ioctest Documentation

Release 2013-04

Pete R. Jemian

May 01, 2013

Contents

1 preparing the virtual machines **ii**
1.1 Add *~/.bash_aliases* file . . . iii
1.2 Add OS package: synaptic (optional) . . . iii

2 Install EPICS base **iii**
2.1 install Ubuntu package additions . . . iii
2.2 download . . . iii
2.3 unpack EPICS base . . . iv
2.4 compile EPICS base . . . iv
2.5 add EPICS base to your default path . . . iv
2.6 EPICS base Environment Additions . . . iv
2.7 testing . . . v
start *caRepeater* . . . v
test database file . . . v
start the test IOC . . . vi
access PVs from the test IOC . . . vi

3 Install MEDM **vi**
3.1 prerequisites . . . vi
3.2 downloads . . . vi
3.3 preparation of extensions directory . . . vii
3.4 edit extensions configuration . . . vii
3.5 build MSI and MEDM . . . viii
3.6 font alises . . . viii
3.7 add EPICS extensions to the environment setup . . . viii
3.8 run MEDM . . . viii

4 Install CSS-BOY **viii**
4.1 prerequisites . . . ix
4.2 download . . . ix
Java . . . ix
CSS-BOY . . . ix
4.3 install . . . ix
Java . . . ix
CSS-BOY . . . x
4.4 test . . . x

$Id: index.rst 1300 2013-05-01 06:46:50Z jemian $

author Pete R. Jemian

date 2013-04-30

subversion https://subversion.xray.aps.anl.gov/bcdaext/ioctest

URL https://subversion.xray.aps.anl.gov/bcdaext/ioctest/doc/_build/html/index.html

tinyURL http://tinyurl.com/ioctest

PDF `ioctest.pdf`

Install EPICS base and MEDM & CSS-BOY extensions on Ubuntu Linux 11.10 (no 11.04 release is available now on the mirrors). Document the installation steps, required drivers, OS package additions, download packages and sources, and environment variable settings required to obtain a functioning system.

Specific version numbers of software were those available at the time this document was written (2013-04-30).

The major sections of this document are:

- *preparing the virtual machines*
- *Install EPICS base*
- *Install MEDM*
- *Install CSS-BOY*

---

1 preparing the virtual machines

Prepare fresh Linux installations using the VirtualBox Graphical User Interface. (https://www.virtualbox.org/)

- created VirtualBox virtual disk images for x86 and x86_64 installations.
- configured each for 1024 MB RAM and to have RO mount of host root partition
- internet connection during OS installation:
  - 20 GB dynamically-expanding partitions, 1 GB RAM
  - host is connected
  - guest should report it is connected
  - did not download updates while installing
- http://mirror.mcs.anl.gov/pub/ubuntu-iso/CDs-Ubuntu/11.10/
- do *not* do any software upgrades, yet
- do not install VirtualBox Guest Additions, yet

architecture	disk image	image size
x86	ubuntu-11.10-desktop-i386.iso	711,980 kB
x86_64	ubuntu-11.10-desktop-amd64.iso	714,028 kB

1.1 Add *~/.bash_aliases* file

The standard Ubuntu account uses the *bash* shell. User modifications to the shell configuration can be added by creating the file: `~/.bash_aliases` from a terminal session, as follows:

```
touch ~/.bash_aliases
```

Since, we will be adding items to this file, open it with a graphical text editor:

```
gedit ~/.bash_aliases &
```

Note: Changes to *~/.bash_aliases* will not take effect until you create a new terminal session.

1.2 Add OS package: synaptic (optional)

Add the *synaptic* package to make it easier to identify needed packages to be added:

```
sudo apt-get install synaptic
# your account password here
```

---

2 Install EPICS base

Compilation of EPICS base consists of a few steps. Each one is not complicated.

- install the prerequisite Ubuntu package additions
- download the EPICS base compressed tar file
- unpack the tar.gz file into a directory
- compile using the *make* command

2.1 install Ubuntu package additions

Add these packages *before* you build EPICS base using this command:

```
sudo apt-get install g++ libreadline-dev
```

2.2 download

URL http://www.aps.anl.gov/epics/download/base/baseR3.14.12.3.tar.gz

size 1448 kB

In a terminal session (start one by pressing these three keys at the same time: <ctrl> <alt> t):

```
cd ~/Downloads
wget http://www.aps.anl.gov/epics/download/base/baseR3.14.12.3.tar.gz
```

