EPICS Client Install

2015-07-20

1 Install notes

This assumes you have a clean install of Linux Mint (I'm using 17.1 KDE)

1.1 Preparation

- Install a few things

```
sudo apt-get install ssh emacs build-essential libreadline-dev \
     mesa-common-dev libgl1-mesa-dev python-dev
```

1.2 EPICS Base

- Boot into Mint
- Log in as user
- Make a new user: enge, if you don't have it

```
sudo adduser --home /home/enge enge
```

- Add that user to the sudoers list

```
sudo adduser enge sudo
```

- Log out and then in as that user
- Make the file structure (I actually put these in a separate partition and linked to them from here)

```
mkdir bin
mkdir project
mkdir GUI
```

- Download and untar the EPICS base package

```
wget http://www.aps.anl.gov/epics/download/base/base-3.15.2.tar.gz
tar -xzvf base-3.15.2.tar.gz
ln -s base-3.15.2 base
```

- Paste the following into .bashrc

```
##########################################################################
##  EPICS
##########################################################################

## Base
export EPICS_ROOT=/home/enge
export EPICS_BASE=${EPICS_ROOT}/base/
export EPICS_HOST_ARCH=`${EPICS_BASE}/startup/EpicsHostArch`
export EPICS_BASE_BIN=${EPICS_BASE}/bin/${EPICS_HOST_ARCH}
export EPICS_BASE_LIB=${EPICS_BASE}/lib/${EPICS_HOST_ARCH}
if [ "" = "${LD_LIBRARY_PATH}" ]; then
    export LD_LIBRARY_PATH=${EPICS_BASE_LIB}
else
    export LD_LIBRARY_PATH=${EPICS_BASE_LIB}:${LD_LIBRARY_PATH}
fi
export PATH=${PATH}:${EPICS_BASE_BIN}

## EPICS Extensions
export EPICS_EXT=${EPICS_ROOT}/extensions
export EPICS_EXT_BIN=${EPICS_EXT}/bin/${EPICS_HOST_ARCH}
export EPICS_EXT_LIB=${EPICS_EXT}/lib/${EPICS_HOST_ARCH}
if [ "" = "${LD_LIBRARY_PATH}" ]; then
    export LD_LIBRARY_PATH=${EPICS_EXT_LIB}
else
    export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${EPICS_BASE_LIB}
fi
export EPICS_SYNAPPS_BASE=${EPICS_ROOT}/synApps
export EPICS_SYNAPPS_BIN=${EPICS_SYNAPPS_BASE}/support/utils
export PATH=${PATH}:${EPICS_EXT_BIN}:${EPICS_SYNAPPS_BIN}
```

- Load it

```
source ~/.bashrc
```

- Compile EPICS

```
cd base
make -j2
```

- Buy and drink some coffee
- Once finished, check that it works

```
softIoc
```

- Epics should have started. Now run the IOC

```
iocInit
```

- You should see something that looks like

```
epics> iocInit
Starting iocInit
############################################################################
## EPICS R3.15.2 $Date: Thu 2015-05-14 14:09:28 +0200$
## EPICS Base built Jul 17 2015
############################################################################
iocRun: All initialization complete
epics> exit
```

1.3 synApps (needed for streamdevice and serial connections)

- Get the extensions and msi first

```
cd ~
wget http://www.aps.anl.gov/epics/download/extensions/extensionsTop_20120904.ta
tar -xzvf extensionsTop_20120904.tar.gz
wget http://www.aps.anl.gov/epics/download/extensions/msi1-7.tar.gz
cd extensions/src
tar -xzvf ../../msi1-7.tar.gz
cd msi1-7
make
```

- Install re2c (I don't know what it's for)

```
sudo apt-get install re2c
```

- Download and unzip synApps

```
cd ~
wget http://www.aps.anl.gov/bcda/synApps/tar/synApps_5_8.tar.gz
tar -xzvf synApps_5_8.tar.gz
ln -s synApps_5_8 synApps
```

- We don't need all the junk included

```
cd synApps/support/configure
emacs RELEASE
```

- Edit the SUPPORT line

```
SUPPORT=/home/enge/synApps/support
```

- Edit $EPICS_{BASE}$

```
EPICS_BASE=/home/enge/base
```

- Comment out (with a '#') the modules we don't want
  - ALLEN_BRADLEY
  - AREA_DETECTOR
  - ADCORE
  - ADBINARIES
  - CAPUTRECORDER
  - CAMAC
  - DAC128V
  - DXP
  - IP
  - IP330
  - IPUNIDIG
  - OPTICS
  - QUADEM
  - SOFTGLUE
  - VME
- Prepare the makefile

```
cd ~/synApps/support
make release
```

- Compile!

```
make -j2 rebuild
```

1.4 Tidy up

- Make a folder to keep zip files

```
cd ~
mkdir Downloads
mv *.tar.gz Downloads
```

1.5 Qt GUI stuff

I've quite liked using Qt as a GUI. So far, `EpicsQt` has worked quite nicely, but I haven't tried to do anything complicated yet. In the mean time, we should also install CaQtDM.

