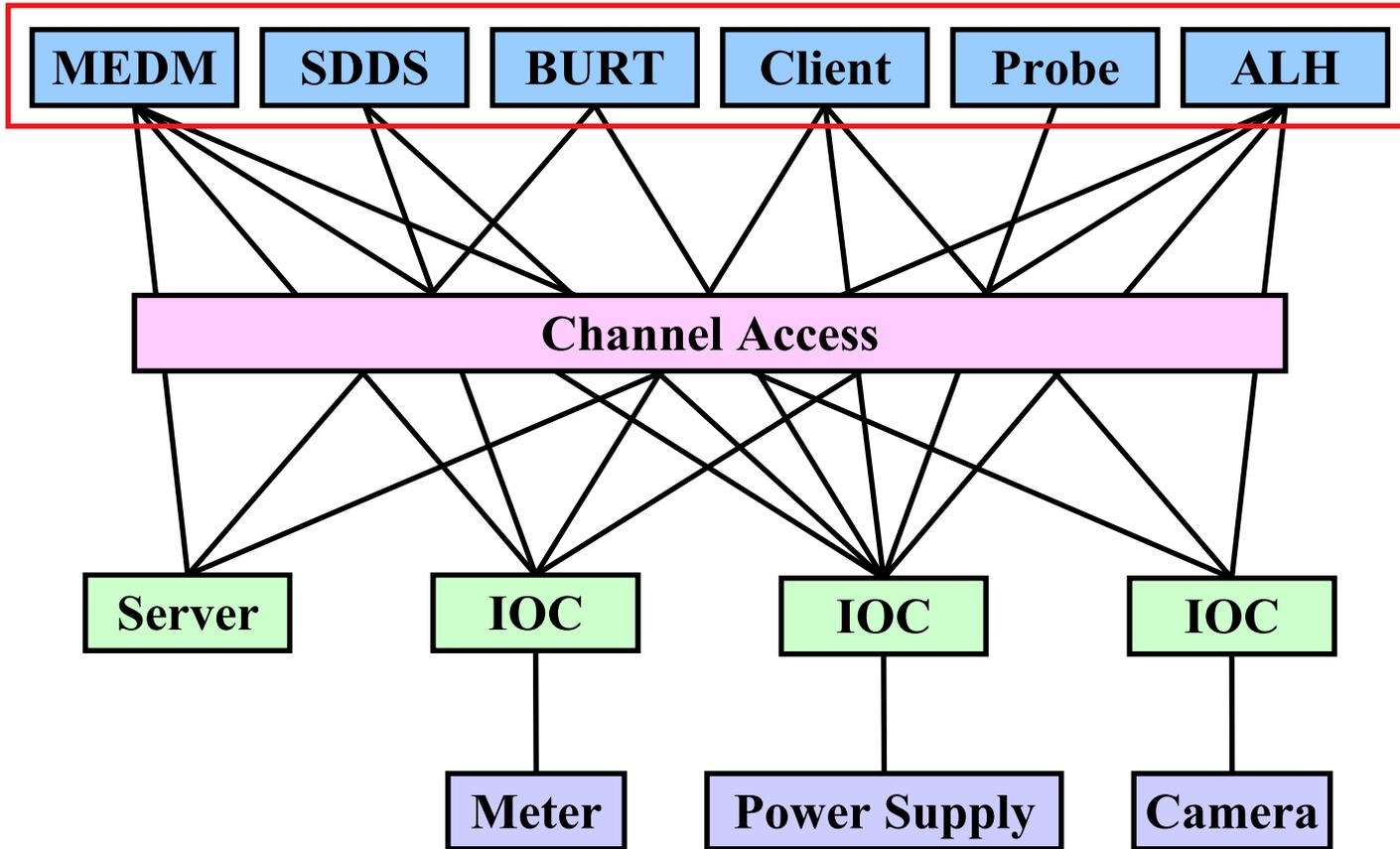


# Channel Access and Client Tools

Author: Kenneth Evans, Jr., August 2004  
Modified: Kay Kasemir, October 2006  
Andrew Johnson, January 2007 & 2011

