

# Mrss Generator



Created By: Andy Rose  
For: UWTV

# Overview

The purpose of this project was to generate mrss code that would be ingested by an online database management system (DBMS) for UWTV. The code is generated from different media components and allows the DBMS to link to the media on UW's servers and stream it on the web.

The reason for creating this project is to avoid writing long code files for every media object. By automating the process, this program will save the company time and resources.

# System

- The Platform

- Links to media files that are stored on local servers for streaming content
- Stores thumbnail image files that are used for each media object
- "Ingests" feeds that are inserted into a "Watch" folder
- Holds the codec information used to read the file
- Provides the embedded viewing window to watch the media

- MRSS

- RSS feeds for media
- Provides a link to the media on the local server
- Declares the properties of inserted media

# Required Information

Before a media object can be ingested, the user must provide certain information. This includes the type of ingest (thumbnails or media), media object IDs, server IDs, and Transform IDs.

Media Object IDs point to the specific media object in the system that the files will belong to.

- Media objects must be created before adding files

Server IDs determine what server on the platform will be used to host a specific media file or thumbnail

Transform IDs are linked to the codec used to read a specific media file

# Program Requirements

- Generate mrss script for both thumbnails and media to be ingested by the platform
- Store all of the types of media
- Gather all necessary information from the user
- Allow the user to locate files to be ingested
- Store, alter, and use all forms of media (Media Assets)
  - Must hold all properties relevant to asset
- Change and set the server and transform IDs for each media asset
- Use the system quickly and efficiently in coherence with the platform

# Program Layout

## User Settings

Settings in this program have both functional, and preference qualities. They exist to make using the program more efficient. They can all be changed by the user through the program.

- Functional
  - Location of the "Watch" and "Backup" folders
  - Dimensions for the large and small thumbnails
  - Server ID for thumbnails
- Preference
  - Selecting the starting tab (thumbnail or media) which is most commonly used
  - Keeping the window on top of others
    - For working with the platform
  - Location of the thumbnail and media folders
    - Avoids having to locate them for each entry

