

# SNS Control Systems

---



*EPICS builds Control Systems*

## Software Engineering

*“SNS Standards for Building IOC Applications”  
Application Development Environment*

*January 20, 2004*

*Ernest L. Williams Jr.*

*ICS – Software Engineering Group*

1



**BROOKHAVEN**  
NATIONAL LABORATORY



**Los Alamos**  
NATIONAL LABORATORY

**ornl**

# OUTLINE FOR FIRST MEETING

---



- **Directory Structure**
  - » Production
  - » Shadow
- **Building IOC Applications**
  - » Templates
  - » configuration
  - » src
  - » Databases
  - » Screens
  - » iocCommon
  - » st.cmd
- **Software Release Process**
- **The Record/Device/Driver Developer**
- **The IOC Application Developer**
- **Summary of Standards**
- **Issues/Concerns**



# Control Systems Software Standards and Configuration Control.



## Production/Operations Server Directory Structure

### Production

- ADE\_TOP=/ade/epics
- SUPTOP=\$ADE\_TOP/supTop
  - » base
    - R3.13.7
    - R3.13.9
    - R3.14.4
  - » extensions
    - R3.13.7
    - R3.13.9
    - R3.14.4
  - » share
    - R3.13.7
    - R3.13.9
    - R3.14.4
- IOCTOP=\$ADE\_TOP/iocTop
  - » R3.13.7
  - » R3.13.9
  - » R3.14.4

### Shadow

- SHADOW=/ade/epics/shadow
- SUPTOP=\$SHADOW/supTop
  - » base
    - R3.13.7
    - R3.13.9
    - R3.14.4
  - » extensions
    - R3.13.7
    - R3.13.9
    - R3.14.4
  - » share
    - R3.13.7
    - R3.13.9
    - R3.14.4
- IOCTOP=\$SHADOW/iocTop
  - » R3.13.7
  - » R3.13.9
  - » R3.14.4

# Control Systems Software Standards and Configuration Control.



## *Production/Operations Server Directory Structure*

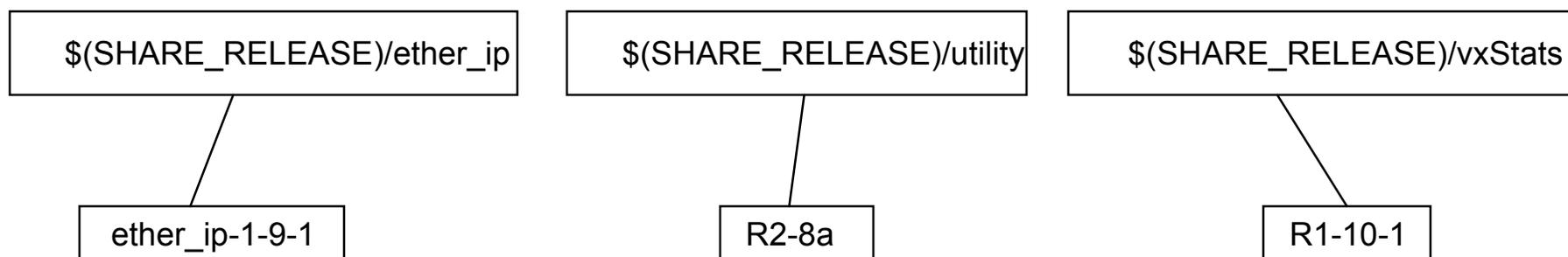
- **Production Areas**
  - » Configuration Management is under tighter control by the EPICS SysAdmin
  - » Software must be **RELEASED** with explicit **CVS RELEASE Tags**. (e.g. ether\_ip-1-9-1)
- **Shadow Areas**
  - » Configuration Management of IOC software and applications is managed by “IOC engineers” and “EPICS sysAdmin”
  - » This area is used for initial software releases, and to facilitate bug fixes to a production product.
  - » Beta applications can also be initially launched here. (e.g. a new version of EDM or VDCT)
  - » Development should not be done here.
    - We have an EPICS development server (i.e. ics-srv02.sns.ornl.gov)

# Control Systems Software Standards and Configuration Control.



## Example of Released Software directories

*SHARE\_RELEASE=\$SUPTOP/share/R3.13.9*



Use release names with syntax following a standard convention:  
RI-J-K or <cvsmoduleName>-I-J-K

I = Major release number (major software change was made or hardware changes occurred)

J = Minor release number (minor software changes or feature additions)

K = Software patch sequence number

# Control Systems Software Standards and Configuration Control.



## *Preparing to Build IOC Applications*

### **When IOC is ready for installation at the site:**

- **IOC engineer gives the computer system administrator (CSA) the IOC Name.**
- **<iocName> has been standardized to always be the DNS network name of the IOC.**
- **The CSA creates the <iocName> directory in the iocCommon directory and assigns the proper file permissions.**
- **The CSA creates the <iocName>/var directory and assigns the proper file permissions.**

