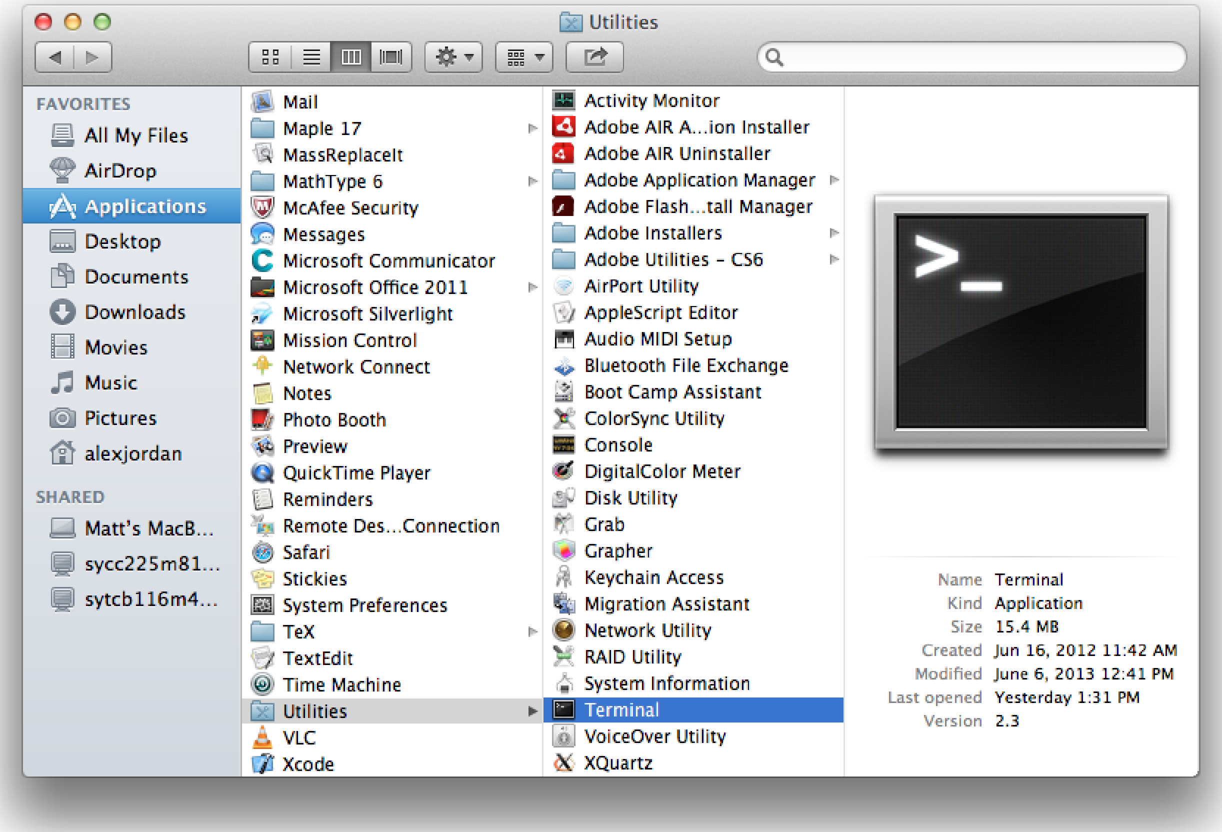
Getting Started with an MBX Project

MBX is the mechanism used to create the PCC calculus lab manual and other OER math books. This guide will get you started if you want to contribute on one of these projects.

This guide is for anyone who feels like a complete newcomer to projects like this. If you already know about things like git, then you should consider visiting the MBX homepage instead.

Until further notice, this guide assumes that you are working on a Mac. If you have a Linux machine, it is probably easy for you to work out the corresponding instructions. If you use Windows, the appropriate tools are still experimental and I'm unable to help with how to use them. For Windows users, for now, if you really want to work on one of these projects, you should find a Mac that you can use.

We'll do almost everything from the Terminal:



Name Terminal
Kind Application
Size 15.4 MB
Created Jun 16, 2012 11:42 AM
Modified June 6, 2013 12:41 PM
Last opened Yesterday 1:31 PM
Version 2.3

Installing MathBook XML (MBX) and Software Dependencies

MBX and MBX-based projects are not complicated things to install, but they rely on a fair amount of other software that we need first. The following steps worked on 12/01/2016 on a fresh MacBook Pro running OS X El Capitan. It's possible that if your OS is different, you may run into (overcome-able) obstacles.

1. From the Terminal prompt, install Homebrew by running:

```
/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

   You will be prompted for your Mac OS account's password.

2. Istall git:

```
brew install git
```

3. Make sure you are still in your user profile's root folder:

```
cd ~/
```

That is a tilde character, not a hyphen.

4. Clone the MathBook XML repository:

   ```
   git clone https://github.com/rbeezer/mathbook
   ```

5. Check out the dev branch of mathbook:

   ```
   cd ~/mathbook
   git checkout dev
   ```

6. Install the utility that converts pdf images to svg images:

   ```
   brew install pdf2svg
   ```

7. Install ImageMagick; among other things it provides the "convert" command which can convert pdf images to png images:

   ```
   brew install imagemagick
   ```

8. Install pip, the Python package manager:

   ```
   sudo easy_install pip
   ```

9. Install the Python requests package:

   ```
   sudo pip install requests
   ```

10. Install a full LaTeX distribution by visiting http://tug.org/mactex/. (This is for Mac OS X.)
    Download the .pkg there. It's a very big file, so you may want to download it using a wired internet connection as opposed to wifi.
    Once downloaded double click to open the installer and just go with all the default settings.

11. (Optional, if you would like to be able to make gource videos animating your repository history) install gource:

    ```
    brew install gource
    ```

12. (Optional, if you would like to be able to save gource videos as mpeg videos) install ffmpeg:

    ```
    brew install ffmpeg
    ```

13. You may need to exit Terminal and re-enter before some of what you have just installed is available.

Clone the MBX project that you wish to work on

1. From the Terminal, still in your main user profile folder (which you can be sure you are in by first running "cd /~"), clone the repository using one of the following (or something like these):

   For the PCC Calculus Lab Manual:

   ```
   git clone https://github.com/Alex-Jordan/Calculus-Lab-Manual
   ```

   For ORCCA (Open Resources for Community College Algebra):

   ```
   git clone https://github.com/PCCMathSAC/orcca
   ```

For APEX Calculus:

```
git clone https://github.com/PCCMathSAC/apex-mbx
```

2. Navigate into the project's root folder. You can do this through Finder if you like rather than Terminal.
   In this folder, there might be a file called "Makefile.paths.original". Copy this file as "Makefile.paths". Open the file and edit it according to the instructions there. Possibly, there is also a "Makefile.paths.example" you could use to help understand what you are supposed to edit.

3. Set up the git repository to know who you are:

```
cd ~/Calculus-Lab-Manual
git config user.name "First Last"
git config user.email name@domain.dom
```

Now what?

At this point you should have all the software that you need to compile the various output formats for the project. And you should be able to edit the project's source files to make edits and contribute. The details of how to actually do any of these things vary from project to project, so it's not feasible to give comprehensive instructions here. If you make it this far, contact Alex Jordan to proceed.