# EPICS Overview

## Client Tools



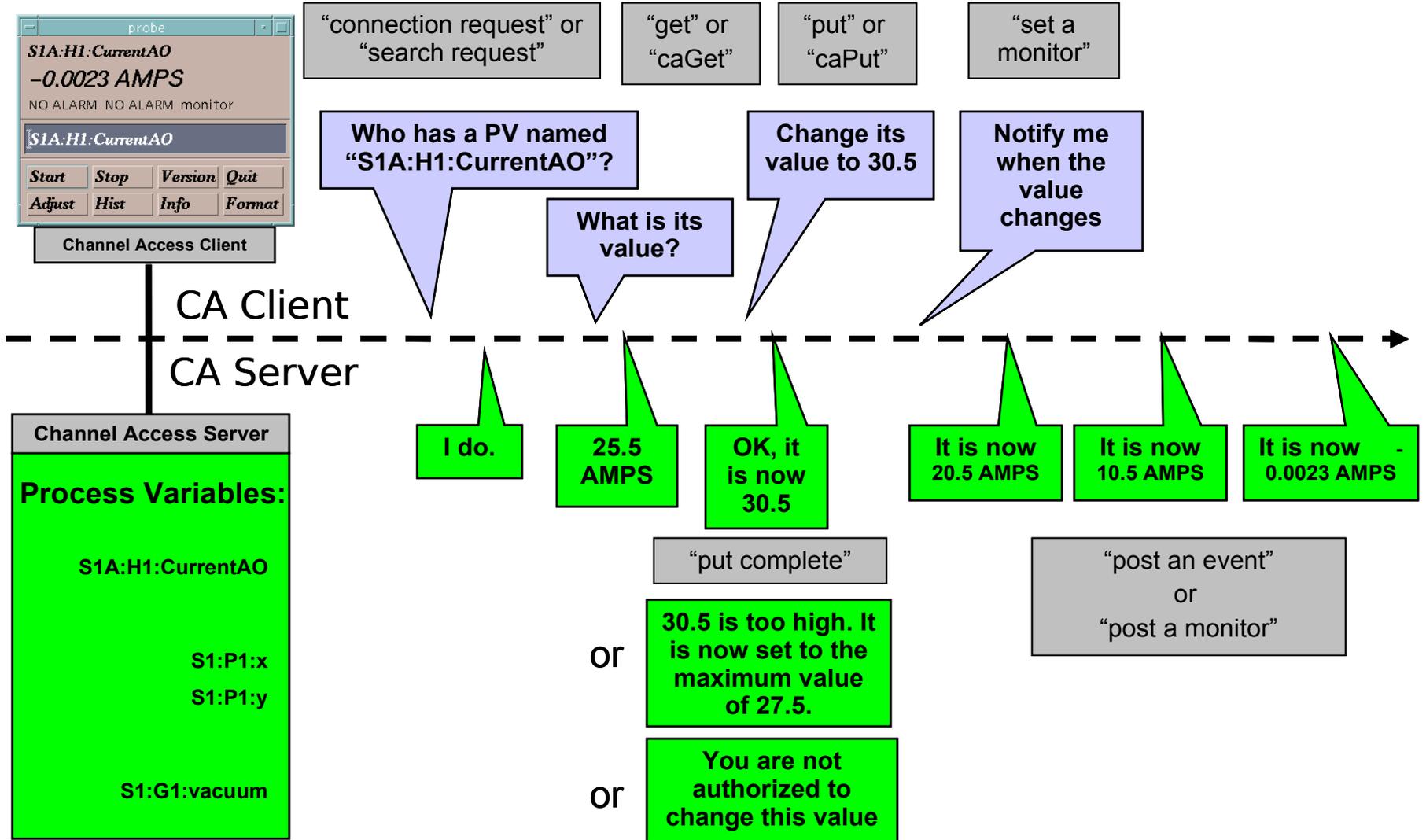
# Channel Access

- The EPICS “software bus”
- Used to read and write values to/from Process Variables
- To many people, Channel Access *is* EPICS
  - Especially those that have no IOC experience
  - “Integrate X into EPICS” often means “Be able to control X via CA”
- CA is not defined by a protocol specification
  - Jeff Hill (LANL) maintains the CA client and server libraries
  - A single expert maintainer for both ensures very robust control systems

