

Data Archiving

Organization: [Project SCINI](#) at [Moss Landing Marine Labs](#)
Date: December, 2007
Disclaimer: This material is based on work supported by the National Science Foundation under Grant No. ANT-0619622 (<http://www.nsf.gov>). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Backup Policy

The key to backing up data is to decide on a reasonable procedure and level of caution ahead of time and just stick to it until it becomes a habit; as soon as you decide you can skip something to save time, or it seems that one of the steps is too pessimistic, that's when you drop a hard drive and realize there's no second copy. Most data loss isn't caused by physical damage or degradation, it's by rushing to make copies or getting confused, so take it slow and make sure you know there's a second copy of something before you delete it. The further physically separated backups are from each other the better; both because it's less likely they will both be damaged at the same time, and because it makes it harder to "clean them up" or go through and make "changes". Having two hard disks with the same content on a desk next to each other connected to the same computer or networked computers is almost useless; anything that would corrupt or destroy one would impact the other as well.

As a gross simplification, storing data to DVDs is cheap, compatible, and relatively long lasting (if you take care not to scratch the disks), but is labor intensive and hard to search through (there could be dozens of disks). Hard disks are reasonably cheap, fairly reliable over time, and very quick to create or search through, but choosing a compatible file system that can take large amounts of video data can be tricky [*]. CDs would take an impossibly long time to create (dozens or hundreds would be required). Other options like magnetic tape or Iomega disk drives usually have high overhead (to purchase the drive) and aren't widely compatible.

Once back to a "landline" location, it makes a lot of sense to backup data to the internet, because most webmasters and systems administrators have solid backup plans as part of their routine. Smaller files (less than a gigabyte total?) could go into standard website hosting, while larger video could be uploaded at lower resolution to a video hosting site like [Google Video](#). There are many companies offering data backup services (multiple locations, insurance, fast access, etc) which are probably a lot more reliable than a local solution; many companies are also interested in archiving and hosting scientific data for free (with advertising?), so that might become an option as well.

File Hierarchy

In the 2007 season we found it very useful to come up with a descriptive name for every scuba or ROV dive, and to create folders with that name on every computer involved with the dive. Later when combining files from multiple computers and sources there is no ambiguity about where the files should go, and it's a lot easier to look this data up when it's all together down the road.

The basic name format is the vehicle (or “scuba”), then the date, then the location, all separated by underscores. Every chunk should be short, lowercase, and simple with no spaces; here are some examples:

```
scini_03nov2007_sideshow
scuba_27oct2007_cinder
scini_27oct2007_dwall      ("dwall" refers to Dayton's Wall)
scini_06nov2007_circus
videoray_12dec2007_hutpoint
```

Within each of these base folders there can be subdirectories for “raw_video”, “raw_camcorder”, “stills” (taken from the video), “video” (processed footage), “clips” (short cherry picked video clips), “nav”, etc. By being consistent it is very easy to get all clips or all raw video by searching for directories with those titles and copying them. Spaces and capitalization are handled differently on some operating systems and in particular on the internet, so some headaches are avoided by leaving them out in all the file names.

SCUBA Camcorder Video and Photos

Ripping DV tape to disk

In 2007 we had some difficulty setting up a more appropriate video ripping solution and ended up using Windows Movie Maker on Windows XP to save the tape footage as .avi files with the “DV-AVI” codec. These files would not open in most editing software, but it was the only solution we had at the time.

To rip a section of tape, queue up the starting position on the camcorder, attach it via firewire (through an attachment card if necessary) to the recording computer, attach an external hard disk if desired, run Windows Movie Maker, and select “Capture from Video Device”. Choose a wise and descriptive file name, the full “DV” format (NOT windows media video), and press “Start Recording” on the last screen. The recording will automatically stop if there is a gap between clips, but it could run over into the next segment if there isn’t a time code hop; in other words, if you are capturing the last clip on a tape it will probably stop recording at the right time, otherwise keep an eye on it so it doesn’t run over.

Re-encoding DV footage

The linux program `kino` can read in the camcorder footage downloaded with Windows Movie Maker and export as standard MPEG-2. Open up the file with `kino`, click “Export” on the far right, select the MPEG tab, “3. Generic MPEG 2” as the format, and under Advanced Options delete the Audio Encoding command.

The camcorder footage is usually interlaced (looks like it’s striped with parallel lines); by playing with the de-interlacing options the image quality might be improved.

These MPEG-2 files should be playable and edit-able with most popular programs; I recommend `avidemux` as mentioned below.

SCINI Elphel Video

The streaming video from SCINI, as recorded with the `Elphel0gm` utility (see `elphel-cheatsheet`), needs some post processing to be useful for either frame by frame analysis or release as video. An alternative recording process (such as VLC or recording straight to DVD) might help cut these steps out of the process, with the disadvantages of speed, reliability, and possibly loss of image quality.