2.3 unpack EPICS base

```
cd /usr/local
# here, we allow users to add to the /usr/local directory
sudo chmod go+w ../local
# enter your account password now to gain root privilege
sudo chmod o+t ../local
# create a directory for all EPICS content in the usual location
mkdir epics
cd epics
tar xzf ~/Downloads/baseR3.14.12.3.tar.gz
ln -s base-3.14.12.3 ./base
cd base
export EPICS_HOST_ARCH=`/usr/local/epics/base/startup/EpicsHostArch`
```

At this point, you will find a difference between 32-bit and 64-bit architectures to this question:

```
echo $EPICS_HOST_ARCH
```

architecture	response
32-bit	`linux-x86`
64-bit	`linux-x86_64`

Tip: Define *EPICS_HOST_ARCH* in *~/.bash_aliases* file (see *Add ~/.bash_aliases file*) so it is set every time you login. Use this command:

```
# point to the EPICS base we are using
export EPICS_BASE=/usr/local/epics/base
# determine the host type to be used by EPICS
export EPICS_HOST_ARCH=`${EPICS_BASE}/startup/EpicsHostArch`
```

2.4 compile EPICS base

Compile EPICS base and write the output of the process (including any errors) to file *makelog.txt*.

```
make 2>&1 | tee makelog.txt
```

2.5 add EPICS base to your default path

To make it easy for the EPICS tools to work together, add the EPICS base executables to your default path in *~/.bash_aliases*:

```
export PATH=${PATH}:${EPICS_BASE}/bin/${EPICS_HOST_ARCH}
```

2.6 EPICS base Environment Additions

With EPICS base now built, you should make some changes in your account's environment setup.

- define EPICS_HOST_ARCH (as noted in *unpack EPICS base*)
- start *caRepeater* (as noted in *start caRepeater*)
- add EPICS exectuables to the default PATH (as noted in *add EPICS base to your default path*)

Test these additions to the environment setup:

```
cd ~
which cainfo
```

The response should be from this table, depending on the architecture:

architecture	response
x86	`/usr/local/epics/base/bin/linux-x86/cainfo`
x86_64	`/usr/local/epics/base/bin/linux-x86_64/cainfo`

2.7 testing

You can test that EPICS has been built by starting a test IOC and then accessing the PVs from that IOC.

start *caRepeater*

A special program, *caRepeater* (from EPICS base), is used to assist EPICS communications amongst multiple client processes on the same host.

```
${EPICS_BASE}/bin/${EPICS_HOST_ARCH}/caRepeater &
```

Tip: start *caRepeater* when your workstation (or account) starts

---

Note: (source [1]):

If the CA Repeater is not already running on a workstation, then the *caRepeater* program must be in your path before using the CA client library for the first time.

If a host based IOC is run on the same workstation with standalone CA client processes, then it is probably best to start the caRepeater process when the workstation is booted. Otherwise it is possible for the standalone CA client processes to become dependent on a CA repeater started within the confines of the host based IOC. As long as the host based IOC continues to run there is nothing wrong with this situation, but problems could arise if this host based IOC process exits before the standalone client processes which are relying on its CA repeater for services exit.

---

test database file

A simple database file will be used to test EPICS base. Put these lines into a file called *ioctest.db*:

```
record(bo, "ioctest:binary")
{
        field(DESC, "boolean")
        field(ZNAM, "off")
        field(ONAM, "on")
}
record(ao, "ioctest:float")
{
        field(DESC, "floating-point number")
        field(PREC, "3")
        field(VAL,  "2.718281828459045")
}
record(stringout, "ioctest:text")
{
```

[1] https://www.slac.stanford.edu/accel/ilc/codes/ATF2/control-software/epics-3.14.8/base/html/CAref.html

```
        field(DESC, "some text")
}
```

start the test IOC

Start a new terminal session. Then, using the *ioctest.db* database we just created, start the IOC using *softIoc* from EPICS base:

```
softIoc -d ioctest.db
```

access PVs from the test IOC

Start another new terminal session and try accessing the PVs.

```
caget ioctest:float
caget ioctest:text.DESC
caget ioctest:text
caput ioctest:text This is some text
caget ioctest:text
cainfo ioctest:binary
```

command	description
caget	show the value of a PV
caput	set the value of a PV
cainfo	show information about a PV

---

3 Install MEDM

A useful graphical user interface for EPICS is *Motif Editing and Display Manager* (**MEDM**).

URL http://www.aps.anl.gov/epics/extensions/medm

Note: EPICS "extensions" (software that adds to EPICS in some way), such as MEDM, are always compiled and installed outside of the EPICS base directory tree.

