

Dance-A-Thon Scripts

Jason Healy, Director of Networks and Systems

Last Updated Mar 18, 2008

Contents

1	Dance-A-Thon Scripts	5
1.1	Introduction	5
1.2	Dance-A-Thon Overview	5
1.2.1	Hardware	6
1.2.2	Software	8
1.3	Usage	9
1.3.1	Software Preparation	9
1.3.2	Dance-A-Thon Preparation	10
1.3.3	After the Dance-A-Thon	11

Chapter 1

Dance-A-Thon Scripts

Last updated 2008/03/18

1.1 Introduction

This document describes the scripts and procedures for running the technical aspects of Suffield's Dance-A-Thon. The Dance-A-Thon is a charity event put on by the students, faculty, and staff. The event runs all night (until early the next day), and involves a majority of the student body dancing at all times.

The event is a lot of fun, and generates many "media-worthy" moments. We typically take over 1,000 pictures and several short movies at the events. To help organize all of this information, we have created some computer scripts which sort, label, and upload these files to the school's web site.

Note: this document is intended primarily for Suffield staff preparing to use our software for the Dance-A-Thon. Others are welcome to use the software for other purposes, but we make no promises about its adaptability to other tasks.

1.2 Dance-A-Thon Overview

The Dance-A-Thon is a several-hour-long event involving dancing, live music performances, contests, and games. Suffield uses technology during the Dance-A-Thon to accomplish the following tasks:

- Downloading and processing digital photos taken during the event.

- Downloading and processing digital movies taken during the event.
- Streaming a live feed of the audio to other computers.
- Displaying "special features" (trivia contests, music videos) on a large projected screen.
- Displaying a "slide show" of pictures taken at the event so far.

Because of the frantic pace of the Dance-A-Thon, there is not a lot of time nor manpower to handle these tasks. We therefore have created a series of scripts and procedures to help keep things flowing.

1.2.1 Hardware

The Dance-A-Thon requires us to use several pieces of hardware, all of which must work together with each other, and with our software.

Computers

We've managed to hone our hardware requirements down to a single primary machine that does most of the work during the event. The machine has the following minimum requirements:

- Dual-monitor capabilities (two independent screens, not mirroring)
- At least 10GB of free space on the hard drive
- Mac OS X (script tested against 10.4.4)

We currently use a dual-processor G5 computer from Apple, with a video card supporting two DVI outputs.

The computer provides a live streaming feed of the audio to other computers on the network. If you wish to receive this stream and play it, you will need another computer. In this case, the requirements are modest; any machine capable of receiving a QuickTime streaming broadcast will suffice.

Still Cameras

For still photography, our scripts require that all photos have the same dimensions (they assume a statically computed size, to save on computations during

image processing). Therefore, it is best to use the same model camera if multiple cameras are desired.

The scripts also make use of EXIF header information to determine image orientation. Therefore, the cameras **must support** this orientation metadata for the scripts to rotate the pictures correctly. If the cameras lack this feature, the user must manually rotate photos before processing.

We currently use Nikon D70 cameras (several of the people shooting photos have this model camera). The images are consistent, and the camera supports image orientation data.

Video Cameras

For motion photography, we use a camera with the following features:

- Low-light or infrared photography support (the dance floor is not well lit, so we use a camera capable of shooting in very low light).
- Native support for recording into a digital file format (*e.g.*, MPEG)
- Direct video output in a format compatible with the projector (see below)

We currently use a Sony TVR-350 handheld camera. It supports a "nightshot" low-light mode, and can save recorded movies directly to 320x240 pixel 15fps MPEG1 movies.

The quality of the MPEG movies produced by the camera is substantially less than that of DV tape. However, the major benefit is that the movies do not need to be imported into the computer and saved into a compressed format; we simply upload the movies directly from the camera to our web server. Because the web site is the intended delivery mechanism, the file quality is acceptable for this use.

Projector

We provide a live "slide show" of photos from the event, updated throughout the eventing. Additionally, we show music videos, trivia contests, and other media via the primary computer.

Additionally, we sometimes directly attach the video camera to the projector to provide a live shot of the room for everyone to watch. Therefore, the projector should have an additional input that works with the camera (*e.g.*, S-Video or composite). Many projectors provide a simple means for switching between inputs.

1.2.2 Software

We use a few pieces of software to manage the files during the evening. Some of this software is available, and some we have create ourselves.

Photo Sifting

We have written a short perl script called `sifter` which takes a folder of raw photos, renames them by date, and sifts them into directories based on the time they were taken. The script then builds an HTML gallery out of the photos (creating thumbnails as necessary).

The resulting gallery allows web users to quickly see all the pictures, and view larger sizes as needed. Additionally, a direct link to a photo ordering service is provided for each picture.

The script requires the `jhead` program for extracting EXIF information from photos, and the `jpegtran` package for performing lossless rotations of photos. Both programs are freely available, and should compile cleanly on Mac OS X.

Additionally, the script requires the `sips` (scriptable image processing system) binary, which comes pre-installed on all versions of Mac OS X since 10.3. It provides extremely quick image manipulation functions, such as resizing photos. We use it to make thumbnails. SIPS could be easily replaced with other image manipulation software (such as ImageMagick or NetPBM).