**vxuser will have write permission in  
\$IOCS / <iocName> / var**

# Control Systems Software Standards and Configuration Control.



## *Preparing to Build IOC Applications*

### **When IOC is ready for installation at the site:**

- The CSA creates the appropriate entries in the NFS exports file. This allows the new <iocName> to mount the following “required” directories on the EPICS File/Boot server:
  - » Mounts – “/ade/epics/supTop/share”
  - » Mounts – “/ade/epics/Shadow/supTop/share”
  - » Mounts – “/ade/epics/iocTop”
  - » Mounts – “/ade/epics/Shadow/iocTop”
  - » Mounts – “/ade/epics/iocCommon”
- The CSA creates an entry in “/home/vxuser/.rhosts” file for each <iocName> on the EPICS File/Boot server
  - » All IOCs use rsh to load the vxWorks kernel; so setting up rsh is important
- The CSA creates an entry in the “/etc/hosts” for each <iocName> file on the EPICS File/Boot server
- The CSA adds an entry on the Domain Name Server (DNS) for each <iocName>

# Control Systems Software Standards and Configuration Control.



## *Preparing to Build IOC Applications*

### **When IOC is ready for installation at the site:**

- **To facilitate remote IOC access via the serial port the CSA will create an entry in the “/dev” file system that links to the serial port driver on the EPICS File/Boot server.**
  - » **/dev/<iocName>**
  - » **The following command connects to the IOC’s debug port:**
    - **cu -l /dev/<iocName>**
  - » **The following keyboard sequence disconnects from the IOC’s debug port:**
    - **<tilde> -- <period>**
  
- **To facilitate remote IOC “AC” power cycle via the serial port the CSA will create an entry in “/dev” file system and an entry in the “power list” file on the EPICS File/Boot server.**
  - » **The powerlist file is located in “/usr/bin/rpsdata”**
  - » **The following command power cycles <iocName>**
    - **powercycle <iocName>**

# Control Systems Software Standards and Configuration Control.



*Building IOC Applications ---- “Application Templates”*

*EPICS BASE R3.13.9*

## ● SNS VxWorks support Build Template

### » Template Name – sns

- `makeBaseApp.pl -T $TEMPLATE_TOP -t sns <name>`
- `makeBaseApp.pl -T $TEMPLATE_TOP -i -t sns <iocname>`
  - Where <name> is the App’s name for example
  - Where <iocname> is the IOC’s SNS network name. Which is derived from the SNS device name found in ORACLE (i.e. JERI)

### » When migrating one from EPICS BASE Release to another always check the templates for changes/additions to Makefiles, RULES, and Perl Scripts. This is where many build problems occur.

- **Suggest a migration script ---- “*copyTemplateApp.pl*”**

# Control Systems Software Standards and Configuration Control.



## Building IOC Applications ---- “Application Templates”

### EPICS BASE R3.14.4

- **SNS driver/device/record support Build Template**
  - » **Template Name – snsShare**
    - makeBaseApp.pl -T \$TEMPLATE\_TOP -t snsShare <name>
    - makeBaseApp.pl -T \$TEMPLATE\_TOP -i -t snsVx <iocname>
      - Where <name> is the driverApp’s name for example
      - Where <iocname> is the IOC’s SNS network name for example
- **SNS softIOC support Build Template**
  - » **Template Name – snsSoft**
    - makeBaseApp.pl -T \$TEMPLATE\_TOP -t snsSoft <name>
    - makeBaseApp.pl -T \$TEMPLATE\_TOP -t snsSoft <iocname>\_<port>
      - Where <name> is the App’s name for example
      - Where <iocname>\_<port> is the IOC’s name and procServ port.
- **SNS VxWorks support Build Template**
  - » **Template Name – snsVx**
    - makeBaseApp.pl -T \$TEMPLATE\_TOP -t snsVx <name>
    - makeBaseApp.pl -T \$TEMPLATE\_TOP -i -t snsVx <iocname>
      - Where <name> is the App’s name for example
      - Where <iocname> is the IOC’s SNS network name for example
- **When migrating from one EPICS BASE Release to another always check the templates for changes/additions to Makefiles, RULES, and Perl Scripts. This is where many build problems occur.**
  - » **Suggest a migration script ---- “copyTemplateApp.pl”**

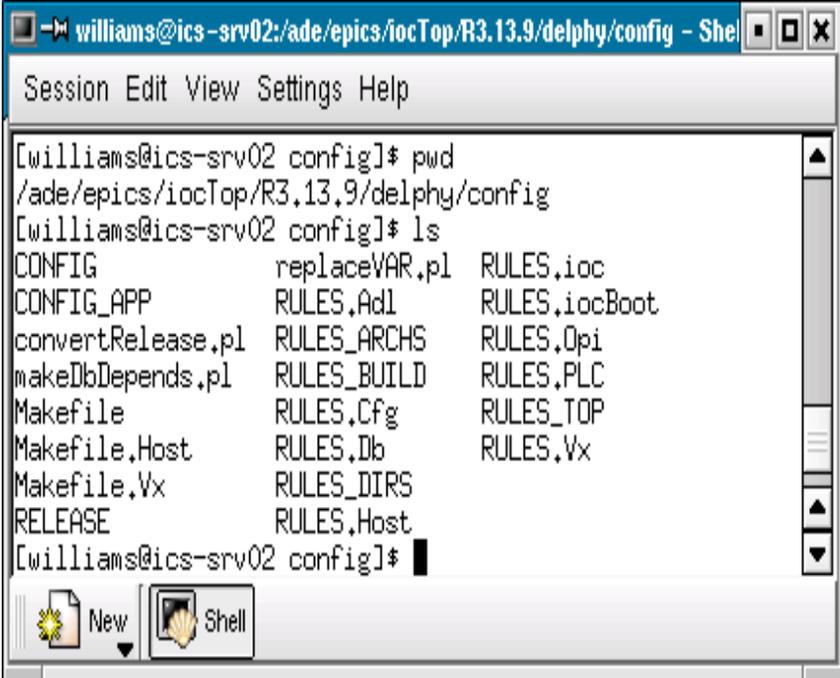
# Control Systems Software Standards and Configuration Control.