3.1 prerequisites

install these Ubuntu add-on packages (other packages will be added automatically to satisfy dependencies):

```
sudo apt-get install libx11-dev  libxt-dev  libxmu-dev
sudo apt-get install libmotif-dev  xfonts-100dpi
```

3.2 downloads

Download these source code packages from the EPICS extensions web site (http://www.aps.anl.gov/epics/extensions):

```
cd ~/Downloads
wget http://www.aps.anl.gov/epics/download/extensions/extensionsTop_20120904.tar.gz
wget http://www.aps.anl.gov/epics/download/extensions/medm3_1_7.tar.gz
wget http://www.aps.anl.gov/epics/EpicsDocumentation/ExtensionsManuals/MEDM/medmfonts.ali.txt
wget http://www.aps.anl.gov/epics/download/extensions/msi1-6.tar.gz
```

3.3 preparation of extensions directory

To build MEDM (and other EPICS extensions), we need to create the directory structure described by http://www.aps.anl.gov/epics/extensions/configure/index.php

---

Important: Use the **New** method from 2012!

The *extensionsTop_20120904.tar.gz* file will create this structure for us.

```
cd /usr/local/epics
tar xzf ~/Downloads/extensionsTop_20120904.tar.gz
cd src
tar xzf ~/Downloads/msi1-6.tar.gz
ln -s msi1-6 ./msi
tar xzf ~/Downloads/medm3_1_7.tar.gz
ln -s medm3_1_7 medm
```

3.4 edit extensions configuration

1. Edit extensions/configure/RELEASE & point to EPICS base. Use these definitions (comment out any others):

   ```
   EPICS_BASE = /usr/local/epics/base
   EPICS_EXTENSIONS = /usr/local/epics/base/extensions
   ```

2. No need to edit the configure/CONFIG_SITE file.

3. Edit the *configure/os/CONFIG_SITE.${EPICS_HOST_ARCH}.${EPICS_HOST_ARCH}* file. Only necessary for linux-x86_64 architecture, x86 version is already this. Use these definitions (comment out any others):

   ```
   X11_LIB = /usr/lib
   MOTIF_LIB = /usr/lib

   #XRTGRAPH_EXTENSIONS = YES
   #XRTGRAPH = /usr/local/xrtgraph
   # Define SCIPLOT = YES if no XRT/graph 3.x
   SCIPLOT=YES
   ```

4. Create these symbolic links as root (noting slight differences for each architecture):

x86_64	**x86**
sudo su	sudo su
cd /usr/lib	cd /usr/lib
ln -s x86_64-linux-gnu/libX11.a ./	ln -s i386-linux-gnu/libX11.a ./
ln -s x86_64-linux-gnu/libXt.a ./	ln -s i386-linux-gnu/libXt.a ./
ln -s x86_64-linux-gnu/libXext.a ./	ln -s i386-linux-gnu/libXext.a ./
ln -s libXp.so.6 libXp.so	ln -s libXp.so.6 libXp.so

3.5 build MSI and MEDM

With the previous steps complete, it is now simple to build MEDM. Note that MSI will be built as a prerequisite for MEDM.

```
cd /usr/local/epics/extensions
make 2>&1 | tee makelog.txt
```

3.6 font alises

In addition you will find it better looking (and will get fewer warnings about missing fonts) to add the MEDM font definitions. Append the definitions from *~/Downloads/medmfonts.ali.txt* to */usr/share/fonts/X11/misc/fonts.alias*. Here's one way to install the MEDM font aliases.

```
sudo cat ~/Downloads/medmfonts.ali.txt >> /usr/share/fonts/X11/misc/fonts.alias
```

Another way is to open both files (as root) in a graphical editor and append them directly:

```
gksudo gedit ~/Downloads/medmfonts.ali.txt /usr/share/fonts/X11/misc/fonts.alias &
```

It is necessary to restart the X11 server to use these aliases. Verify the installation of these aliases using this command:

```
xlsfonts | grep widgetDM
```

This should produce various definitions for different point sizes.

3.7 add EPICS extensions to the environment setup

Add the extensions to the environment PATH in the *~/.bash_aliases* file (see *Add ~/.bash_aliases file*):

```
export PATH=${PATH}:/usr/local/epics/extensions/bin/${EPICS_HOST_ARCH}
```

3.8 run MEDM

Make sure the file `ioctest.adl` is in the starting directory. With the environment variables set as above, start *MEDM* in execute mode (any complaints about missing variables or the like is an indicator that one of the above stapes has not been followed closely):

```
medm -x ioctest.adl &
```

---

4 Install CSS-BOY

A useful graphical user interface for EPICS is the *Control System Studio Best OPI Yet* (**CSS-BOY**).

Installation of the CSS-BOY client is documented by Oak Ridge National Laboratory on their web site.

documentation http://ics-web.sns.ornl.gov/css/index.html

download http://ics-web.sns.ornl.gov/css/products.html

4.1 prerequisites

A Java Runtime Environment 1.6 or higher (http://ics-web.sns.ornl.gov/css/issues.html#Java) is required. A Java Development Kit includes this JRE plus developer tools; only JRE is needed to use CSS-BOY and create new screens for it. The JDK is needed to compile new Java tools from source.