Video Sifting

We have written a short perl script called `vidindex` which takes a folder of raw MPEG videos and builds an HTML index to the files. The script looks for an optional file for each movie that contains a timestamp, title, and short description. If the file is found, the script uses these values when building the HTML index.

Because there are not as many videos taken during the night, the video script does not take as complex an approach as the picture filtering script. All the videos are listed on a single page.

Web Syncing

Once the photos and videos have been sifted, a short script called `sync` may be called to upload the resulting galleries to our web server. The script acts as a simple wrapper around `rsync`, which provides a fast and efficient way to update only the files that have changed since the last synchronization.

We use a passwordless SSH key so that the user does not need to enter a password to update the web site.

Additionally, the `sync` script supports a local (or remote, if reconfigured) synchronization of the photos to a "slide show" directory, so that the slide show photos can be easily updated throughout the evening.

Live Streaming

We stream the audio portion of the dance-a-thon over the network to other computers. The primary use of this stream is for computers elsewhere in the building where students may be taking a break between dances. Because they can hear the feed anywhere, they can return to the dance floor if they hear a song or event they'd like to participate in.

We use Apple's Darwin Streaming Server, which provides a quick and easy way to serve streaming media. The project is open-source, and precompiled binaries are available directly from Apple:

<http://developer.apple.com/darwin/projects/streaming/>

To author the live stream, we use Apple's QuickTime Broadcaster package. This product is available without charge from Apple:

<http://www.apple.com/quicktime/broadcaster/>

Double-Clickable

As a final note, we have created several `.term` files, which are double-clickable under Mac OS X. These files simply launch the command-line programs with the correct arguments.

1.3 Usage

This section serves as a guide for staff running the Dance-A-Thon using our setup. Please excuse the informal nature; this is intended mostly as "notes to self".

1.3.1 Software Preparation

Check out the scripts from the source repository, and locate the folder on your target machine. If you have a second hard drive with plenty of space, you might

want to locate the software there.

Now, edit the `sifter`, `vidindex`, and `sync` scripts, and confirm that the various variables are set correctly. This includes the year, start and end dates of the Dance-A-Thon, and paths to the source folders for photos and videos. The values currently there should give you some idea of what to create (if it doesn't already exist).

If you want to do a dry run, try setting the year to "test" and running the scripts. There's a `picturefaker` script which will pave over the EXIF dates of a folder of pictures (faking them so they appear to occur during the Dance-A-Thon that is yet to happen), so you can try the scripts out.

On the web server, create the subfolder for the current year, and create an HTML file in the top level with the current year (just copy an existing one and modify). Symlink it as "index.html" in the year folder.

While you're on the web server, edit the config file for the virtual host that contains the Dance-A-Thon files. There should already be a configuration section; just edit what's there (if there isn't, search this repository for a vhost config that should have the settings).

The apache files aren't strictly required, but they help with a few things:

- Cache-refresh settings (mainly for interaction with proxies and Coral Cache).
- Forced-download settings for full-size pictures.
- Automatic redirection to Coral Cache for large objects

Normally (*i.e.*, most of the year) the site is set to have a long content-expiration time, so that things get cached well. However, during the Dance-A-Thon we want to keep the updated pages fresh, so you should dial it down a little. The config should include commented out versions for the Dance-A-Thon.

Also, you may consider editing the `sifter` script and commenting out the Shutterfly order feature. That way, you can post the pictures immediately, but not worry about messed up filenames from interfering with orders. Once the Dance-A-Thon is done, you can confirm that the files are named correctly, uncomment the icon, and issue one more "build" command on the gallery.

At this point, you should have a functioning upload system, and you're ready to go!

1.3.2 Dance-A-Thon Preparation

On the night of the Dance-A-Thon, things are pretty straightforward. Here's a basic checklist of functionality that should be working before the event starts:

- `sifter dirs` to create the directory structure
- `sifter build` and `vidindex`, which will create the empty directories. Upload these empty directories so there's something to see (instead of a 404) before you put up any pictures.
- Fire up the streaming software, and test the feed from the DJ console to a remote machine. Check the levels!
- **Check dates and times on all cameras.** The scripts rely heavily on the EXIF dates provided by the cameras. If these dates get out of whack, the pictures will appear in the wrong place, or might be rejected outright by the script (the script ignores grossly inaccurate dates). You can fix bad dates with `jhead`, but it's a pain, so it's easier to make sure everything is set before you start.
- Confirm dual-monitor setup on any applications you'll be using on the computer (*e.g.*, Keynote, QuickTime Player). You'll want to set the application's "full screen" mode to use the projector screen, leaving the main screen free for other tasks.

Once you've run through this checklist, you should be good to go. Drink plenty of water, and settle in for the night!

1.3.3 After the Dance-A-Thon

When it's all over, make sure you've downloaded all final pictures, indexed them, and uploaded the results. If you disabled Shutterfly links, confirm that the photos are correct, and then rebuild the gallery with Shutterfly support.

Once you're **certain** that the galleries are correct, re-edit the Apache configs and turn up the cache expiration time.