# What is a Process Variable (PV)

- “A named item of data, with associated optional attributes”
  - Data is an Integer, Floating point number, enumeration value or string, or an array of any of those types
  - Attributes include timestamp, alarm status/severity, precision, engineering units string, list of enumeration strings, operator/control/alarm limits

# Channel Access in One Slide



# Tools Described in This Presentation

- Base command-line tools
  - caget, caput, camonitor, cainfo
- Probe
- StripTool
- ALH
- MEDM



# More Information

- There is a wealth of information in the EPICS web pages
  - <http://www.aps.anl.gov/epics/index.php>
- Each of the Extensions covered here has its own page there for
  - Documentation
  - Source code for the latest releases
- There are *many* other tools described there too
  
- Extensions **executables** are typically located at
  - ...epics/extensions/bin/<platform>/<executable>
    - /usr/local/epics/extensions/bin/solaris-sparc/...
  - Platforms are solaris-sparc, linux-x86, linux-x86\_64, win32-x86, etc.
- The Base command line tools are typically at
  - ...epics/base/bin/<platform>/<executable>
    - /usr/local/iocapps/R3.14.11/base/3-14-11-asd1/bin/linux-x86\_64/...

# EPICS Extensions Web Page

**EPICS** Experimental Physics and Industrial Control System

Advanced Photon Source  
ARGONNE NATIONAL LABORATORY

## Extensions

The following list gives access to individual pages for most of the standard EPICS host tools and CA clients. Note that some of the minor pages linked below do not appear in the sidebar on the left.

Some of this software can be downloaded from the individual web-pages linked below, and the collection of tools from APS are also available bundled together. See the [Extensions Download](#) page for details.

If your extension does not appear in this list, or there's something wrong with an entry on this page, please [send me an email](#), giving a URL for your web-site if applicable.

### Config Files

- [Extensions build config files \(R3.13\)](#)
- [Extensions build configure files \(R3.14\)](#)

### Standalone CA Clients

- [ADT: Array Display Tool](#)
- [ALH: Alarm Handler](#)
- [AR: Data Archiver](#) (the original, deprecated)
- [BURT: Backup and Restore Tool](#)
- [CAEX: Channel Access Examples](#)
- [CASR: Host-based Save/Restore](#)
- [CAU: Channel Access Utility](#)

# Command-Line Tools

- There used to be several versions of these tools
- We will discuss the ones that now come with EPICS Base
- The tools we will cover are:
  - caget
    - *Gets the value of one or more process variables*
  - caput
    - *Sets the value of one process variable*
  - camonitor
    - *Monitors the value changes of one or more process variables*
  - cainfo
    - *Gets information about one or more process variables*
- All accept –h to display usage and options
- NOTE: The programs with these names found in your default Unix search path are different (older).



# Caget Example

- Get the values of two process variables

```
caget S35DCCT:currentCC S:SRlifeTimeHrsCC
```

- Returns

```
S35DCCT:currentCC      102.037
```

```
S:SRlifeTimeHrsCC     7.46514
```

# Caput Example

- Set the value of a process variable

```
caput Xorbit:S1A:H1:CurrentAO 1.2
```

- Returns

```
Old : Xorbit:S1A:H1:CurrentAO      0
```

```
New : Xorbit:S1A:H1:CurrentAO      1.2
```

# Camonitor Example

- Monitor two process variables

```
camonitor evans:calc evans:bo01
```

- Returns

```
evans:calc      2004-08-05 17:23:04.623245 1
evans:bo01     2004-08-05 17:23:04.623245 On
evans:calc      2004-08-05 17:23:05.123245 2
evans:bo01     2004-08-05 17:23:05.123245 Off
evans:calc      2004-08-05 17:23:05.623245 3
evans:calc      2004-08-05 17:23:06.123245 4
evans:calc      2004-08-05 17:23:06.623233 5
evans:calc      2004-08-05 17:23:07.123183 6
```