## Building IOC Applications ---- “configuration”

EPICS R3.13 build system configuration:

- **<TOP>/config/**
  - » Should contain files from the sns standard template.
    - CVS module: *snsTemplates*
  - » When moving between EPICS BASE software Releases be sure to migrate your “config” area to have the appropriate files.
    - Example: Depending on the CVS software Release of *snsTemplates*; **RULES.Db** could be different

A terminal window titled 'williams@ics-srv02:/ade/epics/iocTop/R3.13.9/delphy/config - Shell'. The window shows the output of 'pwd' and 'ls' commands. The 'ls' command lists various files and directories in a three-column format.

```
williams@ics-srv02:/ade/epics/iocTop/R3.13.9/delphy/config - Shell
Session Edit View Settings Help
[williams@ics-srv02 config]$ pwd
/ade/epics/iocTop/R3.13.9/delphy/config
[williams@ics-srv02 config]$ ls
CONFIG          replaceVAR.pl  RULES.ioc
CONFIG_APP      RULES.Ad1      RULES.iocBoot
convertRelease.pl RULES_ARCHS    RULES.Opi
makeDbDepends.pl RULES_BUILD    RULES.PLC
Makefile        RULES.Cfg      RULES.TOP
Makefile.Host   RULES.Db       RULES.Vx
Makefile.Vx     RULES_DIRS
RELEASE         RULES.Host
```

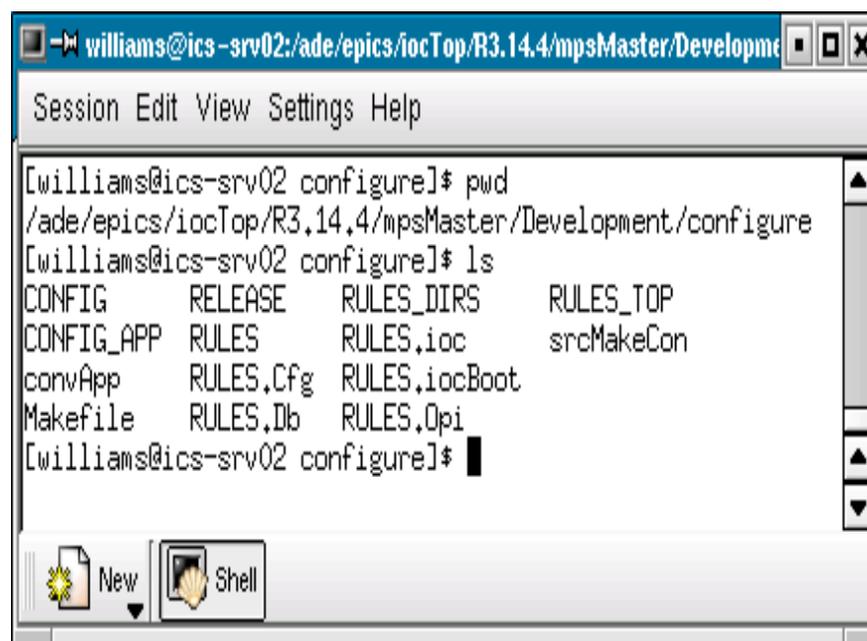
# Control Systems Software Standards and Configuration Control.



## Building IOC Applications ---- “configuration”

EPICS R3.14.4 and higher build system configuration:

- <TOP>/configure
  - » Should contain files from the sns standard template.
    - CVS module: *snsTemplates*
  - » When moving between EPICS BASE software Releases be sure to migrate your “configure” area to have the appropriate files.
    - Example: Depending on the CVS software Release of snsTemplates; RULES.Db could be different

A terminal window titled 'williams@ics-srv02:/ade/epics/iocTop/R3.14.4/mpsMaster/Development' showing a shell session. The user has entered 'pwd' and 'ls' commands. The output of 'ls' shows a directory listing with columns for file names and their permissions or attributes.

```
williams@ics-srv02:/ade/epics/iocTop/R3.14.4/mpsMaster/Development
Session Edit View Settings Help
[williams@ics-srv02 configure]$ pwd
/ade/epics/iocTop/R3.14.4/mpsMaster/Development/configure
[williams@ics-srv02 configure]$ ls
CONFIG      RELEASE    RULES_DIRS  RULES_TOP
CONFIG_APP  RULES      RULES.ioc   srcMakeCon
convApp     RULES.Cfg  RULES.iocBoot
Makefile    RULES.Db   RULES.Opi
[williams@ics-srv02 configure]$
```

# Control Systems Software Standards and Configuration Control.



## *Building IOC Applications ---- “configuration”*

- For R3.13.7 and greater
  - » <TOP>/config/RELEASE
    - This file must be used to reference/include external software components or modules
      - header files
      - Database definition files (i.e. \*.dbd files)
      - Libraries and executables (e.g. \*.o, and \*Lib)
      - Databases and Templates (i.e. \*.db and \*.template files)
  - » <TOP>/config/CONFIG
    - Specify target builds for IOC
- For R3.14.4 and R3.14.5
  - » <TOP>/configure/RELEASE
    - This file must be used to reference/include external software components or modules
      - header files
      - Database definition files (i.e. \*.dbd files)
      - Libraries and executables (\*.o, \*Lib, \*Lib.munch and \*.a)
      - Databases and Templates (i.e. \*.db and \*.template files)
  - » <TOP>/configure/CONFIG
    - Specify target builds for IOC

