base: rtmp://localhost/avalon
  http_base: http://localhost:3000/streams
  stream_token_ttl: 20 #minutes
master_file_management:
  strategy: 'move'
  path: '/mnt/diskarray/avalon-archive'
flash_message:
  type: ['success', 'notice', 'error', 'alert']
  message: '<p>A test message.</p>'

solr:
  configset: avalon
  configset_source_path: <%= File.join(Rails.root, 'solr', 'config') %>
  collection_options:
  async:
  auto_add_replicas:
  collection:
  config_name: avalon
  create_node_set:
  max_shards_per_node:
  num_shards: 1
  replication_factor:
  router:
  name:
  field:
  rule:
  shards:
  snitch:
zookeeper:
  connection_str: "localhost:9983/configs"

redis:
host: localhost
port: 6379

controlled_vocabulary:
  path: config/controlled_vocabulary.yml
  auth:
  configuration:
    # List of available Authentication Providers and associated configurations.
    # The name and logo settings are used when displaying multiple auth providers to choose from at
    login time.
    # provider and params are passed directly to config.omniauth as described in the Devise OmniAuth
    Overview.

  - :name: Avalon Test Auth
    :provider: :identity
    :params:
    :fields:
    - :email
  - :name: Avalon Lti OAuth
    :provider: :lti
    :hidden: true
    :params:
    :oauth_credentials:
      somekey: somevalue
```

	<pre> - :name: My LDAP Server :logo: auth_provider_logo.png :provider: :ldap :params: :method: :ssl :host: ldap.example.edu :port: 636 :bind_dn: cn=avalon,ou=service,dc=example,dc=edu :password: avalon_bind_password </pre>
Notes	<p>Managing Master Files explains available strategies for how avalon can handle master files after they have been processed.</p> <p>Bibliographic Import Configuration explains how to configure Avalon to import metadata from an external source such as an OPAC.</p> <p>Config overwrites should be placed in <code>config/settings/<environment>.local.yml</code></p>
config/database.yml	
Purpose	Environment-specific database connection information for Rails.
Format	YAML
Example	<pre> production: adapter: mysql2 host: localhost database: rails username: rails password: rails pool: 5 timeout: 5000 </pre>
Notes	See Configuring a Database in the Ruby on Rails Getting Started Guide for more options. Changing the database adapter will require updating Gemfile to require the correct adapter. Here are the instructions for installing the mysql adapter called mysql2.
config/fedora.yml	
Contents	Environment-specific Fedora repository connection information for Hydra
Format	YAML
Example	<pre> production: user: fedoraAdmin password: fedoraAdmin url: http://127.0.0.1:8984/fedora4/rest base_path: /prod </pre>
config/intercom.yml	
Contents	Configuration for optional Intercom feature. Allows this instance on Avalon to push contents to another instance of Avalon.
Format	YAML
Example	<pre> intercom: default: url: https://some.avalon.com/ api_token: a_valid_token import_bib_record: true publish: false push_label: text for the 'push' button </pre>
Notes	<p>Intercom utilizes the Ingest API and requires an API token to connect. This token is generated on the target Avalon with <code>`bundle exec rake avalon:token:generate username=archivist email=archivist1@example.com`</code></p> <p>The streaming files will remain in place. Both instances of Avalon must be able to stream from that location.</p>

config/matterhorn.yml	
Contents	Environment-specific Matterhorn connection information for Avalon's Matterhorn adapter
Format	YAML
Example	<pre>production: url: http://matterhorn_system_account:CHANGE_ME@127.0.0.1:8080/</pre>
Notes	This file specifies the system account used to send media to and fetch updates from Matterhorn, as opposed to the public Matterhorn root URL provided in settings.yml, which is used solely for client-side interactions.

config/solr.yml	
Purpose	Environment-specific Solr connection information for Hydra
Format	YAML
Example	<pre>production: url: http://localhost:8983/solr/avalon</pre>

config/role_map_<environment>.yml	
Contents	Role groups and associated members to initialize the Hydra access controls database with
Format	YAML
Example	<pre>collection_manager: - archivist1@example.edu - archivist2@example.edu group_manager: - donor1@example.com manager: - researcher1@example.edu administrator: - user1@example.edu - user2@otherinstitution.edu</pre>
Notes	Example environments: development, production, test. So the filename would look like role_map_production.yml for the production environment.

config/controlled_vocabulary.yml

See [Managing Controlled Vocabularies](#).

config/secrets.yml

Rails 4+ uses a secrets.yml file to store a secret key used for signed cookies as well as other things. See http://edgeguides.rubyonrails.org/upgrading_ruby_on_rails.html#config-secrets-yml for more information.

Opencast Matterhorn

etc/config.properties	
Contents	Configuration info for Matterhorn
Format	key=value
Relevant Keys	<pre>org.opencastproject.server.url: Public URL of the Matterhorn server org.opencastproject.streaming.url: Base URL of the streaming application org.opencastproject.streaming.directory: Directory where streaming derivatives will be written org.opencastproject.composer.ffmpegpath: Location of the ffmpeg binary org.opencastproject.inspection.mediainfo.path: Location of the mediainfo binary org.avalonmediasystem.avalon.url: Base URL of the Avalon server</pre>
Notes	Other keys and their values are documented in comments within the file

etc/encoding/avalon.properties (ADVANCED)

Contents	Specifics of Avalon transcoding operations
Format	key=value
Notes	Allows specific tweaking of ffmpeg parameters for different content types and quality levels

etc/workflows/avalon-video.xml (ADVANCED)**etc/workflows/avalon-audio.xml (ADVANCED)**

Contents	Avalon-related transcoding and processing workflows
Format	XML
Notes	See the Matterhorn Workflow Documentation (registration required). If using Adobe Media Server be sure to disable the distribute-hls step in both files.