- Use Ctrl-C to stop monitoring

# Cainfo Example

- Get information about a process variable

```
cainfo S35DCCT:currentCC
```

- Returns

```
State:      connected
```

```
Host:      ctlapps41188:5064
```

```
Access:    read, no write
```

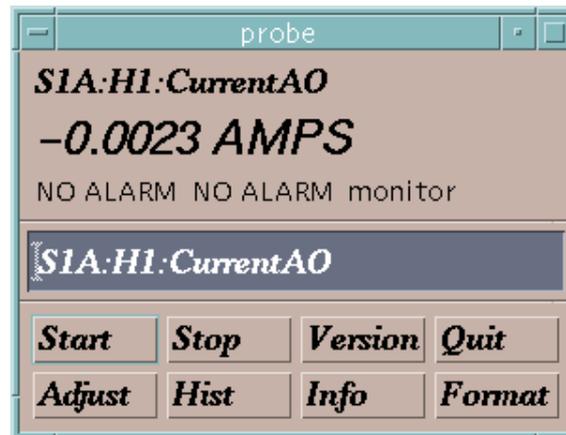
```
Data type: DBR_DOUBLE (native: DBF_DOUBLE)
```

```
Element count: 1
```

- Some additional information can be found using Probe

# Probe

- Simple way to get information about a single process variable
- Combines the features of caget, caput, camonitor, and cainfo in a graphical interface
- Very useful in diagnosing problems