# Control Systems Software Standards and Configuration Control.



## *Building IOC Applications ---- "The source (src)"*

- Under R3.13.9
  - » <TOP>/<xxx>App/src
    - <xxx>Include.dbd
      - List all of the database definitions required by your App
    - Makefile.Host
      - DBEXPAND = <xxx>Included.dbd
      - DBDNAME = <xxx>App.dbd
    - baseLIBOBSJS
      - Reference objects required by your App.
      - De-reference objects not needed by your App
    - base.dbd
      - Reference definitions required by your App.
      - De-reference definitions not needed by your App
    - Note: baseLIBOBSJS and base.dbd must be in synch
    - Makefile.Vx
      - Include ../baseLIBOBSJS
      - LIBNAME = <xxx>Lib
      - BIN\_INSTALLS += \$(EPICS\_BASE\_BIN)/iocCore
      - BIN\_INSTALLS += \$(EPICS\_BASE\_BIN)/seq

# Control Systems Software Standards and Configuration Control.



*Building IOC Applications ---- "The source (src)"*

- **Under R3.14.4**

- » **<TOP>/<xxx>App/src**

- **<xxx>Include.dbd**

- List all of the database definitions required by your App

- **<xxx>Main.cpp**

- **Note: baseLIBOBS and base.dbd have been unbundled from EPICS BASE**

- **Makefile**

- » **For more details see the ADE for R3.14**

- <http://ics-web1.sns.ornl.gov/ADE/sns-ADE-miniHOWTOfor314-rev1.html>

# Control Systems Software Standards and Configuration Control.



## Building IOC Applications ---- “The source (src)”

<TOP>/<yourAppName>App/src/Makefile.Vx

- Bringing in external libraries into the “R3.13” build process
  - » Method 1: (Compile into your main IOC App library)
    - LIBOBS += (YYY\_BIN)/<libraryName>
    - Include ../baseLIBOBS
    - LIBNAME = xxxLib
      - Where “YYY\_BIN” is a variable derived from the RELEASE file that references the location of the library
      - Where “libraryName” is the name of the external library
      - Where “xxxLib” is the name of your IOC’s main Library
  - » Method 2: (pull from external location to your local bin)
    - BIN\_INSTALLS += (YYY\_BIN)/<libraryName>
      - This will copy the external library to <TOP>/bin/<ARCH>
        - » Where ARCH is ppc603 for example
  - » Method 3: (Don’t reference external library as part of the build)
    - The IOC can load external libraries by referencing them via:  
<TOP>/iocBoot/ioc<iocName>cdCommands when booting. Remember “cdCommands” is a product of the build process and must not be modified manually!! If you are missing something in the “cdCommands” file, you have left out the proper reference in your “<TOP>/config/RELEASE” file
      - The database definition file that accompanies the library should be referenced via “cdCommands” during IOC startup as well.
  - » The standard for production IOCs is Method 3

# Control Systems Software Standards and Configuration Control.



*Building IOC Applications ( vxWorks-based) ---- “The source (src)”*

`<TOP>/<yourAppName>App/src/Makefile`

- **Bringing in external libraries into the “R3.14” build process**
  - » **Method 1: (Compile into your main IOC App library)**
    - `PROD_IOC_vxWorks += xxx`
    - `DBD += xxx.dbd`
    - `xxx_LIBS += libraryName`
    - `xxx_LIBS += $(EPICS_BASE_IOC_LIBS)`
      - Where “libraryName” is the name of the external library
      - Where “xxx” is the name of your IOC’s main Library
  - » **Method 2: (Don’t reference external library as part of the build)**
    - The IOC can load external libraries by referencing them via:  
`<TOP>/iocBoot/ioc<iocName>cdCommands` when booting. Remember “cdCommands” is a product of the build process and must not be modified manually!! If you are missing something in the “cdCommands” file, you have left out the proper reference in your “<TOP>/config/RELEASE” file
      - The database definition file that accompanies the library should be referenced via “cdCommands” during IOC startup as well.
  - » **The standard for production IOCs is Method 2**

# Control Systems Software Standards and Configuration Control.



## Building IOC Applications ---- “Databases (Db)”

<TOP>/<yourAppName>App/Db

- Supported EPICS Database design tools
  - » vi (text-based editor)
  - » Vdct (graphical-based editor)
  - » JERI (Oracle-based editor)
- Database Source Files:
  - » <xxx>.template
  - » <xxx>.substitutions
  - » <xxx>.db
  - » <xxx>.vdb (VDCT hierarchical database files)
    - The EPICS Build system does not currently support <xxx>.vdb files
  - » <xxx>.acs (channel access security rules source file)
- Database Product Files
  - » <xxx>.template
  - » <xxx>.substitutions
  - » <xxx>.db
  - » <xxx>.acf (channel access security rules combined with common.acs)
- All database editors will operate on and/or create source files. These source files will then be turned into products via the EPICS make system or the JERI Tool.
- All source files for IOC applications will be under CVS control, especially database source files.

