# vrecord Testing

## Testing Overview and Protocol

### Overview

Thank you for taking the time to be part of the testing cycle for this round of FADGI-funded development of the vrecord capture software. This current round of development is meant to focus on three major updates to the vrecord software:

- Allow for the creation of MP4 Access Files concurrently with the creation of Preservation files of any format
- The creation of a *Simplified Capture Interface*
- The creation of a package installer

This current round of testing will address the first bullet point, along with other updates that have added functionality surrounding the capture and decoding of Closed Caption data present on line 21 of the video signal.

### Protocol

The testing protocol has been designed to be thorough, without posing a significant time burden for testers. Each tester will create test files to test out the new vrecord features. They will then be asked to analyze the test files according to defined parameters. After analysis, testers will be asked to fill out short forms with their findings and send the resulting files to the Library of Congress Point of Contact (Morgan Morel: mmorel@loc.gov) via LC's MediaShuttle app.

## What To Do Before Testing

### Pre-Testing vrecord Setup

For the purpose of testing, we expect you to use vrecord as you normally would for the most part. You will be sending us a log that contains all your vrecord settings and computer details, so you do not need to record them otherwise.

You will be testing the latest build of vrecord (*version 2024-08-28*). This can be installed using homebrew. Here are instruction to perform a full installation if you are starting from a fresh OS install:

- Install hombrew using the instructions available here: [Homebrew — The Missing Package Manager for macOS (or Linux)](#)
- Run the following two commands in terminal:
    - brew tap amiaopensource/amiaos
    - brew install vrecord

If you already have vrecord installed and simply need to upgrade to the latest versions, run the following command in Terminal:

- `brew update`
- `brew upgrade vrecord`

## Pre-Testing Form

Before testing, [please fill out the Pre-Test form](#) with the requested information. For the sake of consistency in testing please use the same digitization station for all testing. For the storage speed please the Blackmagic Disk Speed Test application. Make sure to set your target drive as the drive you use for capture. You can set the file size to 1GB.

# Concurrent Access Files Testing

## Test Setup

You can leave the majority of your vrecord settings the same, however please make the following adjustments:

- **Decklink Tab:**
    - **Sidecar file options:**
        - **Access MP4:** Checked
- **Config Tab:**
    - **Custom Output Options:** [put in whatever ffmpeg options you would typically use to make a MP4s]
    - *Note: if you don't have a preferred MP4 string, use the following:*
        - -ar 48000 -s 640x480 -vf crop=720:480:0:4,yadif,setdar=4/3

## Running The Test

Create the following files according to the prescribed filename template.

- 5-minute test files
    - 10-bit Uncompressed MOV + Access
        - [TesterName]_5minAccessMOV.mov

- ▪ [TesterName]_5minAccessMOV.mp4
  - ○ MKV/FFV1 + Access
    - ▪ [TesterName]_5minAccessMKV.mkv
    - ▪ [TesterName]_5minAccessMKV.mp4
- Create 60-minute test file
  - ○ *Note: If you don't have time to make a 60 minute file, please just make the longest file you can and name the file appropriately*
  - ○ MKV/FFV1 + Access
    - ▪ [TesterName]_60minAccess.mkv
    - ▪ [TesterName]_60minAccess.mp4

## Test Analysis

When your analysis is complete, please fill out the [accompanying form](#) and send the following files to [mmorel@loc.gov](mailto:mmorel@loc.gov) using [MediaShuttle](#). The Log files are found in the auxiliary files directory.

- [TesterName]_5minAccessMOV.mov
- [TesterName]_5minAccessMOV.mp4
- [TesterName]_5minAccessMOV_capture_options.log
- [TesterName]_5minAccessMOV_vrecord_input.log
- [TesterName]_5minAccessMKV.mkv
- [TesterName]_5minAccessMKV.mp4
- [TesterName]_5minAccessMKV_capture_options.log
- [TesterName]_5minAccessMKV_vrecord_input.log
- [TesterName]_60minAccessMKV.mkv
- [TesterName]_60minAccessMKV.mp4
- [TesterName]_60minAccessMKV_capture_options.log
- [TesterName]_60minAccessMKV_vrecord_input.log

# Closed Caption Testing

## Test Setup

vrecord's closed captioning decoding will happen automatically if it detects caption data on the input video signal. With that in mind, you can capture any format using any options, but we suggest that you use Matroska/FFV1 for space concerns. In order to complete this test, you'll need a tape that has closed caption data on it. Most commercial videos (like movies) are required to have closed captioning, so if you have a movie you use as a test

tape this would be ideal. We are not worried about the quality of the content, but keep in mind that severe flagging at the top of the frame will make it impossible for vrecord to decode the caption info.

You will need to make one adjustment in vrecord to have it create the webVTT files:

- **Config Tab:**
  - **Closed Captioning Tab:**
    - **Make a webVTT:** Checked

## Running The Test

Please create a 5-minute test file. Start the tape somewhere in the middle to ensure that vrecord is receiving captions immediately. Please name the file using this template:

- [TesterName]_5minCaption.mkv

If the captions were properly decoded, there will be a file with the following name in the auxiliary files directory:

- [TesterName]_5minCaption.vtt

## Test Analysis

If created properly, the webVTT will contain text captions. To see if this works you can perform the following:

- Open the MKV file using VLC
- Drag the webVTT file into the VLC window
- At the top bar select *Subtitle -> Subtitle Track -> Track 1*

If the webVTT file was properly made you'll see the closed captioning data displayed on the screen.

When your analysis is complete, please fill out the accompanying form and send the following files to mmorel@loc.gov using MediaShuttle. The Log files are found in the auxiliary files directory.

- [TesterName]_[5minCaption].mkv
- [TesterName]_[5minCaption].vtt
- [TesterName]_[5minCaption]_capture_options.log
- [TesterName]_[5minCaption]_vrecord_input.log

# Links

- Pre-Test Form: https://forms.gle/cNyMNZn5VYbMugWx7
- Concurrent MP4 Post-Test Form: https://forms.gle/tYERDNfdf8AnvG6r5
- Closed Captions Post-Test Form: https://forms.gle/hjHb6xdM9hZTzyD5A
- LC MediaShuttle: https://pso.mediashuttle.com/