JRE Java Runtime Environment

JDK Java Development Kit

4.2 download

Java

Download a JRE and the CSS-BOY (*Basic EPICS* edition) specific for the architecture:

For Java, visit http://www.oracle.com/technetwork/java/javase/downloads/index.html and click the **Download** button under the bold text **JRE** to start the process. You can't just use *wget* since you'll need to review and agree to a license to download the software.

I picked these for the *Java SE Runtime Environment 7u21*:

architecture	packages
x86	jre-7u21-linux-i586.tar.gz
x86_64x86	jre-7u21-linux-x64.tar.gz

CSS-BOY

Download the CSS-BOY *Basic EPICS* edition specific for the architecture:

```
cd ~/Downloads
```

architecture	packages
x86	`wget http://ics-web.sns.ornl.gov/css/updates/apps/epics_css_3.1.4-linux.gtk.x86.zip`
x86_64	`wget http://ics-web.sns.ornl.gov/css/updates/apps/epics_css_3.1.4-linux.gtk.x86_64.zi`

4.3 install

Java

Unpack the JRE into */usr/local/java* and add the Java exectuables (bin) directory to the PATH:

x86	x86_64
cd /usr/local	cd /usr/local
mkdir java	mkdir java
cd java	cd java
tar xzf ~/Downloads/jre-7u21-linux-i586.tar.gz	tar xzf ~/Downloads/jre-7u21-linux-x64.tar.gz

Add this line to the end of *~/.bash_aliases*:

```
export PATH=${PATH}:/usr/local/java/jre1.7.0_21/bin
```

CSS-BOY

Unpack CSS-BOY into */usr/local/CSS-BOY* and add to the path:

```
cd /usr/local
mkdir CSS-BOY
cd CSS-BOY
cp ~/Downloads/epics_css_3.1.4-linux.gtk.x86.zip ./
unzip epics_css_3.1.4-linux.gtk.x86.zip
rm epics_css_3.1.4-linux.gtk.x86.zip
```

Add this line to the end of *~/.bash_aliases*:

```
export PATH=${PATH}:/usr/local/CSS-BOY/CSS_EPICS_3.1.4/
```

4.4 test

1. Start CSS-BOY from the command line:

   ```
   cd ~
   css &
   ```

2. Use the default working directory for now.
3. Click on the **Workbench** icon near the top-right corner of the window.
4. In the *Navigator* panel on the left, click the triangle next to *CSS* to open the folder.
5. Right-click and select *New*, then *Other* ...
6. open the *BOY* triangle and select *OPI File*
7. click **Next**
8. select *CSS* as the parent folder
9. type this text in the *OPI File Name* box: *ioctest*
10. click **Finish**
11. close the newly-opened *ioctest.opi* from the graphical editor (click the "x" in the upper right corner of its graphical editor window
12. In the *Navigator* panel, right-click the *ioctest.opi* file and select *Open With*, then *Text Editor*
13. Erase all text in the ioctest.opi file.
14. Download, copy, and paste the contents of `ioctest.opi` into the *ioctest.opi* file you just opened.
15. Save the *ioctest.opi* file and close it.
16. In the *Navigator* panel, right-click the *ioctest.opi* file and select *Open With*, then *OPI Runtime*

Compare this with the MEDM session started above.

5 Complete *~/.bash_aliases* file

The additions to *~/.bash_aliases* are now complete:

```
# $Id: bash_aliases 1301 2013-05-01 06:54:25Z jemian $

# point to the EPICS base we are using
export EPICS_BASE=/usr/local/epics/base
# determine the host type to be used by EPICS
export EPICS_HOST_ARCH=`${EPICS_BASE}/startup/EpicsHostArch`

# add EPICS base executables
export PATH=${PATH}:${EPICS_BASE}/bin/${EPICS_HOST_ARCH}

# add EPICS extensions executables
export PATH=${PATH}:/usr/local/epics/extensions/bin/${EPICS_HOST_ARCH}

# start a caRepeater daemon if not already started
if [ "" == "`pgrep caRepeater`" ] ; then
  #echo "no caRepeater running, starting one now"
  ${EPICS_BASE}/bin/${EPICS_HOST_ARCH}/caRepeater &
fi

# add Java JRE executables
export PATH=${PATH}:/usr/local/java/jre1.7.0_21/bin

# add CSS-BOY executable
export PATH=${PATH}:/usr/local/CSS-BOY/CSS_EPICS_3.1.4
```