Introduction to EPICS

John Maclean, Argonne National Laboratory
Overview

- Lay the foundation for understanding an EPICS control system
- Introduce IOCs
  - Channel Access (CA)
  - Database
  - Sequencer
  - Device Support
- Choosing the correct tools for the job
  - When to use a database
  - The sequencer, what is it good for?
  - Why write your own CA client program?
- How fast is EPICS?
Canonical Form of an EPICS Control System

Client Software
- MEDM
- OAG Apps
- ALH
- TCL/TK
- StripTool
- Perl Scripts
- Many, many others

Channel Access

IOC Software
- EPICS Database
- Sequence Programs
- Real-time Control
- Custom Programs

CA Server Application
- Process Variables

Introduction to EPICS
Introducing the IOC

- Input Output Controller
- A computer running software called “IOC Core”

The computer can be:
- VME based, running vxWorks, RTEMS, Linux
- PC running Windows, Linux, RTEMS
- Mac running OSX
- Embedded processor running RTEMS, Linux

- Usually has Input and/or Output devices attached
- An EPICS control system must consist of at least one Channel Access Server (usually an IOC)
- An IOC has one or more databases loaded. The database tells it what to do
Inside an IOC

The major software components of an IOC (IOC Core)
Channel Access

- Allows other programs (CA Clients) to see and change values of Process Variables in an IOC (CA Server)
- CA Clients may
  - Put (write)
  - Get (read)
  - Monitor

**data of Process Variables**

- IOCs are both CA clients and CA servers. They can interact with data in other IOCs
- A CA Client can connect to many servers
- A CA Server may serve many clients
- A very efficient and reliable protocol
Inside an IOC

The major software components of an IOC (IOC Core)
EPICS Databases – What are they for?

- Interface to process instrumentation
- Distribute processing
- Provide external access to all process information
- Use common, proven, objects (records) to collect, process and distribute data
- Provide a common toolkit for creating applications
What are records?

- A record is an object with
  - A unique name
  - Properties (fields) that contain information (data)
  - The ability to perform actions on that data
- A personnel record in a relational database has a name, and fields containing data

<table>
<thead>
<tr>
<th>Unique record name</th>
<th>XYZ1234</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee</td>
<td>James Bond</td>
</tr>
<tr>
<td>Badge #</td>
<td>007</td>
</tr>
<tr>
<td>Address</td>
<td>Whitehall, London</td>
</tr>
<tr>
<td>Salary</td>
<td>£70070.07</td>
</tr>
</tbody>
</table>
What are EPICS records?

- A record is an object with...
  - A unique name e.g. `S28:waterPressure`
  - Controllable properties (fields) e.g. `EGU`
  - A behavior - defined by its record type
  - Optional associated hardware I/O (device support)
  - Links to other records

- Each field can be accessed individually by name
- A record name and field name combined give the name of a process variable (PV)
- A Process Variable name is what Channel Access needs to access data
A Process Variable name

- A PV name is comprised of two parts
  - The record name, and
  - The name of a field belonging to that record
- For example...

```
L1:water:temperature.
```

- Note that if no field name is given, Channel Access will default to using the .VAL field
What do records do?

- Records are active, they do things
  - Get data from other records or from hardware
  - Perform calculations
  - Check values are in range and raise alarms
  - Put data to other records or to hardware
  - Activate or disable other records
  - Wait for hardware signals (interrupts)
- What a record does depends upon its type and the values in its fields
- A wide range of records have already been created
- New record types can be added to a new application as needed
- A record does nothing until it is **processed**
Record types

- Classified into four general types
- Input: e.g.
  - Analog In (AI)
  - Binary In (BI)
  - String In (SI)
- Algorithm/control: e.g.
  - Calculation (CALC)
  - Subroutine (SUB)
- Output: e.g.
  - Analog Out (AO)
  - Binary Out (BO)
- Custom: e.g.
  - Beam Position Monitor
  - Multi Channel Analyzer
Some record types

- Analog in
- Analog out
- Binary in
- Binary out
- Calculation
- Calculation out
- Compression
- Data fanout
- Event
- Fanout
- Histogram
- Motor
- Multi bit binary input
- Multi bit binary output
- PID control
- Pulse counter
- Pulse delay
- Scan
- Select
- Sequence
- String in
- String out
- Subarray
- Subroutine
- Waveform
Graphical view of a record
Introduction to EPICS

record(ao, "DemandTemp") {
  field(DESC, "Temperature")
  field(ASG, "")
  field(SCAN, "Passive")
  field(PINI, "NO")
  field(PHAS, "0")
  field(EVNT, "0")
  field(DTYP, "VMIC 4100")
  field(DISV, "1")
  field(SDIS, "")
  field(DISS, "NO_ALARM")
  field(PRIO, "LOW")
  field(FLNK, "")
  field(OUT, "#C0 S0")
  field(OROC, "0.0e+00")
  field(DOL, "")
  field(OMSL, "supervisory")
  field(OIF, "Full")
  field(PREC, "1")
  field(LINR, "NO CONVERSION")
  field(EGUF, "100")
  field(EGUL, "0")
  field(EGU, "Celcius")
}

field(DRVH, "100")
field(DRLVL, "0")
field(HOPR, "80")
field(LOPR, "10")
field(HIHI, "0.0e+00")
field(LOLO, "0.0e+00")
field(HIGH, "0.0e+00")
field(LOW, "0.0e+00")
field(HHSV, "NO_ALARM")
field(LLSV, "NO_ALARM")
field(HSV, "NO_ALARM")
field(LSV, "NO_ALARM")
field(HYST, "0.0e+00")
field(ADEL, "0.0e+00")
field(MDEL, "0.0e+00")
field(SIOL, "")
field(SIML, "")
field(SIMS, "NO_ALARM")
field(IVOA, "Continue normally")
field(IVOV, "0.0e+00")
EPICS Databases – What are they?