mencoder Re-encoding

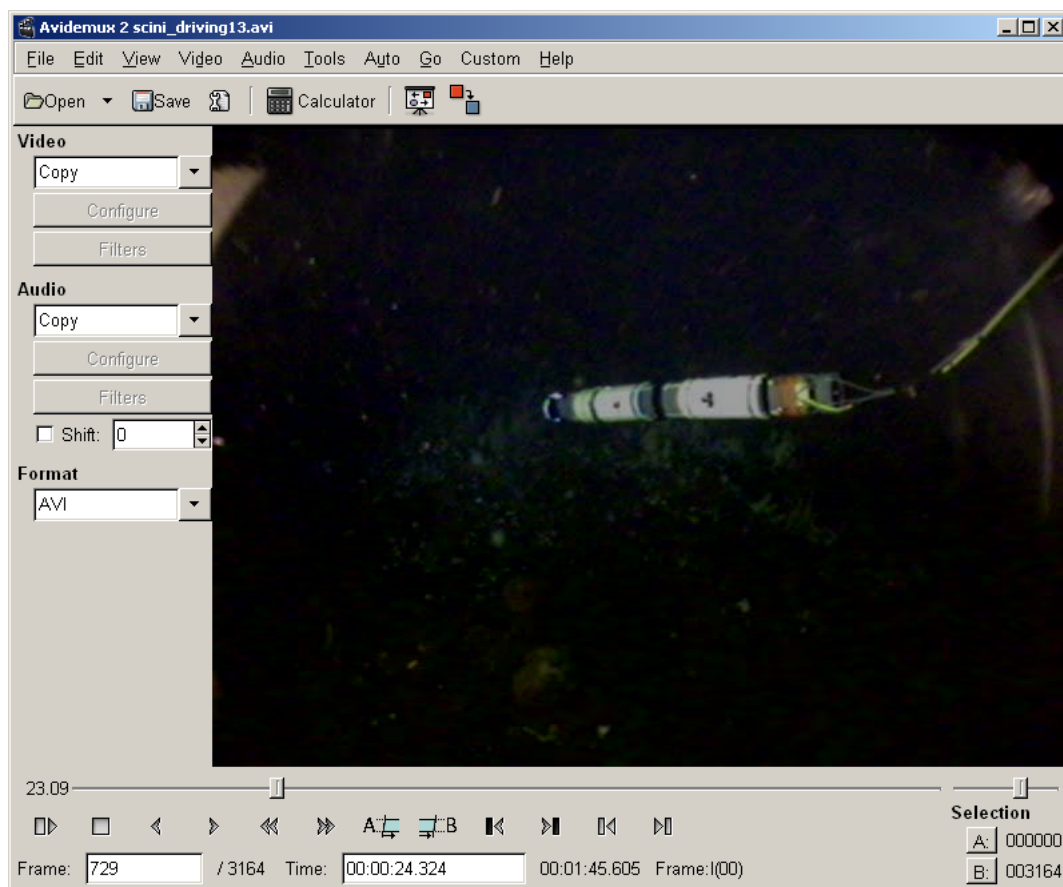
When recording with Elphel10gm, the mjpeg frames are wrapped in the .ogm (ogg media) file format, but apparently not very well; video editing programs like ffmpeg and avidemux throw errors when importing the raw saved streams. Running the files through mencoder in straight frame copying mode seems to solve these problems. Use the following command:

```
$ mencoder vid01 -ovc copy -o vid01.ogm
```

To make this process easier, you can run this one line script from the 'raw_video' folder to re-encode every file::

```
$ mkdir ../video
$ for vid in `ls`; do mencoder $vid -ovc copy -o ../video/$vid.ogm
```

avidemux Editing and Transcoding



Once the video files are in "proper" .ogm format, they can be edited, resized, and exported to different file formats using avidemux, which is available for many operating systems. This is the best time to renormalize the frame rate .

I usually transcode video clips at full size as a simple .avi with the mjpeg codec with 90% quality, and again as .avi files with the Mpeg4 codec (default settings) at 480x360 resolution (using the builtin mplayer resizer). The smaller size is for release on the internet or sharing clips over slow connections.

VideoRay DVR Footage

The footage recorded from the VideoRay with the WinDVR software is often heavily interlaced, and also stored in a strange format. VideoRay (the company) seems interested in recording with VLC in the future which would help out a lot.

For now the footage can be opened in **avidemux**, though it is scaled strangely. Resize to 640x480 with the “mplayer” resize feature (a little confusing, not actually using **mplayer**, they just copied their code) and save as an **.avi** file with the **mjpeg** codec.

Embeded Flash Player for the Blog and Website

To post video on the website for public viewing, a service such as google video or youtube could be used, but the licensing and content control issues are murky and it's likely such free services will begin embedding free ads. Moss Landing has plenty of hosting bandwidth, so self hosting the video is advisable. To get a “youtube-like” embedded video player on the website itself, the video must be re-encoded as an **.flv** flash video file, then the appropriate html code edited to point to this **.flv** file on the Moss Landing server. Here's what I've been using on the blog:

```
<p id="player3"><a href="http://www.macromedia.com/go/getflashplayer">Get the Flash Player</a>
<script type="text/javascript">var s1 = new SWFOb-
ject("/flvplayer.swf","single","480","380","7");s1.addParam("allowfullscreen","true");s1.addVar
Caption Here!(<a href="http://scini.mlml.calstate.edu/video_clips/11aHDI_anemonesuck.avi">downl
```

Make sure you change the **player3** ids to something unique for every video on a page, change all the URLs to the correct video files, and change the caption text.

The “**Riva**” **.flv** encoder for windows is the best I've found, and I use the “**jw**” flash player. **avidemux** is supposed to have an **.flv** export feature in a forthcoming release, which would save a step.

* If the exposure rate is changed without stopping and restarting the video, the frame rate gets thrown off for different parts of the file. Might be worth taking notes on the frame rate as we go?