# Control Systems Software Standards and Configuration Control.



## Building IOC Applications ---- “Databases (Db)”

<TOP>/<yourAppName>App/Db

### ● Instantiating and Loading Databases

#### » Under R3.13.9/R3.14.4 and higher

- R3.13.9 “fully instantiated” databases are legal and preferred:

- Such as in “<TOP>/xxxApp/Db/Makefile.Host”
  - » DB += xxx.db
  - » USES\_TEMPLATE += <aaa>.template
  - » USES\_TEMPLATE += <bbb>.template
  - » USES\_TEMPLATE += <zzz>.template

- R3.14.4 “fully instantiated” databases are legal and preferred:

- Such as in “<TOP>/xxxApp/Db/Makefile”
  - » DB += xxx.db
  - » xxx\_TEMPLATE += <aaa>.template
  - » xxx\_TEMPLATE += <bbb>.template
  - » xxx\_TEMPLATE += <zzz>.template

- Instantiating databases at IOC boot-time is also legal:

- Such as in “<TOP>/iocBoot/ioc<iocName>/st.cmd”
  - » dbLoadRecords (“some.db”, “P=helloWorld”)
  - » dbLoadTemplate (“some.substitution”)

### ● All databases must be imported and “fully instantiated” into JERI (i.e. Oracle-based editor)

- » There are database importing tools in JERI
- » The database import can also be done as part of the EPICS build process.
- » The responsibility of importing databases into JERI lies with the IOC engineer.

### ● The standard for production IOCs is to only load “fully instantiated” databases:

- » JERI can now parse the “<TOP>/iocBoot/ioc<iocName>/st.cmd” file to automate the process of initially importing production databases. The same file can be parsed to keep Oracle in synch with the production IOC.

# Control Systems Software Standards and Configuration Control.



## *Building IOC Applications ---- “Databases and related files”*

- **Substitution or calibration files are allowed to live outside of the application’s <TOP>:**
  - » **Advantage: Not hindered by the IOC CVS RELEASE process when frequent changes occur to alarm limits, engineering limits, deadbands, etc...**
  - » **Substitutions file may reside in ORACLE**
  - » **Substitution files may also reside on a subsystem specific area of the server’s file system:**
    - **Example for MPS**
      - */ade/epics/iocCommon/Support/mps/*
        - » Fully instantiated databases, configuration, and firmware files live here
    - **Example for Magnet Power Supplies**
      - */ade/epics/iocCommon/Support/magnets*
        - » Fully instantiated databases, calibration data file, magnet mapping files
  - » **The database, calibration, etc... files are still versioned by CVS but do not have to be part of a “CVS RELEASE”**

# Control Systems Software Standards and Configuration Control.



## Building IOC Applications ---- “EDM Screens (srcOpi)”

- Extensible Display Manager (EDM) is the standard tool for building EPICS Operator Interfaces.
- Screen Development will follow the Human Machine Interface (HMI) standard:
  - » <http://www.sns.gov/projectinfo/ics/119/ctrlStdsHandbook/operInterfaceStds/hmi/hmiSummary.html>
- Most EDM Screens will be created as a source files. The EDM source files will live in: <TOP>/xxxApp/srcOpi
  - » The EPICS build system will install the source screens in the following product directory: <TOP>/opi
- As operations gets more involved with screen development and we move more into creating operator screens that span across multiple IOCs, we will begin to de-couple screens from an IOC <TOP>.
  - » Screens are not loaded by an IOC
  - » Some screens change often and productivity can be hindered by the CVS RELEASE process. Versioning will still be done by CVS
- Currently most top/system level EDM screens are created in snsMachine. The screens in snsMachine are constantly undergoing modifications. So, snsMachine **will not be CVS Released** but wil still be versioned with CVS

