Instructor's Guide to Deploying Calvin

Technical Requirements:

Instructor: A computer connected to the internet running Mac OS (Windows soon?)
A network drive, such as Box or Google Drive
A copy of Igor Pro (free demo for 1 month; coursework license = $125/yr)

Students: A computer or tablet connected to the internet

Each Calvin experiment requires 5-10 seconds for completion on a 2014 Macbook Pro, so a single computer should suffice for even the largest classes

Setting up Calvin

This process is very straightforward and should take a matter of minutes.

1. Install Igor Pro on the computer that will be serving the data

You can download a fully functional demo of Igor Pro from https://www.wavemetrics.net/. You can use this in demo mode for 30 days, after which you will need a license. A coursework license is currently $125/year and can be ordered at https://www.wavemetrics.com/products/igorpro.

2. Install Calvin on the computer

Startup Igor Pro and select `Help > Show Igor Pro User Files`. Drop all of the .ipf files in the folder named `User Procedures`. (There are currently 4 files: `Calvin.ipf, Calvin Command Reader.ipf, Calvin Spectrum to Color.ipf`, and `Calvin Student Communication.ipf`).

In Igor Pro, select `File > Open File… > Procedure`, navigate to the `Calvin.ipf` file you just installed, then click `Macros > Compile`. You can hide `Calvin.ipf` by clicking the red close box and selecting `Hide` (not `Kill`!) in the ensuing dialog.

3. Save your experiment in a new folder somewhere on your computer

4. Put the folder `StartupData` in the new folder

5. Edit 2 of the csv files in the `Startup Data` folder without changing their names.

`Groups.csv`: This file determines the maximum number of groups for any one TA. You can use more or fewer names. We typically assign 3-4 students to a group, and they all complete the lab together. We chose dog breeds for our group names, but any name that follows the Calvin naming convention would be fine.

Your task: Enter/edit the correct number of group names for your use. Names should follow the Calvin naming convention (Start with a letter and contain only letters, numbers, and the underscore.).

`TAs.csv`: You should enter the names of your TAs in this file, again using the Calvin naming convention. Each TA can have multiple groups, but each TA's groups must have different names from the choices in `Groups.csv`. For example, I have 18 TAs and 30 group names, so there can be a maximum of 540 groups in my class. (The actual number is far lower.)

6. Create two folders in your cloud account. The folders must be accessible in your path.

Cornell has an enterprise account with Box, so we downloaded and installed Box Drive from

https://www.box.com/resources/downloads/drive

After authentication, a new folder named `Box` appears on the Mac. This folder is accessible through the computer's file structure. Within Box, we made two new folders: `Chem_2080_Inbox` and `Chem_2080_Labs`.

We gave all TAs complete access to the second folder, `Chem_2080_Labs`, so that they could access misdirected files. For example, experiments in which the Group or TA name is missing or incorrect get sent to either the top level `_Lost&Found` folder (for no/bad TA name) or the TA's `_Lost&Found` folder (for no/bad Group name).

7. In the Calvin menu near the right-hand side of the Igor menus, select `Initialize Data`.

You will be asked the location of four folders:

Folder containing startup data: Choose the `StartupData` folder from step 4
The cloud folder: This is the folder that will contain the output files (*e.g.,* `Chem_2080_Labs`)
The inbox to monitor: This is the inbox folder (*e.g.,* `Chem_2080_Inbox`)

Save the Igor file so you will not have to reinitialize. Igor Pro can be a bit overwhelming for the uninitiated, so we have hidden most of its menus for simplicity. If you want them back, use `Calvin > Show All Igor Menus`.

8. (Optional, but recommended) Randomize the Unknown Acids, Buffers, and Indicators

You can do this using the menu items `Calvin > Randomize Unknown Xxxx and Save`. Igor Pro will do the rest. Alternatively, you can edit the csv files yourself. See the section later in this document about the files that describe unknowns for more information

9. Make all of the TA folders on the cloud drive

Select `Calvin > Make all TA folders`. Igor Pro does the rest. The folder hierarchy is sketched below. I suggest leaving the unused folders in place, so that students who use the wrong group name don't send their file to a non-existent folder, which will probably trigger an error.

```
├── Azbell (First TA name)
│   ├── Beagles
│   ├── Bloodhounds
│   ├── …
│   ├── Whippets
│   └── _Lost&Found
├── Bigham (Second TA name)
│   ├── Beagles
│   ├── Bloodhounds
│   ├── …
│   ├── Whippets
│   └── _Lost&Found
├── …
├── Yang (Last TA name)
│   ├── Beagles
│   ├── Bloodhounds
│   ├── …
│   ├── Whippets
│   └── _Lost&Found
└── _Lost&Found
    ├── Beagles
    ├── Bloodhounds
    ├── …
    ├── Whippets
    └── _Lost&Found
```

