Building a developement enviroment for the fermitools on macosx

Purpose

I've started this page to keep track of how I'm trying to setup a fermitools development environment on my mac that includes fermipy,

The recipe below works, however it is extremely fragile.   Deviating from the recipe by changing the conda channels or the order of operations tends to cause it to break.

Setup

```
export PYTHON_VERSION=2.7
export FERMI_CONDA_ENV=fermitools-dev
export FERMI_CONDA_CHANNELS="-c conda-forge/label/cf201901 -c fermi"
export CONDA_CHANNELS="conda-forge"
conda create -y --name $FERMI_CONDA_ENV python=$PYTHON_VERSION
conda activate $FERMI_CONDA_ENV
conda install -y --name $FERMI_CONDA_ENV $FERMI_CONDA_CHANNELS -c $CONDA_CHANNELS --only-deps fermitools
conda install -y --name $FERMI_CONDA_ENV $FERMI_CONDA_CHANNELS -c $CONDA_CHANNELS fermi-repoman scons swig
```

Checkout source code

```
git clone https://github.com/fermi-lat/Fermitools-conda.git
git clone https://github.com/fermiPy/fermipy.git
mkdir <area for fermitools src>
cd <area for fermitools src>
repoman --remote-base https://github.com/fermi-lat checkout --force --develop ScienceTools conda # or whatever
tag you want
```

Build fermitools

```
export PREFIX="$CONDA_PREFIX"
export CPU_COUNT=2
export RECIPE_DIR="/Users/echarles/glast/releases/Fermitools-conda"
. ../Fermitools-conda/build.sh
export DYLD_LIBRARY_PATH="$CONDA_PREFIX/lib/fermitools"
```

(Where i commented out the line Fermitools-conda/build.sh that calls repoman to redo the checkout)

Build fermipy

```
cd ../fermipy
export CONDA_DEPS='scipy matplotlib pyyaml numpy astropy gammapy healpy astropy-healpix'
conda install -y --name $FERMI_CONDA_ENV $FERMI_CONDA_CHANNELS -c $CONDA_CHANNELS $CONDA_DEPS
python setup.py develop
```

Test fermipy

py.test

Updates 04 Feb 2020

The recipe above worked at first but then started failing b/c it couldn't load the dylib libraries for the fermtiools packages.   The crazy thing about this is that the libraries are being made, and

are present in the /Users/echarles/anaconda2/envs/fermitools-dev/lib/fermitools area.   But for some reason importlib is not looking there when asked to import the libraries for the various gt apps.

I'm going to retry this process with including a conda-build step in the process. Hopefully that gets the python setup up correctly.

The annoying thing about the conda-build process is that the build.sh script in Fermitools-conda wants to check out all the packages. So I'm going to try and modify it to use the version in place.