# Control Systems Software Standards and Configuration Control.



## *Building IOC Applications ---- “EDM Screens”*

### *“/ade/epics/opiCommon”*

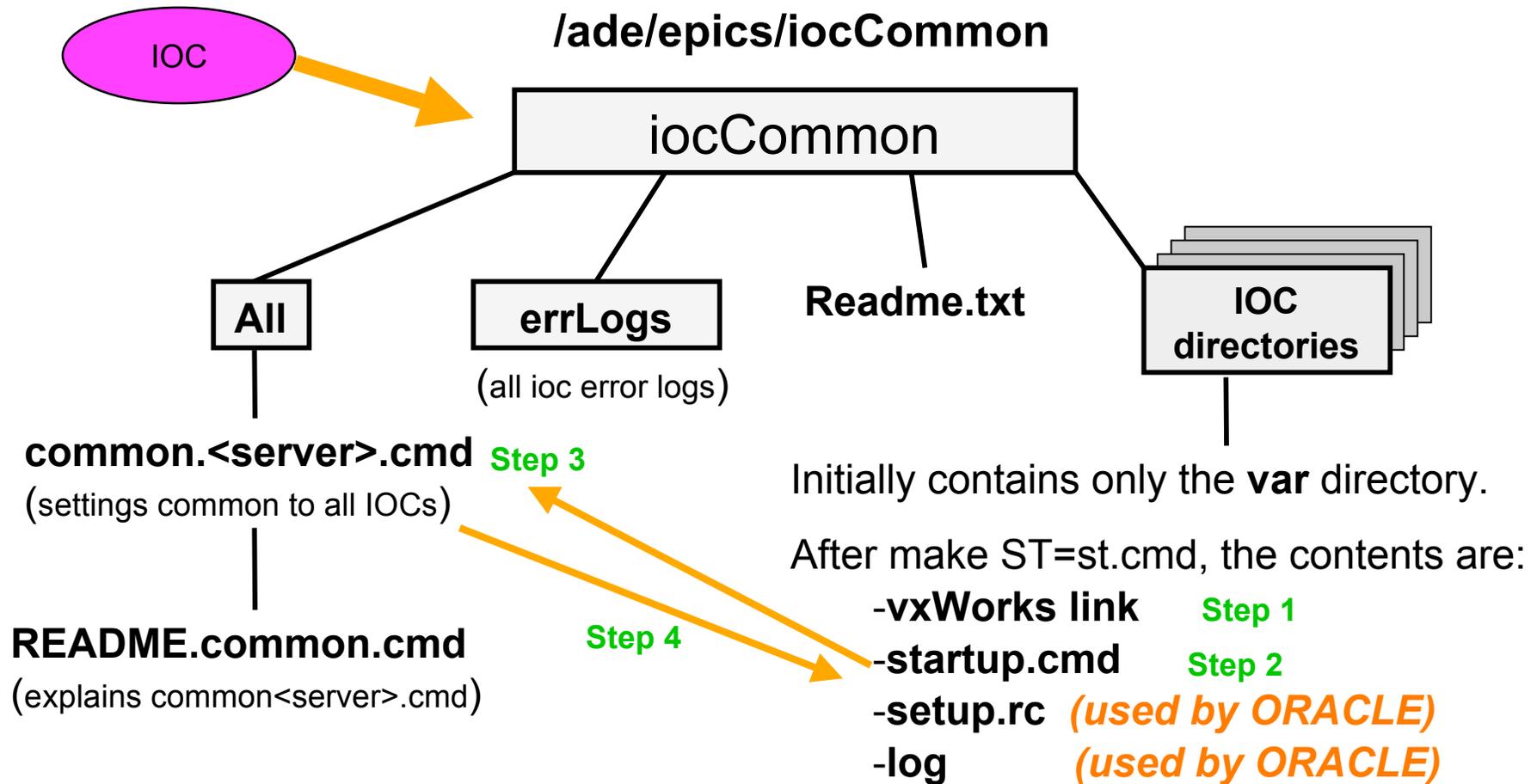
- opiCommon provides the IOC engineer a mechanism to place a file (<xxx>.opi) which contains the path to the desired EDM screens in a common area. We use “OPI\_COMMON” environment variable to contain the path opiCommon area.
  - » Each file in opiCommon can be automatically added to the EDM Display path. (i.e. EDMDATAFILES)
    - A task for the Computer System Administrator (CSA)
  - » Advantage: changes to screens can be pushed into the EDM display path without waiting for the EPICS sysAdmin
  - » Well how does one push the screen path to opiCommon area?
    - cd your <TOP>
    - Modify your RELEASE file to reference snsMachine. This is where the Build RULES reside for handing opiCommon
    - make OPI=yes

# Control Systems Software Standards and Configuration Control.



*st.cmd*

All IOCs must start here



# Control Systems Software Standards and Configuration Control.



## *IOC Boot Process*

- **startup.cmd** – executes:
  - » common.<server>.cmd which is located in “/ade/epics/iocCommon/All” directory.
  - » st.cmd which exists in your <TOP>/iocBoot/ioc<iocName> directory in your App area.
- **common.<server>.cmd** – mounts areas of the servers that need to be accessed by each IOC and sets EPICS environment variables.
- **st.cmd** – local start-up command file for the IOC which is run at boot time.

# Control Systems Software Standards and Configuration Control.



## *Standard Sequence for 3.13.9 & 3.14 st.cmd files*

- **Network Setup**
- **Load “cdCommands”**
  - » **Contains aliases to the full path and CVS Release of software components.**
- **Load EPICS kernel**
  - **Load all libraries needed for your application**
  - **Load MPS Related Software if needed (here)**
  - **Initialize hardware and drivers**
- **Load databases**
- **Load databases**
- **Load Channel Access Security files**
- **Create a file containing a list of PVs loaded on the IOC**
- **Initialize EPICS Kernel (i.e. run ioclnit)**
- **Post-Processing for autoSaveRestore**
- **Start Sequence Programs**

# Control Systems Software Standards and Configuration Control.



*Getting “/ade/epics/iocCommon/<iocName>” populated*

- **cd <TOP> and modify your RELEASE file to reference the CVS Released vxWorks location. If migrating an existing App, be sure to check/follow the snsTemplates.**
- **cd <TOP>/iocBoot/ioc<iocName> and modify the Makefile in this directory for your system again according to the snsTemplates.**
  - » **make ST=st.cmd**
- **This will populate “/ade/epics/iocCommon/<iocName>” with the proper files.**
- ***<http://ics-web1.sns.ornl.gov/ADE/sns-ADE-miniHOWTOfor314-rev1.html>***

# Control Systems Software Standards and Configuration Control.



*st.cmd*

- For an example of *st.cmd* for 3.14.4 go to

`/ade/epics/supTop/share/R3.14.4/snsTemplates/R2-3/makeBaseApp/top/snsVxBoot/ioc/st.cmd@vxWorks`

- For an example of *st.cmd* for 3.13.9 go to

`/ade/epics/supTop/share/R3.13.9/snsTemplates/R1-4/makeBaseApp/top/snsBoot/ioc/st.cmd`