1. Qt Install
    - Download Qt (includes Qt Creator) from the official website
    - Make sure you look for the open source one
    - This should have saved a file `qt-unified-linux-x64-2.0.2-1-online.run` in my case.
    - Make a folder to put this in

      ```
      sudo mkdir /opt/Qt
      ```

    - Install Qt in the folder you just made

      ```
      chmod 755 qt-unified-linux-x64-2.0.2-1-online.run
      ./qt-unified-linux-x64-2.0.2-1-online.run
      ```

    - This should install Qt. Check

      ```
      qtcreator &
      ```

    - Now add the following in .bashrc

      ```
      #### Qt
      export PATH=/opt/Qt/5.5/gcc_64/bin:/opt/Qt/Tools/QtCreator/bin:${PATH}
      export QWT_ROOT=/usr/local/qwt-6.1.2
      export QWT_INCLUDE_PATH=/usr/local/qwt-6.1.2/include/
      export LD_LIBRARY_PATH=/usr/local/qwt-6.1.2/lib/:/opt/Qt/5.5/gcc_64/lib:${I
      ```

    - Also install QWT
    - Download from `http://qwt.sourceforge.net/`

      ```
      source ~/.bashrc
      cd ~/Downloads
      tar -xjvf qwt-6.1.2.tar.bz2
      cd qwt-6.1.2
      qmake
      make
      sudo make install
      ```

2. CaQtDM Install
    - `https://github.com/caqtdm/caqtdm/archive/V3.9.4.tar.gz`
    - Download:

      ```
      cd ~/GUI
      wget https://github.com/caqtdm/caqtdm/archive/V3.9.4.tar.gz
      tar -xzvf V3.9.4.tar.gz
      mv V3.9.4.tar.gz ~/Downloads/caQtDM_V3.9.4.tar.gz
      ```

- caQtDM doesn't find variables on its own, so make sure `caQtDM_Env` has the right variables

```
if [ -z "$QTHOME" ];           then export   QTHOME=/opt/Qt;
fi
if [ -z "$QWTHOME" ];          then export   QWTHOME=/usr/local/qwt-6.1.2;
fi
if [ -z "$QWTINCLUDE" ];       then export   QWTINCLUDE=${QWTHOME}/include;
fi
if [ -z "$QWTLIB" ];           then export   QWTLIB=${QWTHOME}/lib;
fi
if [ -z "$EPICS_BASE" ];       then export   EPICS_BASE=/home/enge/base;
fi
if [ -z "$EPICSINCLUDE" ];     then export   EPICSINCLUDE=${EPICS_BASE}/inc
fi
if [ -z "$EPICSLIB" ];         then  export  EPICSLIB=${EPICS_BASE}/lib/$EP
fi
if [ -z "$EPICSEXTENSIONS" ];  then  export  EPICSEXTENSIONS=/home/enge/ext
fi
if [ -z "$QTCONTROLS_LIBS" ];  then export  QTCONTROLS_LIBS=`pwd`/caQtDM_Bi
fi
if [ -z "$CAQTDM_COLLECT" ];  then export  CAQTDM_COLLECT=`pwd`/caQtDM_Bina
fi
```

- Make sure python is defined as the correct version (I had to put 2.7) in `caQtDM_Env`
- Fix compilerSpecific.h

```
ln -s /home/enge/base/include/compiler/gcc/compilerSpecific.h /home/enge/ba
```

- Run the build script

```
./caQtDM_BuildAll
```

3. EpicsQt Install

- Download from `www.sourceforge.net/project/epicsqt` (I got version 3.1.0)
- Extract

```
mv epicsqt-3.1.0-src.tar.gz ~/GUI
cd ~/GUI
tar -xzvf epicsqt-3.1.0-src.tar.gz
mv 3.1.0 EpicsQt-3.1.0
```

- Add some things to `.bashrc`

```
## QtEpics
export QE_EPICS_BASE=${EPICS_BASE}
export EPICSQT_ROOT=${EPICS_ROOT}/GUI/EpicsQt-3.1.0
export EPICSCAQTDM_ROOT=${EPICS_ROOT}/GUI/caqtdm-3.9.4
export PATH=${PATH}:${EPICSQT_ROOT}/applications/QEGuiApp/bin:${EPICSCAQTDM
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${EPICSQT_ROOT}/framework/designe
export QT_PLUGIN_PATH=${EPICSQT_ROOT}/framework:${EPICSCAQTDM_ROOT}/caQtDM_
```

- and source: `source ~/.bashrc`
- For some reason, I found it easiest to do the rest of this compilation using Qt Creator. So load that now

  ```
  qtcreator &
  ```

- Make sure the correct version of Qt is being used. On a fresh install this should be easy enough, but you'll need to be careful if there are multiple versions of Qt on your computer.
- Load the `epicsqt.pro` file in the EpicsQt base directory
- Uncheck "shadow build" in "Projects"
- Add multi-processor building if you like by adding '-j2' to the make arguments
- Hit the "build" button!
  There will be lots of warnings but eventually it will finish. Hopefully without any errors...
- Close and reopen Qt Creator (from the command line)
- Open a test GUI and make sure it works
  - Open a form
  - Tools -> Form Editor -> About Qt Designer Plugins
  - Scroll down to make sure the EpicsQt plugins are loaded