10. If desired, change the optional settings under the Calvin command. They are:

`Change maximum number of commands:` This is the maximum number of Calvin commands allowed per file. We don't want students performing an unphysical experiment by repeatedly adding one drop of acid/base, measuring *p*H, *ad infinitum.* The initial setting is 50. Until we get a chance to run Calvin in the classroom, we won't know the optimum setting, but this should be more than sufficient.

`Disable solid acids:` Disables commands that use solid acids. This will prevent students from using a solid acid to make a buffer in the unknown indicator experiment, for example.

`Disable buffers:` Disables commands that use buffers. This will prevent students from using a buffer in the unknown indicator experiment.

11. Choose `Calvin > Start Watching Folder`

Calvin will now process any file appearing in the watched folder, file the output in the cloud, and delete the processed file. If there are no files to be processed, Calvin checks for a new file every 15 seconds. If you want to stop watching the folder, hit command-. or click the abort button in the lower right hand corner to stop Calvin. If you want to restart, select `Calvin > Start Watching Folder`.

12. (optional) Submitting files by e-mail to Box

If you are using a Box account, navigate to the inbox through the Box web interface. Hover on the right hand side of the folder until an ellipses (…) menu appears. Choose settings for the folder. In the resulting screen, scroll down to the `Uploading > E-mail uploading` section. Select `allow uploads to this folder by e-mail`. Copy the address shown and distribute to your students.

Setting up the Groups

Once students have been assigned to groups, someone (*e.g.,* the TA) needs to give the students in each group access to their specific folder.

About the Files Describing Unknowns

`Acids.csv:` The first column, `unk_acid_fakeName`, defines the names of the known and unknown solid acids. The first two rows define the known solid acid, `KHP` (potassium hydrogen phthalate), and the acid used to make the buffer experiment, `imadizolium` (from imadizolium chloride). The remaining entries are whimsical names for the unknown acids. These names are less likely to get garbled by the students in transcription than "Unknown Acid 13."

The second column, `unk_acid_realName`, are the actual acids used in the experiment. We used a subset of these acids in real-world experiments because they dissolve easily. Any solid carboxylic acid would work. The third and fourth columns define the pK_a and molar mass of each acid.

`Buffers.csv:` This file is similar to `Acids.csv` in that it defines the names and concentrations of the unknown imidazole/imidazolium chloride solutions. The concentrations of the acid (imidazolium) and base (imidazole) have units of molarity. If you would like to use a different buffer, you need to include information about the acid (here `imidazolium`) in the `Acids.csv` spreadsheet and change the corresponding entry in the column `unk_buffer_acidRealName`. We switched to this buffer after the year that our acetic acid buffers grew bugs, which changed their composition with time.

`Indicators.csv:` The format of this file is similar to `Acids.csv`. The first 4 indicators, `Methyl_orange` through `Bromocresol_green`, are known indicators. The remaining indicators are unknowns.

For the Cognoscenti

Calvin is a custom interpreted language implemented in *Igor Pro* and running on an iMac. All of the data used by Calvin are real and were collected in the undergraduate labs by Melissa Hines shortly before Cornell went on lock down. All of the colors generated by Calvin are calculated from actual spectra of indicators using the CIE 1931 color space. A general introduction to this conversion can be found at https://www.fourmilab.ch/documents/specrend/.

The name Calvin is an homage to Python, which is an interpreted language named after *Monty Python's Flying Circus.* Python was written by engineer Guido von Rossum out of boredom while his research lab was closed over Christmas week in 1989. Boredom was not our primary motivator in writing Calvin, but we are nevertheless cooped up at home.

Igor Pro is an outstanding graphing and data analysis package for scientists and engineers. Igor is, of course, Dr. Frankenstein's lab assistant and also Calvin's supervisor. For more information about *Igor Pro,* visit WaveMetrics at https://www.wavemetrics.com/. Any piece of software that can enable a chemistry professor to write and deploy a new programming language around the world in a couple of weeks is wicked cool in my book.

— Melissa, the programmer, and Hobbes, the golden retriever