# To Release or Not to Release

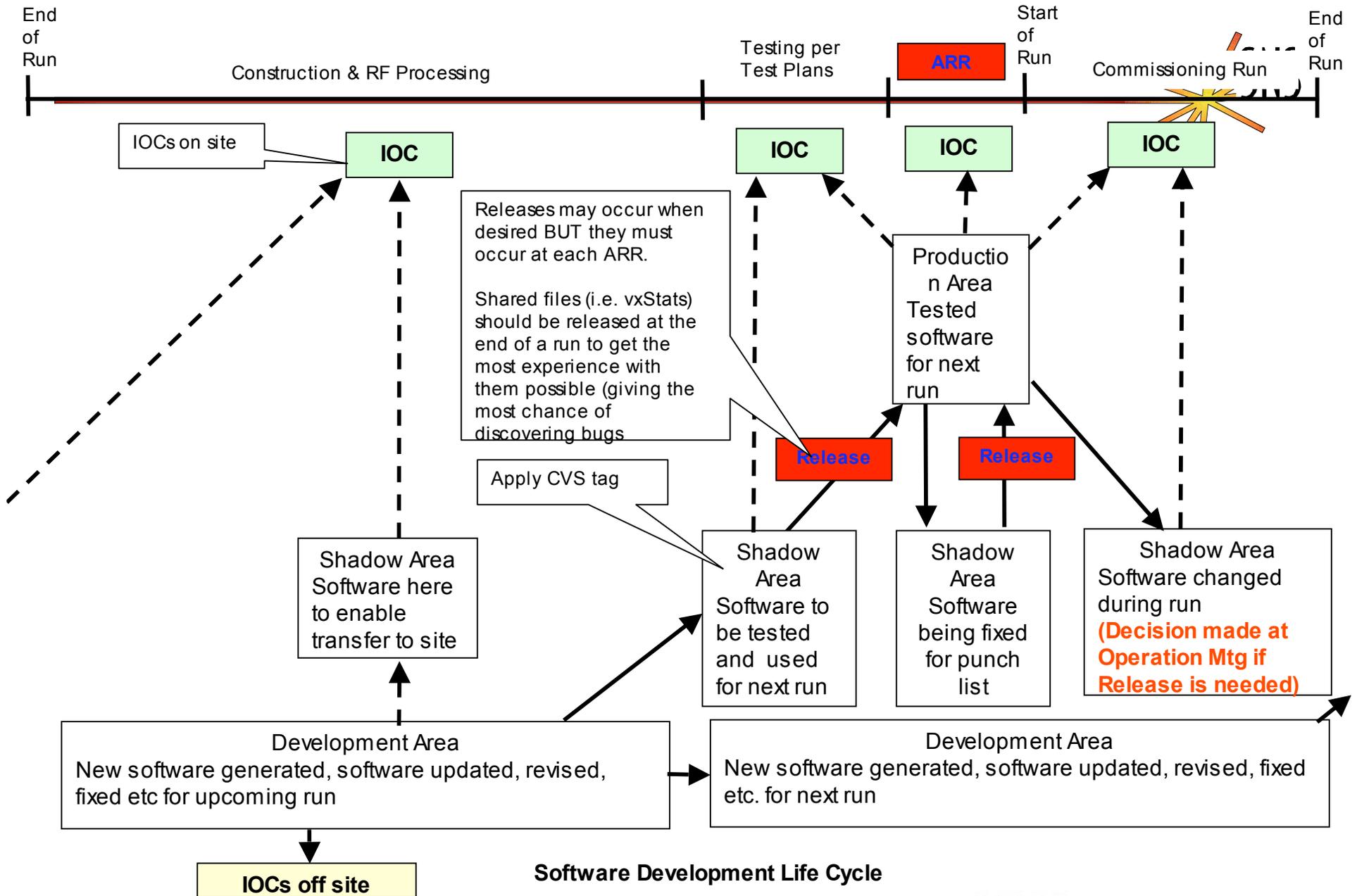


- **Scope – This criteria applies to QA Level 2 & 3 systems. It does not apply to QA Level 1 systems (PPS, TPS, NFSS, Fire Alarm)**
- **All EPICS & PLC software initially being placed in the production area (not previously released) must be released prior to the ARR where it will be used**
- **Any changes to previously released software:**
  - » **Affecting the form, fit, or function of MPS or any other level 2 system**
  - » **Affecting interfaces (signals from or to, trips, resets, etc) to other systems (like vacuum sending a trip to LLRF)**
  - » **Due to FSD revisions (or causing FSD revisions)**
  - » **Affecting Operating (not Expert) screens**
  - » **Affecting Summary PV calculations**
  - » **Causing impact on Alarm Handler Summary PV points**
  - » **Causing the IOC Test Plan to be run**
  - » **Affecting Save/Restore/Bumpless Re-boot**
  - » **Resulting from changes in EPICS versions, EDM versions, or Driver software**
  - » **Affecting PV names**

# Stuff not needing release



- **Changes to previously released software that affect**
  - » Interlock logic not affecting other systems
  - » PID loop tuning parameters
  - » Alarm limits (if they don't impact other systems)
  - » Expert Screens
  - » PLC Forces
  - » Archiver files or settings
- **When in doubt, Dave Gurd, George, or Mario make the call**