Current Settings



Personal Settings

Default Initial Tab: Thumbs

Small thumb size: {Width=200, Height=150}

Large thumb size: {Width=640, Height=480}

Default Folders

Watched: Desktop

Backup: C:\Program Files (x86)\UWTV\Mrss Generator\BackupFiles

Media: Desktop

Thumbnails: Desktop

Default Server ID's

Thumbnails: 15

High Quality Stream: 15

Low Quality Stream: 15

Mezzanine: 15

Podcast (Audio): 155

Podcast (Video): 155

Default Transform ID's

High Quality Stream: 15

Low Quality Stream: 18

Mezzanine: 15

Podcast (Audio): 15

Podcast (Video): 15

OK

# Program Layout

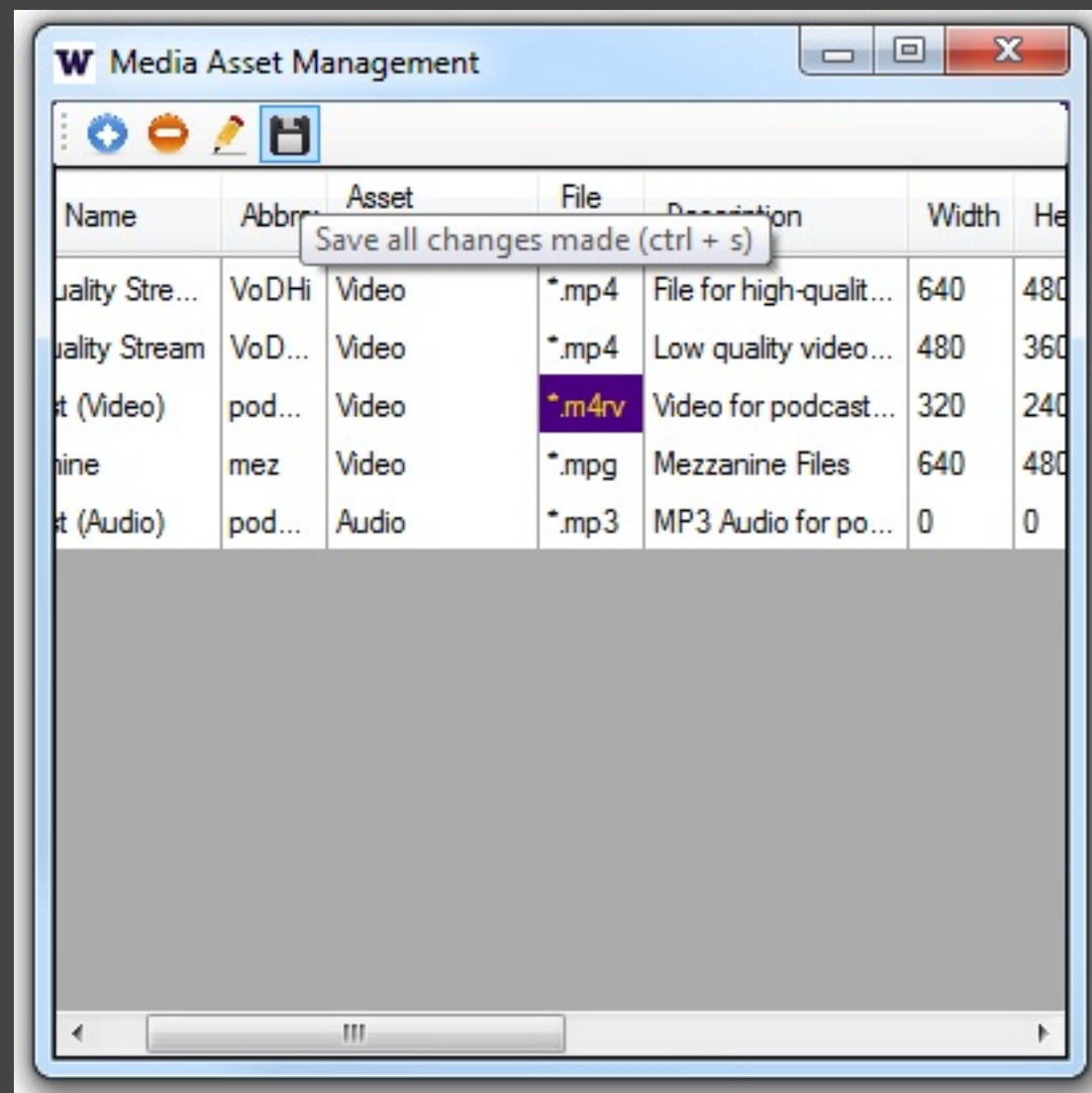
## Media Assets

Assets can be added, altered and deleted by the user. Information on assets is stored in a database with the program.

Attributes:

- Name and abbreviated name for later selection
- Type - Video, Audio, or Image
- File Extension - Used as filter when finding files
- Description
- Width and Height - For Video and Image Assets
- Bitrate - For Video and Audio Assets
- Frame Rate - For Video Assets
- Release - If the media will be streaming content
- Analyze - If the Platform should scan metadata on ingest
- Ingest Method - Copy or Move the file
- Server and transform ID
- Defined - If the asset is pre-defined by the platform





# Program Layout

## File Menu

The provided file menu is an intuitive way to navigate through the system. Shortcut keys are also defined for many of the actions.

Functionality for the menu is as follows:

- Change any of the personal settings
  - View a list of the current settings
- Change the server or transform IDs for any defined asset
  - Assets are inserted into the menu at run-time
- Access edit or view window for all assets
- Access the help window
- Open the backup folder
- Clear all fields with entered information
- Minimize the program to show only the tool strip
  - For getting it out of the way when working with platform
- Closing the program

# Program Layout

## Help Page

A custom help page is available to users with inquiries on the capabilities of the program and how to use it. A navigation pane on the left allows the user to select the topic for which they need help. When an item is clicked, the information appears in the info box and pictures are displayed below. Multiple pictures are available for some steps and can be changed by the user. All of the help information is stored in the Help table of the database. The pictures are located in the application folder in a sub-folder called Help.

- 1 Getting Started
- 2 Required Settings
- 3 Program Settings
- 4 Asset Management
- 5 Using the Program

## Getting Started

This program was built to use with MediaAMP (The Platform) in order to easily and consistently create mrss (media rss) script for media and thumbnails to be "ingested" into the system. The MediaAMP is an online database management system. It does not actually hold the media files that are streamed to the online user. The streamed files are held on a local server and linked to by the MediaAMP. This program writes a script that is delivered to the MediaAMP, giving it instructions on where to access the necessary files and specific rules for each form of media. In order to insert information on media, you must have servers set up to host the links, transform codecs to read the different types of media, as well as know and have access to the location of the actual media files. For inquiries on learning how to use this program in coherence with the mpx suite for MediaAMP, consult the items on the left. (Note: It is important to declare your settings before trying to operate this system).

Version 1.0

Property of University of Washington

Created by: Andy Rose | [andy.rose2588@gmail.com](mailto:andy.rose2588@gmail.com)



# Program Layout

## Inputting Data

- Two tabbed pages for thumbnails and media are available on the mainform
- Both have a text input for a media ID and a button to generate
- Thumbnail tab
  - Requires user to find both small and large thumbnail files
- Media tab
  - User selects an asset from the drop-down menu,
  - then clicks the text box to open a file browser to find the media item
  - Add and remove buttons exist to change number of files
  - Does not require a certain number of files

W Mrss Generator For MediaAMP

File Settings View Help?

Thumbs Ingest Media

Media ID: 1549134561

Mezzanine ▾ //SomePath/SomeFile

Select Media Type ▾

Add Remove Generate

Add one more asset

# Finalization

- The program is all contained in a 3.67MB install file
- Installs onto any Windows system
- Sets the application up in the program files folder
- Offers link to download the package for communicating with database (Access Database)
- Allows user to add shortcuts to desktop and start menu

# Issues Encountered

These are some issues that I encountered and corrected during the development process.

- Fixing small errors with mrss
- Using Visual Studio database adapters
  - Could not get it to function for the necessary tasks
  - Created and accessed exterior database instead
  - Controlling input in the form of a combo box from within the asset table when editing
- Setting default folders dialog will not allow server folders
  - User has to map to a network drive
  - Method created to expand mapped drive network location
- Inputting more than one of the same type of media
  - You cannot add multiples of the same asset type to the platform
  - Adding asset slots will remove the asset selected in all previous slots from the selection to prevent this



# Unexplored Features

These are components that would be nice to have in the program, but were out of the scope of the project

- Server files need appropriate credentials
  - Wanted to give program credentials to provide access to those using the program outside of the IT department
  - Out of the scope for the time being
- Obtaining information directly from the platform
- Creating a web-app interface within the platform suite to perform the same tasks
- Sending/receiving automatic updates so users do not need to perform application-critical changes