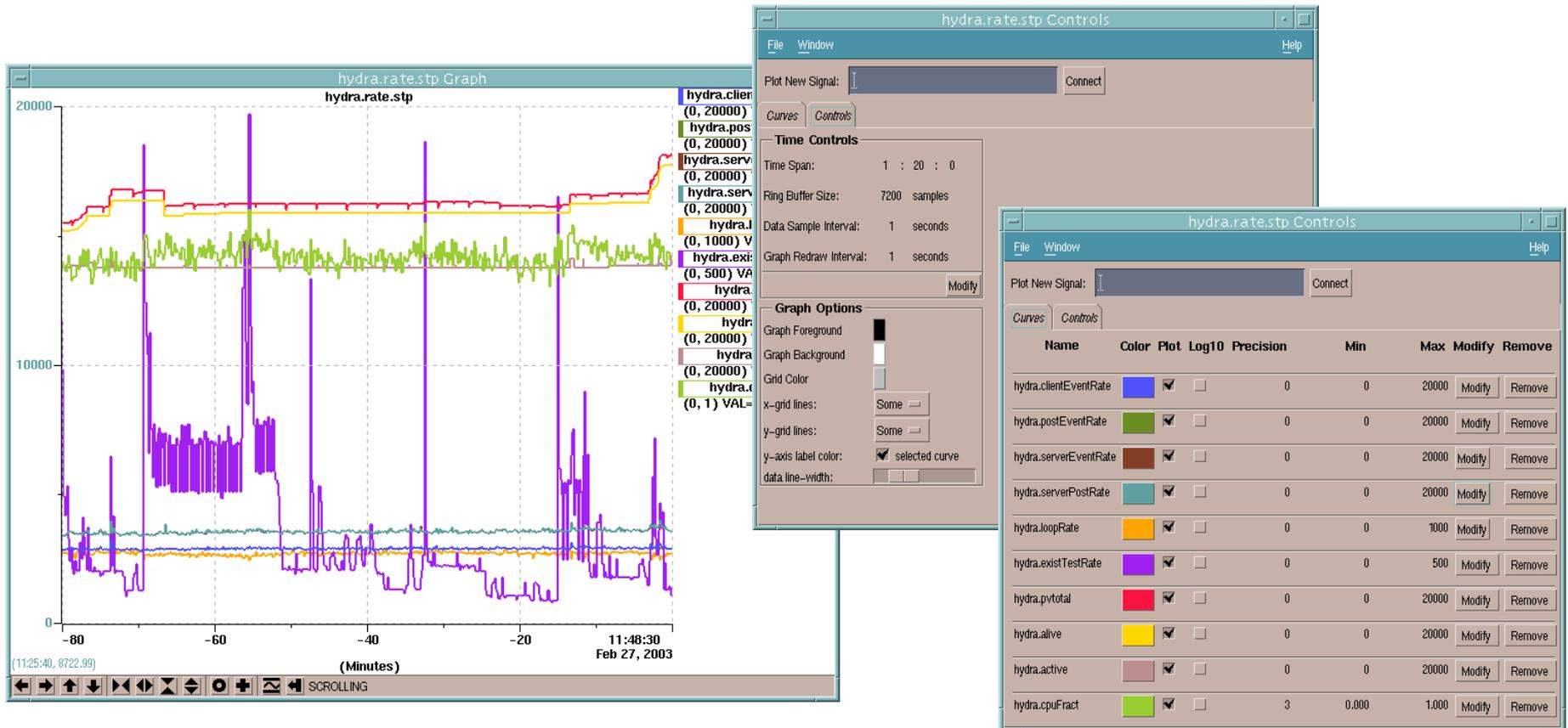
# Probe Demo

## Flash Demonstration of Probe



# StripTool

- Plots process variables in real time on a strip chart
- Widely used



# StripTool Demo

Flash Demonstration of StripTool



# ALH

- Stands for Alarm Handler
- Important GUI application in the APS Control Room
- Brings alarms to the operators' attention
  - It dings and flashes
- Can be configured to require the operator to acknowledge alarms
- Provides a hierarchical display
  - Allows managing alarms in overview or in detail
- Provides guidance for handling specific alarms
- Logs alarms and displays alarm history
  
- The APS Operators configure the Alarm Handler



# MEDM

- Stands for Motif Editor and Display Manager
- The principal human interface to the APS control system
- Used worldwide at many facilities
- Creates and runs control screens



# MEDM Screens

The image displays a collection of MEDM (Machine Element Display Monitor) control screens for a particle accelerator. The screens are arranged in a collage, showing various operational parameters and control elements:

- Beam Current Monitor:** Shows "Beam Current: 102.1 mA" and "Lifetime: 0.0 Hours". It includes a "Beam Current History" graph and a "Storage Ring" schematic diagram.
- WAVEGUIDE SWITCH MONITOR:** Displays a schematic of waveguide switches (S36, S37, S38, S40) and their status. It includes a "Relay Switching Status" table.
- Booster RF Ramp Controls:** Shows "RF Ramp Signal", "Klystron Amp Output Signal", and "Cavity Gun Signal" plots. It includes an "Arbitrary Function Generator" and "Sun DAC" controls.
- MPS Overview:** A large table showing "SECTOR", "VALVES", "FLAGS", "ABSORBERS", and "UNUSED CONTROLLERS" with status indicators.
- LEUT Beamline:** Shows "LEUT PHR Tools OPS Diagnostics RF Water Timing Steering Focusing Vacuum Interlocks Equipment RFGTS" and a detailed schematic of the beamline.
- Booster Extraction Timing PreTrigger:** A control screen for the booster extraction timing, featuring "PreTrigger" and "Thick Section" controls.
- Storage Ring RFG:** Shows "Storage Ring RFG" controls, including "Global NewScan Controls" and "BPM Data Pool".

- And thousands of others

# MEDM

- MEDM is very reliable at both design and run-time
- However it is very hard to extend
  - Uses the Motif X11 toolkit
  - Not written in Object-Oriented style
  - Ken Evans retired last year
- We will now only fix major bugs found in MEDM
  - Eventually it may prove impossible to build on new Linux versions
- Not a good choice for a new control system
  - There are several alternatives available
  - John Hammonds will talk about CSS and BOY on Thursday