Software Development Life Cycle

# Control Systems Software Standards and Configuration Control.



## Software Release Process

- **Pre-Release Actions**
  - » Prepare a Test Plan for testing all system application software if new, or the software that has changed.
  - » Re-vise the FSD if it is impacted and transmit the revision to ProjectWise
  - » Use the test plan to test the software as thoroughly as possible in a development environment. (Sign off not needed)
  - » Move IOC software from CVS into an appropriate shadow area on the production EPICS file server and build.
- **Release Actions**
  - » Step 1: Run the IOC Test plan if applicable
  - » Step 2: Re-Test the software using the test plan and sign off the test plan
  - » Step 3: Following Test Plan approval, coordinate with operations to test/integrate IOC software during a maintenance window; create a Maintenance and Controls E-log entry as well.
  - » Step 4: After successful testing (sign off), create a production CVS RELEASE tag with update README and RELEASE\_NOTES
  - » Step 5: Export official CVS RELEASE into production area and coordinate with operations during maintenance window to install new RELEASE.
  - » Step 6: Create a log entry in the Maintenance and Control Systems Section of the E-log book. Announce new RELEASE via e-mail to EPICS SysAdmin as well.
  - » Step 7: Send e-mail to George, Mario, CF Operations, CHL Operations?, John Munro, Delphy, John Cleaves, Ron Battle, Jeff Patton, Coles Sibley, and all IOC Engineers stating
    - Brief description of change
    - System, subsystems, and IOCs impacted
    - Impact on Alarm Handler, Archive, or Drivers
  - » Step 8: EPICS SysAdmin will now “lock” the portion of the software subject to re-release if changed. Essentially the file permissions will be changed to Read-Only.

# Control Systems Software Standards and Configuration Control.



## *What constitutes a RELEASE?*

- All the source files under a <TOP> that creates a build for an IOC or shared component
- Files that change often can opt not to use RELEASE numbers but must be still versioned under CVS.
  - » Database Substitution Files
  - » Configuration files generated by an external system such as Oracle.
  - » EDM screens that change frequently

# EPICS Record/Device/Driver Maintainer's Responsibility



- Provide “R3.13” and R3.14 drivers following the SNS ADE
  - » TestApp demonstrating driver functionality
  - » Database Templates and/or EDM screen Templates if needed
  - » Provide stand-alone driver libraries and database definition (dbd) files, include files.
  - » Library Scheme for R3.14
    - lib<Driver>.a (Static library)
    - <Driver>Lib (dynamic library)
    - <Driver>Lib.munch (dynamic library)
    - <Driver>.dbd (used for registration in C++ also)
  - » Documentation
    - README
    - RELEASE\_NOTES
    - User’s Guide (HTML/PDF)
  - » Include driver routine to support dbior.
  - » RELEASE Driver/Device/Record Support according to the “**SNS Software RELEASE Standards**”

# ***IOC Engineer Responsibilities***



- **Confirm that IOC is running properly and enter this confirmation in the eLog.**
- **Confirm and record in eLog that data is being properly archived.**
- **Create and maintain a Global Archive for your system. Operations can help with configuration.**
- **Create Archive request files and provide up-to-date request files to the archiver team for installation.**
- **Create Alarm handler files and provide up-to-date alarm handler files to John Munro for installation.**
- **Ensure that your system recovers from a reboot. This means deploy autoSaveRestore software if necessary**
- **Verify that all pvs from your ioc are connecting on edm screens. Any screens with unconnected pvs should be addressed.**
- **In the event that an IOC reboot is necessary, obtain permission from Chief Operator and Dave Gurd prior to rebooting.**
- **Record entry in eLog to document the process immediately following the reboot.**
- **Accept and honor on-call duty assignments for nights and weekends to help provide operator support when it is needed.**



# Standards Summary



- All IOC's use iocCommon for the boot process.
  - » Use make ST=st.cmd
- All “IOC Apps” and “Share Apps” use the SNS Standard Templates (i.e. cvs module snsTemplates)
- Reference all external libraries (e.g. as the ones in \$SHARE) from within your IOC's st.cmd file. Do not link in external libraries at compile time.
- Create only fully-instantiated databases for the IOC to load via the st.cmd file.
- All IOC should be running the stable version of VxWorks, currently SNS06a
  - » Check with EPICS Sys Admin for official Release status of vxWorks
- All IOC's are configured to support Channel Access Security
- All IOC's use at a minimum the following components from \$SHARE:
  - » timestampRecord
  - » vxStats
  - » utility
  - » autoSaveRestore
- All IOC's follow the st.cmd file according to the snsTemplates
  - » No references to softlinks should be in the cdCommands file
  - » No NFSMounts except as provided in IOCCommon
- All Control Systems Software is maintained with the SNS versioning system
- All Control System Software production components that are subject to CVS RELEASE Tags follow the SNS RELEASE Process.
  - » Other components are separated if not subject to CVS RELEASE Tags
    - Example: databases, screens, configuration files from Oracle, etc...

# Issues/Concerns



- We will apply the standards for at least 6 months and then re-evaluate.
- Need to contact CosyLab with concerns on making the hierarchical databases (i.e. xxx.vdb) a part of the EPICS build system.
- Need to get some helpful scripts and tools to help automate more things in the IOC build process.
- There are still IOCs not set-up according to the standard (running under R3.13.5/R3.13.7)
- Goal: The DTL 1-3 run is in March. We need all IOCs set-up according to the SNS standard (**running R3.13.9/R3.14.4.**)
  - » Can we get there **ahead of schedule**?
  - » Can we re-adjust **priorities** so that IOC standardization and EPICS Base Release migration can't be preempted by other tasks?