- A collection of one or more EPICS records of various types
- Records can be interconnected and are used as building blocks to create applications
- A data file that’s loaded into IOC memory at boot time
- Channel access talks to the IOC memory copy of the database
Our First Database

Tell me about ProcTemp

ProcTemp
. VAL   = 45.5
. EGU   = deg C
. STAT  = Normal

Channel Access Client

Channel Access Server

IOC

Database

Analog In

ProcTemp

INP

VAL

EGU : deg C

EGUL: 0

EGUF: 100

HIGH: 51

0 – 100°C

0 – 10V

8 bit ADC

0 – 255 bits

45.5 deg C

116 counts

Normal Operation

5 - 50°C

0 – 10V

Temperature Sensor

Process

45.5°C

4.55V

Introduction to EPICS
Record Processing

- Record processing can be periodic or event driven
- Periodic: Standard scan rates are...
  - 10, 5, 2, 1, 0.5, 0.2 and 0.1 seconds
  - Custom scan rates can be configured up to speeds allowed by operating system and hardware
- Event driven: Events include
  - Hardware interrupts
  - Request from another record via links
  - EPICS Events
  - Channel Access Puts
Introduction to EPICS

Problem:
In the LINAC we have a water chiller that must be turned ON whenever the average temperature of two temperature sensors rises above a set point. The set point is nominally 10 degrees centigrade.

Database Processing

**Calculation**

L1:water:tempChk

**CALC:** \( \frac{(A+B)}{2} > C \)

**C:** 10

**SCAN:** 10 second
Inside an IOC

The major software components of an IOC (IOC Core)
The Sequencer

- Runs programs written in State Notation Language (SNL)
- SNL is a ‘C’ like language to facilitate programming of sequential operations
- Fast execution - compiled code
- Programming interface to extend EPICS in the real-time environment
- Common uses
  - Provide automated start-up sequences like vacuum or RF where subsystems need coordination
  - Provide fault recovery or transition to a safe state
  - Provide automatic calibration of equipment
SNL implements State Transition Diagrams

State A

Transition A to B

Event Action

State B
State Transition Example

Start

Low vacuum

- Pressure > 5.1 uTorr: Open the valve

High vacuum

- Pressure < 4.9 uTorr: Close the valve

Introduction to EPICS
Inside an IOC

The major software components of an IOC (IOC Core)

I/O Hardware

Device Support

Database

Sequencer

Channel Access

IOC

LAN
Device Support

- Device and driver support interface hardware to the database
- Examples of devices….
- VME cards
- ‘Network Attached Devices’
  - Motor controllers
  - Oscilloscopes
  - PLCs
- ‘Message-Based Devices’
  - USB or RS-232 serial devices
  - GPIB devices
Device Support

• Usually has to be written for ‘new’ hardware
• Good news – someone, somewhere has usually written support for your device, or a very similar one before
• See the EPICS web site for available support
• Or ask the EPICS community
When to use databases

- Hardware connection
- Real time performance – no network latencies
- Whenever a database is good enough

<table>
<thead>
<tr>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simplify hardware connection</td>
<td>If you have device support</td>
</tr>
<tr>
<td>Configuring not programming.</td>
<td>You need to understand database use</td>
</tr>
<tr>
<td>Database is easily understood by other EPICS developers</td>
<td></td>
</tr>
<tr>
<td>Speed - All processing (often) in same machine</td>
<td></td>
</tr>
</tbody>
</table>
When to use the sequencer

- For sequencing complex events
- E.g. Parking and unparking a telescope mirror
When to use clients

- To interact with the control system
- Many already exist – EDM/CSS, ALH, Strip Tool, archiver etc.
- For data analysis or visualization
- Supervisory control
- E.g. to manage an accelerator
How fast is EPICS?

- Can be fast or slow, it depends how you use it!
- Use the correct tool for the job; Database, sequencer, custom code (IOC) or custom code (client)
- Ultimately speed depends upon hardware
- Some benchmarks*:

<table>
<thead>
<tr>
<th>Machine</th>
<th>OS</th>
<th>CPU</th>
<th>Speed</th>
<th>Rec/sec</th>
<th>%CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>MVME167</td>
<td>vxWorks</td>
<td>68040</td>
<td>33MHz</td>
<td>6000</td>
<td>50</td>
</tr>
<tr>
<td>MVME 2306</td>
<td>vxWorks</td>
<td>PPC604</td>
<td>300MHz</td>
<td>10000</td>
<td>10</td>
</tr>
<tr>
<td>MVME5100</td>
<td>vxWorks</td>
<td>PPC750</td>
<td>450MHz</td>
<td>40000**</td>
<td>10**</td>
</tr>
<tr>
<td>PC</td>
<td>Linux</td>
<td>PII</td>
<td>233MHz</td>
<td>10000</td>
<td>27</td>
</tr>
<tr>
<td>PC</td>
<td>Linux</td>
<td>P4</td>
<td>2.4GHz</td>
<td>50000</td>
<td>9</td>
</tr>
</tbody>
</table>

*Benchmark figures courtesy of Steve Hunt (PSI)
**Extrapolated from performance figures provided by L.Hoff, BNL

- Database design and periodic scanning effect apparent system speed
Apparent performance

Introduction to EPICS
The EPICS web site

- The central site for EPICS information
- Documentation
- CA Clients
- Device support
- Tech-talk
Acknowledgements

- Andrew Johnson (APS-Controls)
- Bob Dalesio (LANL)
- Deb Kerstiens (LANL)
- Rozelle Wright (LANL)