Help Topic: Downloading and Building the MOOS-IvP Extend Tree

Spring 2020

Michael Benjamin, mikerb@mit.edu
Department of Mechanical Engineering
MIT, Cambridge MA 02139

Downloading and Building the moos-ivp-extend Tree

A first step to writing and adding your MOOS-IvP code is to download the moos-ivp-extend tree.

This tree may be regarded as a template for building your own set of MOOS applications, Helm behaviors, or other utilities. It contains a single MOOS app and a single Helm behavior, but more importantly a file structure and CMake C++ build system for build the source code. Eventually you will want to understand more about Make files and CMake files. But for now, by starting with a working template, the addition of new MOOS applications to the build structure is a simple matter of editing one or two files.

Your goals in this part are:

1. From a terminal window download a copy of the moos-ivp-extend tree with the following svn command:

   ```
   $ svn co https://oceanai.mit.edu/svn/moos-ivp-extend/trunk moos-ivp-extend
   ```

   Verify that you can build this tree by:

   ```
   $ cd moos-ivp-extend
   $ ./build.sh
   ```

   It should add executable, pXRelayTest, in moos-ivp-extend/bin/.

2. Add the following directory to your shell path: moos-ivp-extend/bin

3. The pXRelayTest application built in this tree if very close to the pXRelay app distributed with the moos-ivp tree. Modify your xrelay.moos mission in moos-ivp-extend/missions/xrelay to use the pXRelayTest application instead. To do this, edit the Antler configuration block, in xrelay.moos, to use pXRelayTest on one or both of "Run" lines.

   ```
   // File: moos-ivp-extend/missions/xrelay/xrelay.moos

   o o o
     Run = MOOSDB        @ NewConsole = false
     Run = pXRelay       @ NewConsole = true ~ pXRelay_APPLES
     Run = pXRelayTest   @ NewConsole = true ~ pXRelay_PEARS      <-- Using pXRelayTest
     Run = uXMS          @ NewConsole = true
   o o o
   ```

4. Launch the modified mission and verify that *both* the `pXRelay` and `pXRelayTest` apps are running

```
$ cd moos-ivp-extend/missions/xrelay
$ pAntler xrelay.moos
```

You should see three windows pop up as in Figure 1 below. Note that one of them has the title bar "pXRelay" and "pXRelayTest".
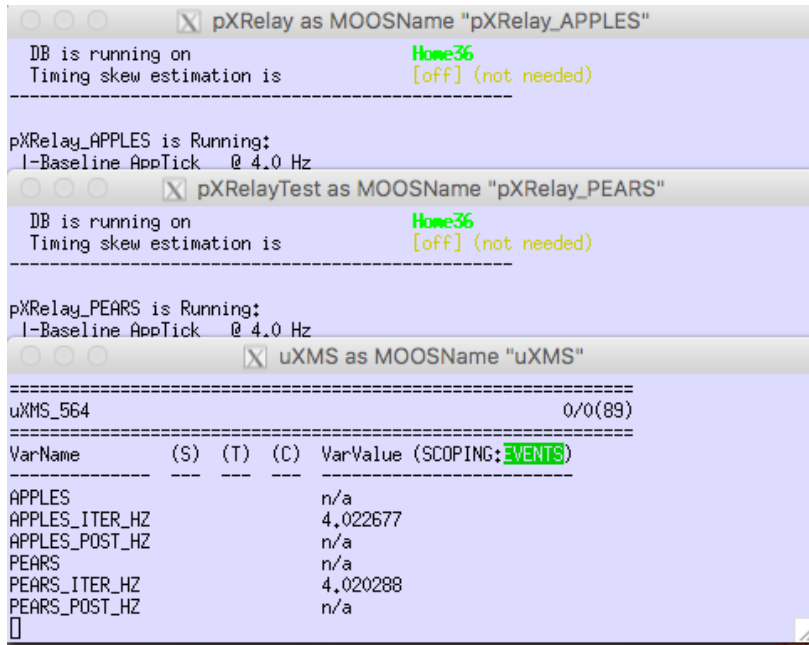

Figure 1: **The XRelay Mission**: Note that an instance of *both* `pXRelay` and `pXRelayTest` are running.

And, if you're familiar with this app, you know you can initiate the ping-pong behavior between the two of them by poking one of them with:

```
$ cd moos-ivp-extend/missions/xrelay
$ uPokeDB xrelay.moos APPLES=1
```

Troubleshooting

If you are not seeing the launching of the terminal windows as in Figure 1, it may be because you do not have the XTerm utility installed. This is likely installed by default in Linux systems, but as of this writing, it is not installed by default in MacOS. You can confirm using the `which` command. Hopefully you see something like:

```
$ which xterm
$ /usr/X11/bin/xterm    <--- Installed and in your shell path - good!
```

And not something like:

```
$ which xterm
$                       <--- Not installed, or not in your shell path - bad!
```

On MacOS, the recommendation is to install XQuartz from https://www.xquartz.org.